SPICE: Simple Polysemantic Feature Interpretation via Clustering-based Explanation

Sehyun Lee[1], Dahee Kwon[1], Damin Lee[1], and Jaesik Choi[1,2]

[1] Korea Advanced Institute of Science and Technology (KAIST), Daejeon, Republic of Korea
{sehyun.lee, daheekwon, jaesik.choi}@kaist.ac.kr, xxdamin@gmail.com
[2] INEEJI, Seongnam, Republic of Korea

Abstract. One of the pivotal recent challenges in neural network interpretability is polysemanticity, where a single neuron is activated by multiple, often unrelated concepts, hindering clear functional understanding. Although prior work has explored this phenomenon, existing approaches remain architecture-specific and depend on manual heuristics such as a fixed number of concept clusters (K), limiting their generality and scalability—especially for modern Transformer-based models. To address these limitations, we introduce SPICE (**S**imple **P**olysemantic Feature **I**nterpretation via **C**lustering-based **E**xplanation), a generalizable framework for analyzing polysemanticity in deep vision architectures. SPICE avoids architecture-dependent propagation rules, enabling the first systematic comparison of polysemanticity across both CNNs and Transformers, and automatically determines the number of concept clusters per neuron, eliminating reliance on a preset K and supporting scalable analysis for large models. Using SPICE, we conduct a comprehensive investigation into how polysemanticity emerges, varies across depth and architecture, and forms through distinct computational pathways.

Keywords: Interpretability · Visual Concept · Polysemanticity

1 Introduction

Deep learning models have achieved remarkable performance across a wide range of applications, but their rapidly increasing complexity and scale have made it ever more difficult to understand their underlying mechanisms [2]. As a result, interpretability research—which seeks to clarify the internal representations and computational processes learned by models—has become increasingly important and is now an active area of study. In particular, mechanistic interpretability, which examines the functions of neurons and modules in storing and processing knowledge, has emerged as one of the most prominent research directions in the field [5, 11, 24, 25].

In the vision domain, such research has been actively pursued, with approaches such as those in [4, 23] analyzing models at the level of individual neurons by identifying shared visual features among the samples that strongly

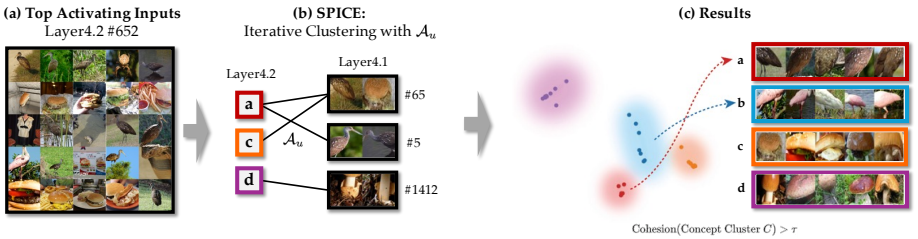


Fig. 1: An overview of the proposed method (SPICE). (a) A polysemantic neuron exhibits mixed selectivity across diverse image inputs. **(b)** SPICE traces the computational pathways of these activations by computing attribution footprints ($\mathcal{A}_u$) from the preceding layer. **(c)** We iteratively cluster these footprints to disentangle the neuron's representations. By strictly separating clusters that satisfy the condition Cohesion(C) $> \tau$, SPICE successfully decomposes the entangled inputs into highly coherent, interpretable conceptual groups.

activate them. This approach rests on the implicit assumption that each neuron corresponds to a single function. Recent studies, however, have shown that multiple concepts often coexist within a single neuron—a phenomenon known as ***polysemanticity*** [8, 9]. As a result, single-neuron analysis faces fundamental limitations in faithfulness, and such interpretations alone are insufficient to fully explain the underlying mechanisms of deep models.

To account for polysemanticity, several approaches have been proposed. While training interpretable Sparse Autoencoders (SAEs) offers one promising path, such methods do not directly reveal the intrinsic structural flow of the network [7, 32]. To address this, attribution clustering has emerged as a key technique for operationalizing circuit-based tracing by grouping a neuron's attribution footprints to separate its encoded concepts. However, existing attribution clustering methods suffer from two critical limitations. First, they exhibit a fundamental lack of generality. To date, these analyses have been almost exclusively confined to CNNs, failing to provide a unified framework applicable to modern Transformer-based architectures. Second, they lack scalability due to their reliance on manual heuristics. Specifically, these approaches often require predefining a fixed number of concept clusters (K) for each neuron-a subjective and laborious process that is infeasible for today's large-scale models. Consequently, developing a generalizable and scalable method to analyze polysemanticity across diverse architectures remains a key unsolved problem.

