Language Elicits Compositional Reasoning in Pretrained VFMs

Jung-Chun Liu [1] **Naihao Deng** [1] **Joyce Chai** [1]

Abstract

Compositional reasoning is a core capability in embodied intelligence. A common assumption holds that reasoning resides in language models (LMs), while vision foundation models (VFMs) are treated as feature extractors. Motivated by classical cognitive theories of visual processing, we ask whether symbolic mechanisms can emerge inside the visual stream. Specifically, we apply language conditioning to a frozen pretrained vision transformer and measure whether intermediate visual geometry tracks question-derived semantic conditions across layers. We find that the model develops a three-stage mechanism: 1) feature binding segregates scenes containing the referenced entity in middle layers, 2) object grounding resolves the binding cluster into object-level substructure in later layers, and 3) answer matching repartitions representations into answer-aligned groups in the final layer. Representational and mechanistic analyses support this hierarchy, suggesting that language-conditioned vision encoders can instantiate symbol-like processing as query-dependent geometric organization.

1. Introduction

Compositional reasoning is a core capability in embodied intelligence (Lake et al., 2017; Ke et al., 2025). For example, given the question "What is the material of the red cube?", we can compose the attributes *red* and *cube* into the entity *red cube*; ground the 'red cube' in the question to an object 'big red rubber cube' in the scene; finally query the material of the grounded object to produce the answer.

Following the long-standing argument that compositionality demands constituent structure (Fodor & Pylyshyn, 1988), previous work has investigated whether neural networks learn compositional representations and symbol processing (Greff et al., 2020; Feng & Steinhardt, 2024; Webb et al., 2021; 2024; Yang et al., 2025; Assouel et al., 2025; Griffiths et al., 2026). Within VLMs, several mechanisms have been identified in language models (LMs), including binding (Assouel et al., 2025; Saxena et al., 2025; Feng & Steinhardt, 2024), grounding (Pavlick, 2023; Wu et al., 2025a; Kang et al., 2025; Neo et al., 2025), and answer retrieval (Yang et al., 2025; Wu et al., 2025b). However, a common assumption holds that reasoning resides in LMs, while vision foundation models (VFMs) are treated as feature extractors. Whether VFMs develop compositional reasoning and corresponding mechanisms remains underexplored.

Motivated by classical cognitive theories of visual processing and perceptual representation (Treisman & Gelade, 1980; Shepard & Metzler, 1971; Marr, 1982; Biederman, 1987; Barsalou, 1999), we posit that compositional reasoning can also reside in the visual stream and can be elicited through language conditioning. Specifically, we hypothesize that symbol-processing mechanisms emerge in a frozen pretrained vision transformer (ViT) by injecting language-conditioned cross-attention. The hypothesis rests on two architectural principles. First, in the residual stream, each layer accumulates the contributions of preceding layers (Elhage et al., 2021; Ferrando et al., 2024). Components can write operations into this additive stream sequentially, where later layers can compose and refine earlier representations. Second, cross-attention injects question-related information into the visual stream, where semantic features are already encoded. The language conditioning routes intermediate visual representations toward semantic interpretations specified by the question. Together, these properties enable the model to progressively reshape geometry, in which answer distinctions become linearly accessible.

To test this hypothesis, we build a controlled analysis framework on CLEVR (Johnson et al., 2017). We first evaluate the performance across different ViT backbones and ablations on each component. We further evaluate compositional generalization on CoGenT, including zero-shot generalization and fine-tuning.

We next conduct representational analysis and formulate specific semantic conditions as predicates to probe feature geometry. Our representational analysis identifies a hierarchical three-stage organization of symbol-like processing.

[*]Equal contribution [1]Department of Computer Science and Engineering, University of Michigan, Ann Arbor, USA. Correspondence to: Joyce Chai <jychai@umich.edu>.

Accepted to the 2nd Workshop on Compositional Learning at ICML 2026, Seoul, South Korea. Copyright 2026 by the author(s).

We characterize the mechanisms as follows: 1) *feature binding* segregates scenes in which the question-referenced entity exists; 2) *object grounding* resolves the binding cluster into substructure that distinguishes object-level matches; and 3) *answer matching* repartitions the representations into answer-aligned groups. Each stage operates conditionally on the preceding one, reflecting compositional queries for multi-hop reasoning. Further mechanistic analysis identifies attribute-binding heads at CA L5-7 that support binding for specific attributes and answer-matching heads at SA L11. Noising these heads disrupts the corresponding structure and propagates downstream, degrading binding separation and answer-matching repartition.

Our contributions are threefold.

- We introduce a predicate-based framework for analyzing intermediate symbolic structure in language-conditioned vision encoders.
- We show that visual representations develop a staged organization corresponding to feature binding, object grounding, and answer matching, with trajectories that adapt to question structure.
- We identify attention heads whose interventions selectively disrupt predicate-aligned geometry, providing causal evidence that these components support feature binding and answer matching.

2. Language-Conditioned Visual Reasoning

2.1. Problem Formulation

In visual reasoning tasks, given a visual input v and a question q, a model $\mathcal{M}$ maps (v, q) to an answer y:

$$\mathcal{M} : (v, q) \mapsto y. \tag{1}$$

In compositional reasoning, the model maps (v, q) to y via an intermediate structured representation (Ke et al., 2025).

2.2. Architecture

A vision transformer (ViT) maps an image to patch tokens $z^{(0)} \in \mathbb{R}^{N \times d}$, obtained by linearly projecting non-overlapping image patches, and updates them through L transformer blocks. Each block applies self-attention (SA) and feed-forward multi-layer perceptron (MLP) with residual connections:

$$z^{(\ell)} = z^{(\ell-1)} + \mathrm{SA}^{(\ell)}(z^{(\ell-1)}) + \mathrm{MLP}^{(\ell)}(z^{(\ell-1)}).$$

Unrolling this recursion yields the residual-stream view (Elhage et al., 2021; Ferrando et al., 2024), each layer writes to a shared additive stream, and the representation at layer

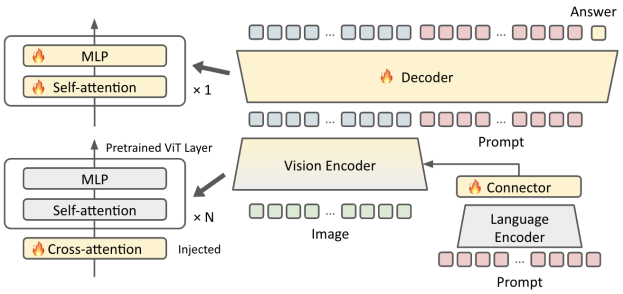

Figure 1. The general architecture of the framework. The vision encoder backbone and language encoder are frozen, and the injected cross-attention, connector, and downstream decoder are trainable.

ℓ accumulates the contributions of preceding layers. This view motivates an analysis of how language conditioning steers the feature geometry.

To enable language conditioning, we inject cross-attention (CA) modules with a $\tanh$ gate before self-attention in selected transformer blocks. Each CA module conditions the visual tokens on text tokens z_t and writes the result into the visual residual stream:

$$\tilde{z}^{(\ell)} = z^{(\ell-1)} + \tanh(\alpha^{(\ell)}) \cdot \mathrm{CA}^{(\ell)}(z^{(\ell-1)}, z_t),$$
$$z^{(\ell)} = \tilde{z}^{(\ell)} + \mathrm{SA}^{(\ell)}(\tilde{z}^{(\ell)}) + \mathrm{MLP}^{(\ell)}(\tilde{z}^{(\ell)}).$$

Here z_t denotes the text-token representations, and $\alpha^{(\ell)}$ is a learned gate initialized near zero.

2.3. Setup

The proposed model comprises a ViT backbone with gated cross-attention, a text encoder with a connector, and a decoder to predict output as illustrated in Figure 1. See Appendix A for more implementation details.

ViT backbone. We build on a pretrained ViT backbone (Oquab et al., 2024) with gated cross-attention appearing at alternating transformer blocks (e.g., layers 1, 3, 5, 7, 9, and 11 for a 12-layer ViT). The choice of gates and CA placement follows prior early-fusion architectures for stability (Alayrac et al., 2022; Ruthardt et al., 2026).

Text encoder. Text queries are encoded by a frozen RoBERTa-Large model (Liu et al., 2019) and projected to the visual dimension via a learned linear connector. The module produces text embeddings z_t fed into CA.

Readout head. We select two types of readout heads for evaluation. The decoder head is a one-layer transformer decoder (SA+MLP) where inputs are concatenated visual patches and text tokens. The vocabulary contains 28 CLEVR answer categories.

Training. We adopt CLEVR (Johnson et al., 2017) as a controlled testbed. The dataset generates the question and scene through a program. We train on 700K training questions and evaluate on 150K validation questions across 28 answer classes. The CA modules with gates, connector, and decoder are randomly initialized, while the visual and text backbones remain frozen.

2.4. Comparison across ViT Backbones

We first compare different ViT backbones under the same language-conditioned architecture. We evaluate on four backbones: DINOv2, SigLIP, MAE, and an ImageNet-21K supervised ViT. This comparison tests whether different pretrained ViTs provide similarly effective bases for language-conditioned visual reasoning.

Results. Table 1 shows that the choice of visual backbone affects performance under language-to-vision cross-attention. DINOv2 and SigLIP achieve high overall and query-attribute accuracy, indicating that language conditioning yields reasoning with semantic visual representations from pretrained ViTs. In contrast, MAE performs much worse, suggesting that reconstruction-oriented pretraining provides a weaker substrate for this form of language-conditioned visual reasoning. See Appendix B for full evaluation results.

2.5. Ablation Studies

We then ablate the architectural components using DINOv2 as the default backbone. We replace the decoder with a linear classifier, train the ViT backbone from scratch, remove gated cross-attention so that the visual stream receives no language conditioning, and replace the frozen RoBERTa encoder with a learned 1-layer text encoder. These ablations isolate the contributions of pretrained visual features, language-conditioned cross-attention, pretrained language representations, and the readout head.

Results. As shown in Table 2, replacing the decoder with a classifier slightly reduces performance from 92.4% to 90.1%, suggesting that the decoder is not solely responsible for the model's performance. Training the ViT backbone from scratch causes a large drop to 52.8%, indicating that pretrained visual representations are essential. Removing gated cross-attention reduces accuracy to 49.4%. This suggests that decoder capacity alone, without language-conditioned visual representations, is insufficient. Finally, replacing RoBERTa with a learned text encoder leads to the largest drop, reaching 24.6%, suggesting that pretrained language representations provide crucial semantic structure for effective language-to-vision conditioning. Together, these ablations show that the model's performance depends on the interaction between pretrained visual features, pretrained language representations, and language conditioning. See Appendix B for full evaluation results.

Table 1. Accuracy across ViT backbones on CLEVR validation set. As we focus on attribute queries in this paper, we specifically report query-attribute accuracy. Results are averaged over 3 seeds.

ViT Backbone	Query Attribute	Overall
DINOv2	**99.2**	**92.4**
SigLIP	**99.0**	**92.6**
Supervised-ViT	94.0	86.6
MAE	92.1	74.8

Table 2. Ablation study on CLEVR validation set. Results are averaged over 3 seeds.

Ablation	Overall
DINOv2 + decoder	92.4
- Decoder + classifier	90.1
ViT backbone from scratch	52.8
- CA (no language conditioning)	49.4
- RoBERTa + learned text encoder	24.6

2.6. Compositional Generalization on CoGenT

We further evaluate compositional generalization on CLEVR-CoGenT, where in condition A cube colors are only {gray, blue, brown, yellow}, and cylinder colors are only {red, green, purple, cyan}, with condition B reversing the assignment. The model is trained on condition A and evaluated on both conditions A and B. Pure neural architecture baselines exhibit substantial gaps at 15-20%, while neuro-symbolic methods enforce disentanglement architecturally to reach near-zero gaps. The model achieves ValA at 92.4%, ValB at 88.0%, with an overall gap 4.4% without explicit symbolic structure.

We fine-tune the model on Condition B to evaluate sample efficiency and catastrophic forgetting. Fine-tuning is conducted for 8 epochs using the same learning rate and training setup. As the number of Condition B samples increases, ValA and ValB converge to 92.7% overall accuracy, close to the CLEVR validation performance of 92.4% in Section 2.4. ValB improves by 4.0 points, while ValA drops by only 1.6 points, showing that the model adapts efficiently to the new condition while largely avoiding catastrophic forgetting.

Beyond CoGenT, we further evaluate zero-shot and fine-tuning transfer on several CLEVR variants, including CLEVR-Humans, CLEVR-Math, and CLOSURE. These benchmarks test generalization to human-written questions, arithmetic reasoning, and novel compositional structures, respectively. The full results are reported in Appendix C.

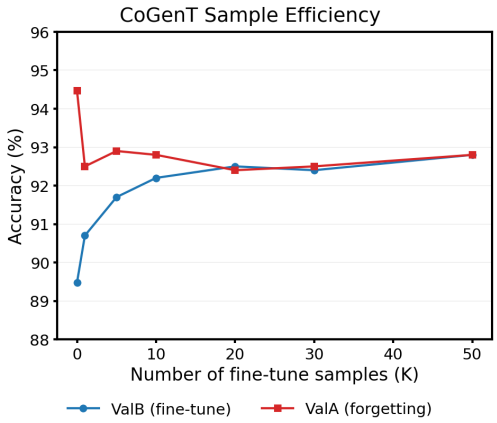

Figure 2. The zero-shot and fine-tuning accuracy and sample efficiency on CoGenT. The accuracy converges after fine-tuning on more than 20K samples.

3. Representational Analysis

Having established the model's performance and compositional generalization, we next ask how language conditioning reshapes intermediate visual representations. We first visualize the geometric structures at each layer, and then verify our hypothesis with representational similarity analysis. In the following analyses, we use the DINOv2 backbone because it provides strong self-supervised visual features without direct language supervision or supervised classification labels.