Therefore, we introduce SPICE (**S**imple **P**olysemantic Feature **I**nterpretation via **C**lustering-based **E**xplanation), a method that disentangles polysemantic neurons with an emphasis on generality and scalability. SPICE is both simple and effective, addressing two major limitations of prior work. (1) We introduce a scalable and flexible attribution-clustering framework for dissecting polysemantic neurons. By adaptively determining the optimal number of concept clusters (K), SPICE eliminates the reliance on manual heuristics. Furthermore, extensive

evaluations and ablations demonstrate that our method achieves highly robust disentanglement performance across diverse architectures, extreme activation ranges, and various attribution methods. (2) Using this framework, we provide an in-depth analysis of polysemanticity to explore the representational differences across models. Our findings highlight distinct inductive biases between CNNs and Transformers, and illustrate the diverse computational pathways that underlie concept formation.

Using the proposed method, we conduct an in-depth investigation into the formation patterns of polysemanticity and their internal structure. Beyond extending interpretability, this analysis offers novel insights into the fundamental principles by which models organize and represent concepts.

2 Related Work

Mechanistic Interpretability A significant branch of interpretability research focuses on identifying human-understandable concepts and mechanisms within neural networks [5]. This line of work seeks to move beyond input–output correlations and explain how a model computes internally. Early studies focused on associating individual neurons with specific visual concepts [4, 24, 25]. Since then, research has broadened to include methods for automatically discovering concepts with limited human supervision [15,23], as well as approaches that analyze circuits—pathways of functionally related neurons—to uncover higher-level computational structures [6,17,18,33]. Our work builds on this tradition, aiming to provide a scalable and systematic analysis of a neuron's internal conceptual structure.

Polysemanticity as a Challenge Early interpretability methods often relied on an implicit monosemantic assumption—that each neuron corresponds to a single coherent concept. However, polysemanticity is essentially a fundamental property of deep networks, arising from data statistics and superposition constraints [9, 21]. To account for this, Sparse Autoencoders (SAEs) are used to disentangle compressed features [7, 12, 32], though they do not directly reveal the model's intrinsic attribution flow. Furthermore, while several works leverage Vision-Language Models (VLMs) to probe polysemantic concepts [1, 22, 34, 36], these approaches are fundamentally bottlenecked by representation misalignment and the sensitivity to the chosen external model.

Disentanglement via Attribution Approach To directly analyze the internal structure of polysemantic neurons, several approaches leverage attribution clustering—grouping a neuron's attribution footprints to separate its encoded concepts [8, 14, 34]. These methods are powerful, but the literature exhibits two critical limitations that our work addresses. First, a lack of generality across architectures. To date, these analyses [8, 14] have been almost exclusively confined to CNN architectures. A unified framework to systematically analyze and compare polysemanticity in modern Transformer-based models has been notably

absent. Second, prior work lacks scalability in concept discovery, largely due to reliance on manual or restrictive heuristics. Pioneering work like PURE [8] requires manually specifying the number of concepts (K) for each neuron, while methods such as [14] perform explicit disentanglement but are constrained to separating only a small, fixed number of components rather than discovering an arbitrary, data-driven set of clusters. Both approaches are subjective, time-consuming, and practically infeasible for large-scale analysis. Therefore, a framework that is both generalizable and scalable for attribution-based analysis has remained a key unsolved problem, which we address in this paper.

3 Method

The visual concept captured by a single neuron is often inferred from the shared features of the samples that most strongly activate it [4, 15, 23]. While this approach can reveal what a neuron has learned, it becomes challenging to disentangle multiple concepts when a neuron is polysemantic—i.e., when several distinct concepts coexist within the same unit. To analyze this polysemanticity within a model's learned visual representations, our goal is therefore to cluster the multiple visual concepts encoded by a single neuron. To this end, we introduce **S**imple **P**olysemantic Feature **I**nterpretation via **C**lustering-based **E**xplanation (SPICE).

Identifying Representational Footprints via Attribution Information in deep neural networks is widely understood to be formed hierarchically, beginning at the input and propagating through successive layers until it culminates in the final decision [19, 24, 35]. Consequently, understanding an internal visual concept requires tracing its origins along this information flow. Within this framework, we seek to explain the information encoded by each neuron by defining its representational footprint-a trace of its compositional origins. Building upon this premise, we assume that concepts sharing similar footprints are treated as 'related' by the model. To realize this empirically, we compute these footprints using attribution methods, which are widely employed for analyzing deep vision models [27–29].