3.1. Predicate Representation

For analysis, each v is associated with an underlying scene X, represented as a set of objects. An object in the scene is denoted by $o \in X$. Object attributes are denoted by $\mathit{attr}(o)$ (*e.g.*, $\mathit{color}(o)$), and pairwise relations between objects are denoted by $\mathit{rel}(o_i, o_j)$ (*e.g.*, $\mathit{behind}(o_i, o_j)$). A scene graph $G(X)$ represents the objects, their attributes, and their pairwise relations in X:

$$G(X) = \{(o_i, \mathit{attr}(o_i), \mathit{rel}(o_i, o_j)) : o_i, o_j \in X\}. \quad (2)$$

This paper focuses on query-attribute questions, where q specifies a set of referent objects and a unique queried object in X, and the answer a is an attribute of the queried object. These questions include direct attribute queries, same-attribute relations, and spatial relations. Each question type corresponds to a program template that exposes the multi-step structure required to answer it.

To probe feature geometry in the visual stream, we formulate the semantic conditions as *predicates*. Given a scene X with scene graph $G(X)$, we express a semantic condition as a predicate $P(X, q)$ whose truth value can be verified from $G(X)$. For example, the phrase "red cube" defines an object predicate $\mathit{red}(o) \wedge \mathit{cube}(o)$, and the statement "a red cube exists in the scene" defines a scene predicate $\exists o \in X$ s.t. $\mathit{red}(o) \wedge \mathit{cube}(o)$. The conversion enables us to measure whether such predicates can be traced in the representation space, and whether this predicate-aligned geometry emerges at specific layers.

3.2. Operationalizing Symbol-Like Processing

Throughout this paper, we operationalize symbol processing as geometry shift in the representation space. Given an image-question pair (v, q) and its scene X_q with unique query object $o_q \in X_q$, we define predicates for three stages and ask whether representations are aligned to these predicates across layers. $A(o)$ denotes the attribute set of o, A_d denotes the described attribute set explicitly specified by the question, and $A_q(o)$ denotes the query attribute from o. In this work, we assume each object has a fixed set of attribute dimensions, and each dimension takes a unique value.

Feature binding. Feature binding is the segregation of scenes according to whether they contain an entity satisfying the attributes mentioned in the question. A scene X satisfies the feature-binding predicate if it contains an object whose described attributes are jointly satisfied.

$$P_B(X, q) \iff \exists o \in X \text{ s.t. } A(o) \supseteq A_d.$$

For example, if the question asks "What is the material of the red cube?", this stage draws out scenes that contain a red cube from scenes that do not, and the predicate tests whether a red cube exists in X: $\exists o \in X$ s.t. $\mathit{red}(o) \wedge \mathit{cube}(o)$.

Object grounding. Object grounding is the formation of object-level substructure within the binding-positive set. Among scenes that contain a red cube, this stage organizes representations according to specific object matches in the scene, such as a small matte red cube versus a large shiny red cube. A scene X satisfies the object-grounding predicate if it contains an object matching the query object in X_q across the full attribute specification.

$$P_G(X, q) \iff \exists o \in X, o_q \in X_q \text{ s.t. } A(o_q) \supseteq A_d \wedge A(o) = A(o_q).$$

This condition refines described attributes such as "red cube" into an object-level match "small red rubber cube" in the query scene.

Answer matching. Answer matching repartitions representations by the value of the question-specified query attribute, organizing scenes into answer-aligned groups. A scene satisfies the answer-matching predicate if it yields the same answer as the query scene under the same question program. For direct attribute queries, this means that X_r

contains an object matching the grounded object and sharing the queried attribute value:

$$P_A(X, q) \iff \exists o \in X \text{ s.t. } A(o) \supseteq A_d \wedge A(o_q) \supseteq A_d \wedge A_q(o) = A_q(o_q).$$

It is worth noting that P_G and P_A can be derived from P_B, indicating their dependency:

$$P_G(X, q) \iff \exists o \in X \text{ s.t. } P_B(X_q, q) \wedge A(o) = A(o_q),$$

$$P_A(X, q) \iff \exists o \in X \text{ s.t. } P_B(X, q) \wedge P_B(X_q, q) \wedge A_q(o) = A_q(o_q).$$

In direct queries, this predicate can coincide with object grounding when the full object specification already determines the queried attribute, while in spatial and same-attribute questions, answer matching depends on the output of the full question program rather than on a single object's attributes alone. Therefore, for spatial and same-attribute questions, we define feature binding and object grounding predicates separately for the anchor object and the target object. For answer matching, given anchor o^a and target o^t in X_q, we define a corresponding relation and the same queried attribute value:

$$\exists o^a, o^t \in X \text{ s.t. } \mathit{rel}(o^a, o^t) = \mathit{rel}(o^a_q, o^t_q) \wedge A_q(o^t) = A_q(o^t_q).$$

Together, these stages define our operational hypothesis. We test this by translating each stage into predicates over reference scenes and measuring the emergence of predicate-aligned geometry.

3.3. Layer-wise Visualization

We start by using t-distributed stochastic neighbor embedding (t-SNE) to visualize the geometric structure at each layer. t-SNE visualizes local neighborhood structure in low dimensions. We select one query and build a reference pool of 500 distinct images. CLS tokens at each layer are used for visualization. Each reference image is fed into the model conditioned with the identical question to generate representations and is labeled from its scene graph according to whether it satisfies the predicate. We apply t-SNE to the representations of the reference pool. This visualization provides diagnostic, qualitative evidence for the layer-wise geometric progression.

Result. As shown in Figure 3, scenes satisfying the feature-binding predicate first become separable around L7. By L9, the binding cluster develops object-grounding substructure, separating scenes according to whether they contain an object-level match to the query object. Finally, at L11, these substructures are repartitioned into subgroups aligned with the answer of the queried attribute, making answer distinctions accessible to the decoder. Notably, scenes that do not satisfy the feature-binding predicate do not undergo the same answer-aligned repartitioning, indicating that the geometric shift is question-dependent. This progression shows how language-conditioned cross-attention steers visual representations from feature binding, through object grounding, toward answer-matching structure.

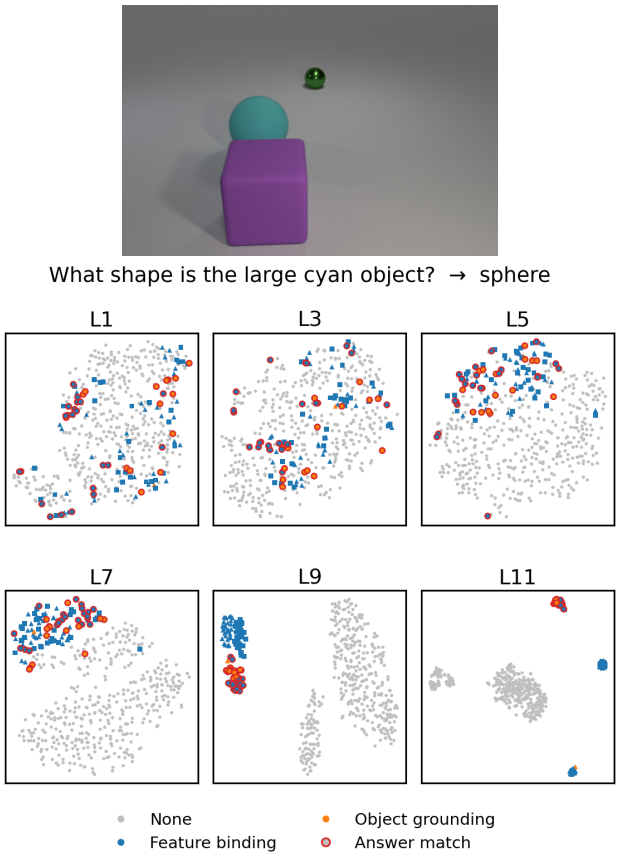

Figure 3. t-SNE results given the query "What shape is the large cyan thing?" → sphere, and an image that contains a large cyan shiny sphere. Each point is a scene, colored by the predicate satisfaction. Markers denote the queried attribute value: ○ for spheres, △ for cylinders, and □ for cubes. Feature binding becomes separable around L7-L9, and at L11 the cluster splits into answer-aligned subclusters, illustrating the hierarchy from binding to answer matching.

3.4. Representational Similarity Analysis

We further quantify stage-wise organization with conditional RSA. The goal is to test whether representations across layers group reference scenes according to the predicate

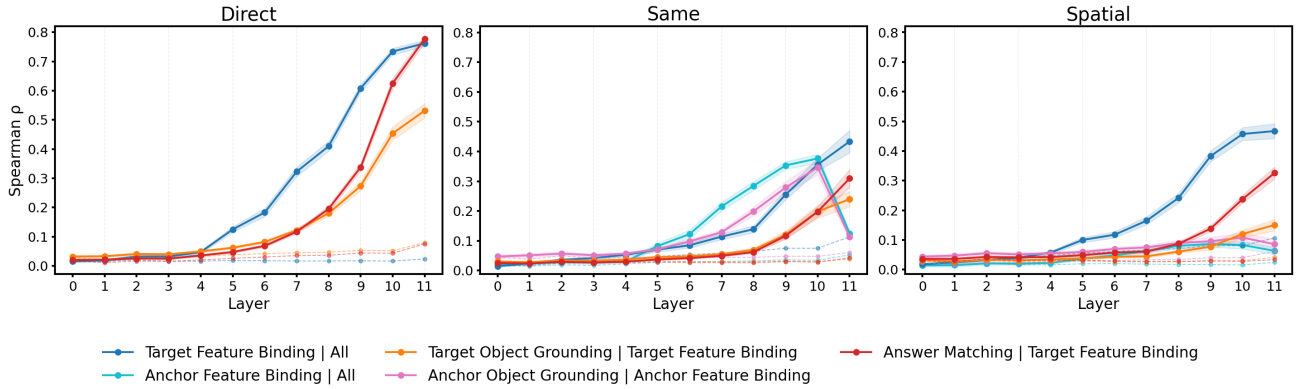

Figure 4. Conditional RSA across layers, evaluated on three question types. Solid lines denote the model with language conditioning, dashed lines denote the no-CA baseline, and the shaded area indicates the standard error of the mean. The results show that the stage-wise organization depends on question type. In direct attribute queries, binding, grounding, and answer matching rise progressively.

structure associated with a reasoning stage.

Representations and distances. Given a query scene X_q, a question q, and a reference pool $\mathcal{D} = \{X_1, \ldots, X_n\}$, we extract layer-$\ell$ representations $\{H_\ell(X_i \mid q)\}_{i=1}^n$ from the vision encoder. We compute the neural distance matrix $D^\ell \in \mathbb{R}^{n\times n}$ using correlation distance:

$$D^\ell_{ij} = 1 - \frac{\langle H_\ell(X_i \mid q) - \bar{H}_i,\ H_\ell(X_j \mid q) - \bar{H}_j\rangle}{\|H_\ell(X_i \mid q) - \bar{H}_i\|_2 \cdot \|H_\ell(X_j \mid q) - \bar{H}_j\|_2}, \tag{3}$$

Stage-specific RDMs. For each stage, we construct a semantic RDM whose entries indicate whether two reference scenes differ in the stage-relevant semantic value. Feature binding is evaluated over the full reference pool, since it tests whether a scene contains the question-described entity. Object grounding and answer matching are evaluated on the feature-binding subset, as both stages presuppose the referenced entity exists; they measure finer-grained organization within scenes satisfying the binding predicate. The stage-specific RDMs are defined as follows.

- **Feature binding.** The feature-binding RDM M_B is defined over the full reference pool $\mathcal{D}$. It separates scenes that satisfy the feature-binding predicate from scenes that do not:

$$M_\mathrm{B}[i,j] = \mathbb{1}\left[P_\mathrm{B}(X_i; q) \neq P_\mathrm{B}(X_j; q)\right]. \tag{4}$$

- **Object grounding.** The object-grounding RDM M_G is defined over the feature-binding subset $\mathcal{D}_\mathrm{B} = \{X \in \mathcal{D} : P_\mathrm{B}(X; q) = 1\}$. It groups scenes by the full attribute specification of the object satisfying the bound description:

$$M_\mathrm{G}[i,j] = \mathbb{1}\left[A(o_i^\star) \neq A(o_j^\star)\right], \tag{5}$$

where $o_i^\star$ denotes the object in X_i that satisfies the bound description. We define $o_i^\star$ as the unique object selected by the CLEVR program; scenes without a unique selected object are excluded.

- **Answer matching.** The answer-matching RDM M_A is also defined over the feature-binding subset. It groups scenes by the answer returned by the question program:

$$M_\mathrm{A}[i,j] = \mathbb{1}\left[\mathrm{Ans}(X_i, q) \neq \mathrm{Ans}(X_j, q)\right]. \tag{6}$$

For direct attribute queries, $\mathrm{Ans}(X_i, q)$ is the queried attribute value of the object satisfying the question description. For spatial and same-attribute questions, it is the output of the full question program.

Conditional RSA score. We compute RSA as the Spearman correlation between the neural RDM and the stage-specific semantic RDM on the relevant subset:

$$\mathrm{RSA}_\ell(M_s; \mathcal{S}) = \mathrm{Spearman}\left(\mathrm{vec}(D^\ell_\mathcal{S}), \mathrm{vec}(M_{s,\mathcal{S}})\right), \tag{7}$$

where $\mathcal{S}$ is the subset of reference scenes used for evaluation, $D^\ell_\mathcal{S}$ is the neural distance matrix restricted to $\mathcal{S}$, $M_{s,\mathcal{S}}$ is the corresponding stage-specific semantic RDM, and $\mathrm{vec}(\cdot)$ denotes the vectorized upper triangle of a matrix. We compute:

$$\rho^\mathrm{B}_\ell = \mathrm{RSA}_\ell(M_\mathrm{B}; \mathcal{D}), \tag{8}$$

$$\rho^{\mathrm{G}|\mathrm{B}}_\ell = \mathrm{RSA}_\ell(M_\mathrm{G}; \mathcal{D}_\mathrm{B}), \tag{9}$$

$$\rho^{\mathrm{A}|\mathrm{B}}_\ell = \mathrm{RSA}_\ell(M_\mathrm{A}; \mathcal{D}_\mathrm{B}). \tag{10}$$

Higher scores indicate stronger alignment between neural geometry and the semantic structure predicted by a stage. This conditional RDM design allows us to test the hypothesized hierarchy: feature binding separates the full reference pool by predicate satisfaction, while object grounding and answer matching yield representational substructures within the feature-binding set.