Formally, given a dataset $\mathcal{D} = \{x_i\}$ and a network f, we first interpret the concept represented by a neuron u^l in layer l through its set of most highly activating samples, $\mathcal{D}_{u^l} \subset \mathcal{D}$. We define a *neuron* according to the network architecture. For convolutional models, each channel is treated as a neuron, as it acts as a distinct feature detector across spatial locations.[3] For transformer-based models, each hidden dimension is treated as a neuron, as it represents a distinct feature across all input patches.

To trace the origin of the visual concept represented by a single neuron u^l, we quantify how each predecessor neuron in layer $l-1$ contributes to its activation. Since a neuron corresponds to an entire channel in CNNs or a hidden

[3] The term "neuron" is often used interchangeably with "feature" in the literature, or "channel" in the context of CNNs.

dimension in Transformers, its activation is distributed over multiple output positions, i.e., a spatial feature map in CNNs or a sequence of patch activations in Transformers. We therefore aggregate the positional activations to obtain a scalar neuron activation,

$$\bar{a}_u(x) := \begin{cases} \sum_{h,w} a_u(x)_{h,w}, & \text{(CNN)}, \\ \sum_p a_u(x)_p, & \text{(Transformer)}. \end{cases} \tag{1}$$

For a given sample x_i, the aggregated activations of all predecessor neurons form the activation vector

$$\bar{\mathbf{a}}^{\,l-1}(x_i) = \left[\bar{a}_{u_1^{l-1}}(x_i), \ldots, \bar{a}_{u_{d_{l-1}}^{l-1}}(x_i) \right] \in \mathbb{R}^{d_{l-1}}. \tag{2}$$

Using the aggregated activation of the target neuron as the scalar objective, we compute the Input×Gradient attribution from all predecessor neurons to target neuron,

$$\boldsymbol{\phi}(x_i) = \bar{\mathbf{a}}^{\,l-1}(x_i) \odot \frac{\partial \bar{a}_{u^l}(x_i)}{\partial \bar{\mathbf{a}}^{\,l-1}(x_i)} \in \mathbb{R}^{d_{l-1}}, \tag{3}$$

where $\odot$ denotes element-wise multiplication. The j-th entry of $\boldsymbol{\phi}(x_i)$ represents the attribution from predecessor neuron u_j^{l-1} to the target neuron u^l.

Stacking the attribution vectors of all highly activating samples $x_i \in \mathcal{D}_{u^l}$ yields the attribution footprint matrix,

$$\mathcal{A}_{u^l} = \begin{bmatrix} \boldsymbol{\phi}(x_1)^\top \\ \boldsymbol{\phi}(x_2)^\top \\ \vdots \\ \boldsymbol{\phi}(x_{|\mathcal{D}_{u^l}|})^\top \end{bmatrix} \in \mathbb{R}^{|\mathcal{D}_{u^l}| \times d_{l-1}}. \tag{4}$$

Clustering the rows of $\mathcal{A}_{u^l}$ therefore groups samples with similar predecessor-attribution patterns, revealing distinct concepts that drive the activation of u^l.

As model depth increases, the attribution vectors of different highly activating samples of the same neuron (i.e., the rows of $\mathcal{A}_u$) become substantially more similar to one another. We hypothesize that this effect stems from the increasing anisotropy of the feature space, a phenomenon widely reported in prior work on large models [3,10,13,26]. To mitigate its impact and enable effective attribution footprint-based clustering, we normalize attribution matrices along the neuron dimension before clustering. Rather than simply placing dimensions on a comparable scale, this normalization actively mitigates the anisotropy problem, allowing the clustering algorithm to focus on relative semantic patterns rather than absolute magnitude biases. With this normalization step, our method achieves more reliable concept disentanglement.

Disentangling Concepts by Clustering Footprints Having defined the representational footprints, we now introduce our method, namely SPICE. Rather than assuming a fixed number of underlying concepts, we propose an iterative algorithm that progressively identifies and separates cohesive groups of footprints based on their pairwise cosine similarity. As detailed in Algorithm 1, the process begins by attempting to partition the rows of the footprint matrix $\mathcal{A}_u$ into $k = 2$ clusters. A cluster $\mathcal{C}$ is considered a cohesive concept if its internal coherence, which we define as the average pairwise cosine similarity between its members, exceeds an adaptive threshold τ. Cohesive concepts are accepted as final and removed from the working set $\mathcal{A}'$. The algorithm then dynamically adjusts the number of clusters k and repeats this process on the remaining attributions until no further cohesive concepts can be disentangled.