Setup. For each question type, we evaluate 72 queries sampled from CLEVR templates. To ensure the target object is unique, we restrict reference scenes to those containing 3-5 objects and exclude scenes in which the question-described object is not uniquely identifiable. For each query, we construct a reference pool of 500 distinct images excluding the query image. Layer-wise representations are extracted from the model and the no-CA baseline using the query question, with the CLS tokens as the layer-wise representation. The neural RDM is computed via pairwise correlation distance. Semantic RDMs are constructed from predicate-derived partitions, and RSA scores are obtained as the Spearman correlation between the neural RDM and the corresponding semantic RDM.

Results. Figure 4 shows stage-wise progressive organization across question types. In direct attribute queries, feature binding begins to emerge around L5, followed by object grounding and answer matching after L6; all three RSA curves then increase progressively across layers. Object grounding and answer matching track each other closely, consistent with the fact that the model queries the answer directly from the grounded object. In same-attribute relations, the RSA curves exhibit a handoff: anchor-aligned structure rises first, then declines as target-aligned structure emerges together with answer matching. This pattern suggests a semantic shift from representing the anchor object to resolving the target object. In spatial relations, by contrast, this handoff is absent: anchor-aligned RSA remains weak while target-aligned RSA increases. This suggests that the model may use spatial or attribute cues in the question to route representations directly toward the target, without forming an anchor-aligned structure.

4. Mechanistic Analysis

The representational analyses suggest that the proposed symbol processing mechanisms are reflected in predicate-aligned geometry across the visual stream. We next ask which information sources and model components support these mechanisms. We first inspect layer-wise cross-attention patterns to test whether different parts of the question are accessed at different stages of processing. We then use head-wise activation patching to identify components whose activations causally affect feature binding and its propagation to object grounding and answer matching.

4.1. Activation patching

We further investigate which components account for the mechanism through head-wise activation patching (Meng et al., 2022; Wang et al., 2023). We construct corrupted inputs by swapping keywords in the question: *attribute* (e.g., "red" → "blue") and *attribute query* (e.g., "what shape" → "what material") across four attributes. Patching restores the clean activation of a single head and measures recovery of the clean prediction.

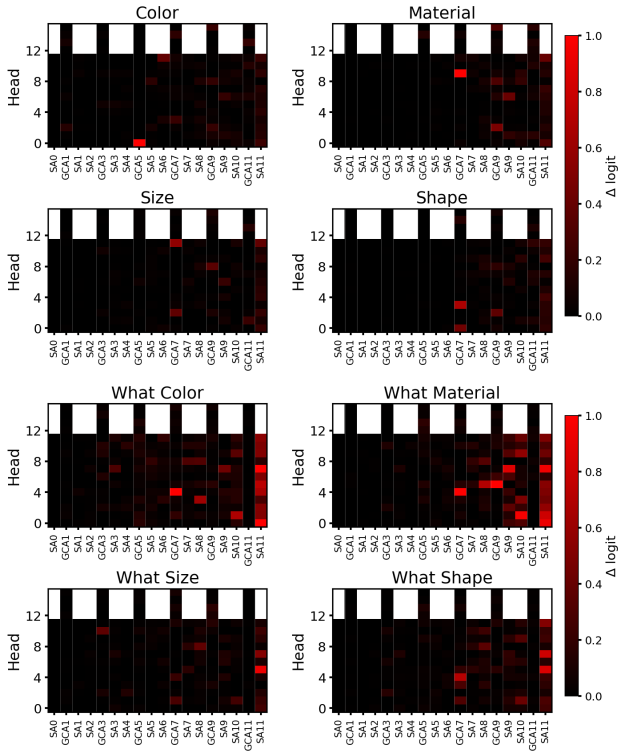

Figure 5. Activation patching under described attribute and query attribute corruptions. Described-attribute effects localize to CA heads in L5 and L7 (L5H0, L7H9, L7H11, L7H3), with heads specialized for color, material, size, and shape.

To identify the causal contribution of individual model components to text-conditioned visual reasoning, we employ activation patching in both denoising and noising directions. Given a query question q and its text-corrupted counterpart $\tilde{q}$ obtained by swapping attribute words or spatial terms, we perform two complementary interventions. In the denoising direction, a forward pass is performed with the corrupted input $\tilde{q}$ while restoring the activations of a target component from a clean forward pass with q. In the noising direction, we run a clean forward pass with q but replace the target head activation with the corresponding activation recorded from the corrupted-question forward pass $\tilde{q}$.

Specifically, for a CA head at layer ℓ with head index h, we record the pre-projection activations $\mathbf{z}^{(\ell,h)}$ at the output of the cross-attention module and substitute the corresponding slice during the counterfactual forward pass on the projection layer. We report the change in the ground-truth answer logit: $\Delta = \text{logit}_{\text{patched}}(a^*) - \text{logit}_{\text{baseline}}(a^*)$, averaged over corruption pairs sampled per attribute type.

As shown in Figure 5, the results reveal attribute-binding CA heads in the middle layers. Under described attribute

Figure 6. t-SNE results after noising the color-binding head CA L5H0 with a color-swapped question. Noising disrupts feature binding and propagates the error to downstream grounding and answer matching. Although final-layer answer partitions still emerge, the scenes become mixed and degraded. This suggests that downstream stages depend on intact upstream binding components and cannot fully recover from corrupted binding.

corruptions, patching individual heads restores the clean answer logit for specific attribute dimensions: color at L5H0 ($\Delta_{\text{logit}} = +2.83$), material at L7H9 ($\Delta_{\text{logit}} = +4.39$), size at L7H11 ($\Delta_{\text{logit}} = +3.03$), and shape at L7H3 ($\Delta_{\text{logit}} = +3.92$). These heads align with the binding stage identified by the representational analysis: corrupting a described attribute changes which object description the visual stream binds. On the other hand, under queried attribute corruptions, patching most heads in later self-attention, especially SA11, restores the clean answer logits. We therefore treat SA11 as a late answer-matching component in the following intervention analysis.

4.2. Causal Intervention

After the attribute-binding and answer-matching components are identified, we conduct causal interventions to probe their functional roles. We first present case study results by t-SNE visualization. A case study on color-binding head L5H0 is shown in Figure 6. Noising the head disrupts feature binding, where scenes containing the queried object no longer cluster at the grounding stage. The disruption propagates downstream. While later layers still attempt answer-matching separation, the degraded grounding produces mixed clusters rather than clean answer-aligned subgroups. This causal chain (color binding → object grounding → answer matching) confirms that the layer-wise hierarchy is functionally sequential, not merely correlated.

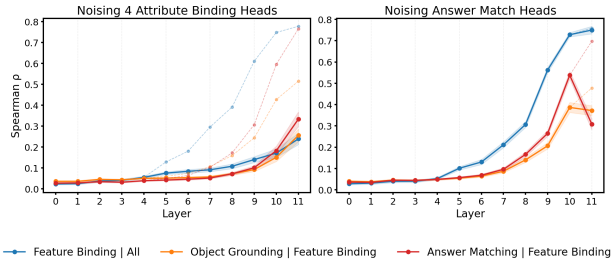

Figure 7. RSA after noising attribute-binding heads and SA11. Noising attribute-binding heads degrades binding structure, and the disruption propagates downstream. Later layers still attempt to form answer-matching partitions, but the resulting structure is corrupted. In contrast, SA11 late-stage intervention selectively disrupts answer-matching structure while preserving feature-binding structure.

We further conduct intervention RSA to test the causal chain on both binding heads and answer-matching components. As shown in Figure 7, noising the four attribute-binding heads (L5H0, L7H9, L7H11, L7H3) significantly reduces RSA on binding structures. Notably, at L11 the answer-matching RSA still rises despite the disrupted upstream geometry. The model attempts to partition by answer-equivalence even when grounding has collapsed, producing the mixed clusters observed in the patched t-SNE. In contrast, SA11 late-stage intervention selectively disrupts answer-matching structure while preserving feature-binding structure. This dissociation supports the causal chain, with each stage depending on upstream structure rather than operating independently.

5. Related work

5.1. Symbol processing mechanisms

Prior work has studied how symbol-like mechanisms emerge in neural networks across architectures and modalities. Early studies showed that architectural inductive biases and subnetwork structure can support emergent symbols (Webb et al., 2021; Lepori et al., 2023). More recent mechanistic analyses have localized specific operations, including entity binding (Feng & Steinhardt, 2024) and abstract reasoning (Yang et al., 2025). Within VLMs, related work has identified binding and retrieval (Assouel et al., 2025), in-context visual binding (Saxena et al., 2025), and grounding (Wu et al., 2025a; Kang et al., 2025). Our work asks whether such mechanisms also arise when language conditions the visual stream, shifting the analysis from language-side components to vision encoders.

5.2. Cross-Modal Integration in VLMs

Vision-language models differ in where they fuse modalities. Late-fusion architectures encode images into query-independent visual tokens and route most cross-modal computation through the language model (Liu et al., 2024; Wang et al., 2024). Early-fusion architectures allow one modality to condition the other during representation formation. FiLM applies language-conditioned affine transformations to visual modules (Perez et al., 2018). Flamingo injects visual information into a language model for few-shot generation (Alayrac et al., 2022); Q-former architectures use learned queries to extract visual features for language-image alignment (Li et al., 2022; 2023); SteerViT injects language into vision transformers to steer visual representations (Ruthardt et al., 2026). Cross-attention also supports language-to-vision control in image generation (Chefer et al., 2023) and grounded pre-training (Zhang et al., 2022). We adopt the language-to-vision early-fusion setting, but redirect the question from capability to mechanism rather than asking whether cross-attention improves performance.

5.3. Mechanistic Interpretability in VLMs

Mechanistic interpretability has developed mature tools for language models, including activation patching, causal tracing, and circuit discovery (Wang et al., 2023; Conmy et al., 2023; Meng et al., 2022; Olsson et al., 2022). Extending these tools to vision and vision-language models remains difficult because visual representations encode dense, continuous, and often ambiguous scene information (Liu et al., 2025). Recent work has analyzed visual concepts in image encoders (Joseph et al., 2025; Tong et al., 2025; Gandelsman et al., 2024) and semantic structure in cross-attention (Park et al., 2025). NOTICE uses semantically controlled image-text corruptions with activation patching to support causal interpretation (Golovanevsky et al., 2025). V-SEAM combines visual semantic editing with head-level modulation to measure causal effects across semantic categories (Wang et al., 2025). Our work introduces a predicate-based analysis framework for compositional visual reasoning, which provides semantic labels by formulating predicates.

6. Conclusion

We analyzed symbol-like processing in language-conditioned vision encoders by tracking how visual representations align with question-derived predicates across layers. Across layers, the model develops a three-stage organization: feature binding separates scenes that satisfy described attributes, object grounding clusters these scenes by object-level matches to the grounded description, and answer matching repartitions them into answer-aligned groups according to the queried attribute. These operations are analogous to mechanisms previously localized in language-model components (Assouel et al., 2025). In the visual stream, however, symbol-like processing appears as predicate-aligned geometry rather than position-based indexing over discrete tokens. Our results provide preliminary evidence linking model function to intermediate visual geometry, but they do not yet constitute a complete circuit-level account. Future work should use path patching and related causal interventions to identify the pathways connecting the components, and to test more directly whether predicate-aligned geometry mediates the model's final predictions.

Our analysis uses CLEVR because its scene graphs and programs permit exact predicate evaluation. This control also limits external validity due to the ambiguous nature of real-world images. We therefore interpret the results as evidence for a mechanism in a controlled setting, not as a claim that the same stages fully explain open-world VLM reasoning. Extending the analysis to natural images will require reliable scene-graph extraction, predicate estimators, or intervention methods that can define semantic conditions without ground-truth scene graphs.

Our findings suggest that reasoning-relevant structures and mechanisms may reside in different modalities in multimodal models. Fusion strategies can determine where such structure forms, and modality-specific substrates can instantiate different symbolic mechanisms. Future work should test whether the same geometric organization persists across architectures, fusion strategies, and real-world visual grounding and reasoning domains.

References

Alayrac, J.-B., Donahue, J., Luc, P., Miech, A., Barr, I., Hasson, Y., Lenc, K., Mensch, A., Millican, K., Reynolds, M., et al. Flamingo: a visual language model for few-shot learning. In *NeurIPS*, 2022.

Assouel, R., Campbell, D., Bengio, Y., and Webb, T. Visual symbolic mechanisms: Emergent symbol processing in vision language models. *arXiv preprint arXiv:2506.15871*, 2025.

Barsalou, L. W. Perceptual symbol systems. *Behavioral and brain sciences*, 22(4):577–660, 1999.

Biederman, I. Recognition-by-components: A theory of human image understanding. *Psychological Review*, 94 (2):115–147, 1987.

Chefer, H., Alaluf, Y., Vinker, Y., Wolf, L., and Cohen-Or, D. Attend-and-excite: Attention-based semantic guidance for text-to-image diffusion models. *ACM Transactions on Graphics*, 42(4):1–10, 2023. doi: 10.1145/3592116.

Conmy, A., Mavor-Parker, A. N., Lynch, A., Heimersheim, S., and Garriga-Alonso, A. Towards automated circuit discovery for mechanistic interpretability. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023.

Elhage, N., Nanda, N., Olsson, C., Henighan, T., Joseph, N., Mann, B., Askell, A., Bai, Y., Chen, A., Conerly, T., DasSarma, N., Drain, D., Ganguli, D., Hatfield-Dodds, Z., Hernandez, D., Jones, A., Kernion, J., Lovitt, L., Ndousse, K., Amodei, D., Brown, T., Clark, J., Kaplan, J., McCandlish, S., and Olah, C. A mathematical framework for transformer circuits. *Transformer Circuits Thread*, 2021. URL https://transformer-circuits.pub/2021/framework/index.html.

Feng, J. and Steinhardt, J. How do language models bind entities in context? In *International Conference on Learning Representations (ICLR)*, 2024.

Ferrando, J., Sarti, G., Bisazza, A., and Costa-jussà, M. R. A primer on the inner workings of transformer-based language models. *arXiv preprint arXiv:2405.00208*, 2024. URL https://arxiv.org/abs/2405.00208.

Fodor, J. A. and Pylyshyn, Z. W. Connectionism and cognitive architecture: A critical analysis. *Cognition*, 28(1–2): 3–71, 1988. doi: 10.1016/0010-0277(88)90031-5.

Gandelsman, Y., Efros, A. A., and Steinhardt, J. Interpreting CLIP's image representation via text-based decomposition. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=5Ca9sSzuDp.

Golovanevsky, M., Schoenegger, W., Sharma, A., Singh, R., Pavlick, E., and Sun, C. What do VLMs NOTICE? a mechanistic interpretability pipeline for gaussian-noise-free text-image corruption and evaluation. In *Proceedings of the 2025 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL)*, 2025. URL https://aclanthology.org/2025.naacl-long.571/.

Greff, K., van Steenkiste, S., and Schmidhuber, J. On the binding problem in artificial neural networks. *arXiv preprint arXiv:2012.05208*, 2020.

Griffiths, T. L., Lake, B. M., McCoy, R. T., Pavlick, E., and Webb, T. W. Whither symbols in the era of advanced neural networks? *Trends in Cognitive Sciences*, 2026. arXiv:2508.05776.

Johnson, J., Hariharan, B., van der Maaten, L., Fei-Fei, L., Lawrence Zitnick, C., and Girshick, R. CLEVR: A diagnostic dataset for compositional language and elementary visual reasoning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2901–2910, 2017.

Joseph, S., Nanda, N., et al. Prisma: An open source toolkit for mechanistic interpretability in vision and video. In *CVPR 2025 Workshop on Mechanistic Interpretability in Vision (MIV)*, 2025. Oral Presentation, arXiv:2504.19475.

Kang, S., Kim, J., Kim, J., and Hwang, S. J. Your large vision-language model only needs a few attention heads for visual grounding. *arXiv preprint arXiv:2503.06287*, 2025.

Ke, F., Hsu, J., Cai, Z., Ma, Z., Zheng, X., Wu, X., Huang, S., Wang, W., Haghighi, P. D., Haffari, G., Krishna, R., Wu, J., and Rezatofighi, H. Explain before you answer: A survey on compositional visual reasoning, 2025. URL https://arxiv.org/abs/2508.17298.

Lake, B. M., Ullman, T. D., Tenenbaum, J. B., and Gershman, S. J. Building machines that learn and think like people. *Behavioral and Brain Sciences*, 40:e253, 2017. doi: 10.1017/S0140525X16001837.

Lepori, M. A., Serre, T., and Pavlick, E. Break it down: Evidence for structural compositionality in neural networks. *Advances in Neural Information Processing Systems (NeurIPS)*, 2023.

Li, J., Li, D., Xiong, C., and Hoi, S. BLIP: Bootstrapping language-image pre-training for unified vision-language understanding and generation. In *International Conference on Machine Learning (ICML)*, 2022.

Li, J., Li, D., Savarese, S., and Hoi, S. BLIP-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. In *International Conference on Machine Learning (ICML)*, 2023.

Liu, H., Li, C., Li, Y., and Lee, Y. J. Improved baselines with visual instruction tuning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 26296–26306, 2024.

Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L., and Stoyanov, V. RoBERTa: A robustly optimized BERT pretraining approach. *arXiv preprint arXiv:1907.11692*, 2019.

Liu, Y., Zhang, Y., and Yeung-Levy, S. Mechanistic interpretability meets vision language models: Insights and limitations. ICLR 2025 Blogpost Track, 2025. URL https://openreview.net/forum?id=pZqvfsUpeh.

Marr, D. *Vision: A Computational Investigation into the Human Representation and Processing of Visual Information*. W. H. Freeman, 1982.

Meng, K., Bau, D., Andonian, A., and Belinkov, Y. Locating and editing factual associations in GPT. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2022.

Neo, C. et al. Towards interpreting visual information processing in vision-language models. In *International Conference on Learning Representations (ICLR)*, 2025.

Olsson, C., Elhage, N., Nanda, N., Joseph, N., DasSarma, N., Henighan, T., Mann, B., Askell, A., Bai, Y., Chen, A., et al. In-context learning and induction heads. *Transformer Circuits Thread*, 2022.

Oquab, M., Darcet, T., Moutakanni, T., Vo, H., Szafraniec, M., Khalidov, V., Fernandez, P., Haziza, D., Massa, F., El-Nouby, A., Assran, M., Ballas, N., Galuba, W., Howes, R., Huang, P.-Y., Li, S.-W., Misra, I., Rabbat, M., Sharma, V., Synnaeve, G., Xu, H., Jegou, H., Mairal, J., Labatut, P., Joulin, A., and Bojanowski, P. DINOv2: Learning robust visual features without supervision. *Transactions on Machine Learning Research (TMLR)*, 2024.

Park, J., Ko, J., Byun, D., Suh, J., and Rhee, W. Cross-attention head position patterns can align with human visual concepts in text-to-image generative models. In *International Conference on Learning Representations (ICLR)*, 2025.

Pavlick, E. Symbols and grounding in large language models. *Philosophical Transactions of the Royal Society A*, 381(2251), 2023.

Perez, E., Strub, F., de Vries, H., Dumoulin, V., and Courville, A. Film: visual reasoning with a general conditioning layer. In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence and Thirtieth Innovative Applications of Artificial Intelligence Conference and Eighth AAAI Symposium on Educational Advances in Artificial Intelligence*, AAAI'18/IAAI'18/EAAI'18. AAAI Press, 2018. ISBN 978-1-57735-800-8.

Ruthardt, J., Gaur, M., Ramanan, D., Tapaswi, M., and Asano, Y. M. Steerable visual representations, 2026. URL https://arxiv.org/abs/2604.02327.

Saxena, D. et al. Investigating mechanisms for in-context vision language binding. *arXiv preprint arXiv:2505.22200*, 2025.

Shepard, R. N. and Metzler, J. Mental rotation of three-dimensional objects. *Science*, 171(3972):701–703, 1971.

Tong, S. et al. Interpretable and testable vision features via sparse autoencoders. *arXiv preprint arXiv:2502.06755*, 2025.

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

Wang, K., Variengien, A., Conmy, A., Shlegeris, B., and Steinhardt, J. Interpretability in the wild: A circuit for indirect object identification in GPT-2 small. In *International Conference on Learning Representations (ICLR)*, 2023.

Wang, P., Bai, S., Tan, S., Wang, S., Fan, Z., Bai, J., Chen, K., Liu, X., Wang, J., Ge, W., Fan, Y., Dang, K., Du, M., Ren, X., Men, R., Liu, D., Zhou, C., Zhou, J., and Lin, J. Qwen2-VL: Enhancing vision-language model's perception of the world at any resolution. *arXiv preprint arXiv:2409.12191*, 2024.

Wang, Q., Hu, J., and Jiang, M. V-SEAM: Visual semantic editing and attention modulating for causal interpretability of vision-language models. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2025. URL https://aclanthology.org/2025.emnlp-main.880/.

Webb, T. W., Sinha, I., and Cohen, J. D. Emergent symbols through binding in external memory. In *International Conference on Learning Representations (ICLR)*, 2021.

Webb, T. W., Frankland, S. M., Altabaa, A., Segert, S., Krishnamurthy, K., Campbell, D., Russin, J., Giallanza, T., O'Reilly, R., Lafferty, J., et al. The relational bottleneck as an inductive bias for efficient abstraction. *Trends in Cognitive Sciences*, 2024.

Wu, S., Ma, Z., Luo, X., Huang, Y., Torres-Fonseca, J., Shi, F., and Chai, J. The mechanistic emergence of symbol grounding in language models. *arXiv preprint arXiv:2510.13796*, 2025a.

Wu, W., Wang, Y., Xiao, G., Peng, H., and Fu, Y. Retrieval head mechanistically explains long-context factuality. In *International Conference on Learning Representations (ICLR)*, 2025b.

Yang, Y., Campbell, D., Huang, K., Wang, M., Cohen, J., and Webb, T. Emergent symbolic mechanisms support abstract reasoning in large language models. *arXiv preprint arXiv:2502.20332*, 2025.

Zhang, H., Zhang, P., Hu, X., Chen, Y.-C., Li, L. H., Dai, X., Wang, L., Yuan, L., Hwang, J.-N., and Gao, J. Glipv2: Unifying localization and vision-language understanding. *arXiv preprint arXiv:2206.05836*, 2022.

A. Implementation Details

General training setup. All experiments use a frozen ViT backbone and a frozen RoBERTa-large text encoder. The model is trained for 16 epochs with AdamW, learning rate 1×10^{-4}, weight decay 0.05, gradient clipping at 1.0, and a 2-epoch warmup. We use bfloat16 mixed precision. Results are averaged over three random seeds: 42, 43, and 44.

Readout methods. For the main model, we use a one-layer self-attention decoder. The decoder input concatenates visual patch tokens, question tokens, and a BOS token. The vocabulary includes 28 CLEVR answer classes. The decoder uses hidden dimension $d_{\text{model}} = 512$, 8 attention heads, one transformer layer, and a maximum output length of 8.

Backbone comparison. For the comparison in Table 1, we keep the language-conditioned architecture fixed and vary only the pretrained ViT backbone. The DINOv2 model uses a ViT-B/14 backbone at resolution 336, yielding 576 visual patches. SigLIP uses a ViT-B/16 backbone at resolution 256, yielding 256 patches; because this backbone does not use a CLS token in our setup, we use mean pooling for its visual representation. The ImageNet21K-supervised ViT uses a ViT-B/16 backbone at resolution 384, yielding 576 patches. The MAE backbone uses a ViT-B/16 model at resolution 224, yielding 196 patches.

DINOv2 ablation settings. For the component ablations in Table 2, the baseline model uses the frozen pretrained DINOv2 backbone, trainable CA modules, the frozen RoBERTa-large text encoder with a trainable connector, and the one-layer concat self-attention decoder. In the classifier ablation, we replace the decoder with an MLP classification head. In the from-scratch ablation, the ViT backbone is randomly initialized and trained instead of using frozen pretrained weights. In the no-CA ablation, all CA modules are removed, so the visual encoder is not conditioned on the question. In the learned-text ablation, RoBERTa-large is replaced with a learned text encoder trained from scratch.

B. Full Evaluation Results

This section reports the full category-wise evaluation results for the backbone comparison and component ablations in Tables 3, 4, and 5. The main text reports overall accuracy and query-attribute accuracy for compactness; here we additionally include equivalent-attribute, existence, counting, and integer-comparison question categories.

Table 3. Full evaluation accuracy for the backbone comparison using the self-attention decoder. All models use language-to-vision CA and the same decoder design.

Backbone	Overall	QryAttr	EqAttr	Exist	Count	CmpInt
DINOv2	92.4	99.1	92.4	96.0	85.3	78.6
SigLIP	92.6	99.0	92.0	96.4	86.3	78.5
Supervised-ViT	86.6	94.0	83.9	91.4	79.2	74.2
MAE	74.8	92.1	58.5	77.6	60.3	71.9

Table 4. Full evaluation accuracy for the backbone comparison using a classifier readout. Compared with the concat self-attention decoder, the classifier readout generally reduces performance, especially for SigLIP.

Backbone	Overall	QryAttr	EqAttr	Exist	Count	CmpInt
DINOv2	90.1	98.1	88.7	94.2	82.4	75.7
SigLIP	84.8	92.8	82.6	89.8	76.2	72.2
Supervised-ViT	86.6	92.5	87.6	91.8	79.3	73.0
MAE	77.0	89.9	60.6	80.1	59.8	68.2

Table 5. Full evaluation accuracy for component ablations. The baseline uses a frozen pretrained DINOv2 backbone, language-to-vision CA, a frozen RoBERTa text encoder, and a self-attention decoder.

Ablation	Overall	QryAttr	EqAttr	Exist	Count	CmpInt
DINOv2 + decoder	92.4	99.1	92.4	96.0	85.3	78.6
- Decoder + classifier	90.1	98.1	88.7	94.2	82.4	75.7
ViT backbone from scratch	52.8	49.1	51.7	66.4	45.9	67.2
- CA (no language conditioning)	49.4	51.9	52.1	57.1	39.1	50.0
- RoBERTa + learned text encoder	24.6	2.7	48.8	49.3	15.8	48.8

C. More results on CLEVR Series

We evaluate whether the CLEVR-trained model (*i.e.*, DINOv2+decoder) transfers beyond the original validation distribution on a series of CLEVR variants. We consider two settings: zero-shot transfer, where the model is evaluated directly on a target benchmark without additional training, and fine-tuning transfer, where the model is adapted to the target dataset. These benchmarks test different forms of generalization, including human-written question paraphrases in CLEVR-Humans, novel compositional structures in CLOSURE, and arithmetic reasoning in CLEVR-Math. The overall results are summarized in Table 6.

Table 6. Overall zero-shot and fine-tuned accuracy on compositional generalization and transfer benchmarks. Zero-shot results evaluate the CLEVR-trained model without downstream fine-tuning.

Dataset	Zero-shot	Fine-tuned
CLEVR-Humans	54.5	73.8
CLEVR-Math	10.5	94.1
CLOSURE	57.9	70.3