A key component is the dynamic calculation of this cohesion threshold τ. A fixed, universal threshold would fail to adapt to the diverse similarity distributions exhibited by the footprints of different neurons. We therefore compute τ adaptively from the set of footprints currently under analysis, $\mathcal{A}'$. Our empirical analysis reveals that the character of the similarity distribution systematically changes with network depth, as illustrated in Figure 7(right). Early layers typically exhibit a unimodal, long-tailed distribution of similarities. In contrast, later layers often display a bimodal distribution, where the emergence of a second peak suggests the formation of distinct, highly cohesive conceptual groups.

To create a robust criterion that accommodates both scenarios, we define the threshold τ as the maximum of two candidates: (1) the 95th percentile of the pairwise similarity values, and (2) the value corresponding to the second peak of a Kernel Density Estimate (KDE) fitted to the similarity distribution. Formally, letting S denote the set of pairwise cosine similarities of the current footprints $\mathcal{A}'$,

$$\tau = \max\big(Q_{0.95}(S),\, p_2(S)\big), \tag{5}$$

where $Q_{0.95}(S)$ is the 95th percentile of S and $p_2(S)$ is the second local maximum (in order of increasing similarity) of the Gaussian KDE of S whose PDF value exceeds 0.1; when the distribution is unimodal and $p_2(S)$ does not exist, τ falls back to $Q_{0.95}(S)$. The percentile provides a robust baseline, particularly for the unimodal distributions in early layers, while the second peak offers a more semantically meaningful partition for the well-formed concepts in later layers. This dual-criterion approach ensures that our definition of a concept is contextually grounded across the network's hierarchy.

4 Experiment

In this section, we present both the qualitative and quantitative evaluation of our proposed method, along with a comprehensive analysis.

4.1 Quantitative Comparison with Baselines

To demonstrate the effectiveness of our method in disentangling and explaining multiple concepts encoded within a single neuron, we extensively evaluate

Algorithm 1 SPICE

Input: A set of attribution footprints $\mathcal{A}_u$, a cohesion threshold τ.
Output: A set of disjoint concepts $\mathcal{C}^*$, where each concept is a set of footprints.

procedure DisentangleConcepts($\mathcal{A}_u, \tau$)
  ▷ *Initialize working variables*
  $\mathcal{A}' \leftarrow \mathcal{A}_u$
  $\mathcal{C}^* \leftarrow \emptyset$
  $k \leftarrow 2$
  while $|\mathcal{A}'| \geq k$ **do**
    ▷ *Search for cohesive clusters of size k*
    Let $\mathcal{C}_{cohesive}$ be the set of all clusters $C \in \text{Cluster}(\mathcal{A}', k)$ where $\text{Cohesion}(C) > \tau$.
    if $\mathcal{C}_{cohesive} \neq \emptyset$ **then**
      ▷ *Found cohesive clusters: update*
      $\mathcal{C}^* \leftarrow \mathcal{C}^* \cup \mathcal{C}_{cohesive}$
      $\mathcal{A}' \leftarrow \mathcal{A}' \setminus \bigcup_{C \in \mathcal{C}_{cohesive}} C$
      $k \leftarrow \max(2, k - (|\mathcal{C}_{cohesive}| - 1))$
    else
      ▷ *No cohesive cluster: increase search size*
      $k \leftarrow k + 1$
    end if
  end while
  ▷ *Treat remaining footprints as individual concepts*
  $\mathcal{C}^* \leftarrow \mathcal{C}^* \cup \{\{a\} \mid a \in \mathcal{A}'\}$
  return $\mathcal{C}^*$
end procedure

where $\text{Cohesion}(C) = \text{mean}_{a_i, a_j \in C}(\text{sim}(a_i, a_j))$

SPICE across four diverse architectures (ResNet-50, ViT-B, DenseNet, and CLIP ViT-B). We compare our approach against several recent baselines, including PURE [8], CPE [34], and LE [22]. While PURE was originally restricted to CNNs due to its reliance on CRP grammar, we extended and adapted its implementation for Transformer architectures to ensure a comprehensive evaluation. Moreover, unlike SPICE-which adaptively determines the optimal number of clusters per neuron-PURE requires the number of clusters to be specified in advance. Accordingly, we configure it by selecting the optimal number of clusters using the silhouette score.

Separability based on External Semantic Space. To quantitatively evaluate the effectiveness of our concept disentanglement, we first adopt *separability* as our primary metric, defined as the ratio of intra-cluster similarity to inter-cluster similarity. Following the established protocol of PURE, we utilize CLIP as the external verification tool to measure these similarities, grounded in the premise that visually similar images cluster closely in the hidden space. Con-

Table 1: Separability comparison of baselines versus **Ours** across models and layers. As PURE was originally designed for CNNs, we adapted its CRP implementation for Transformer models for this comparison.

Model / Layer	PURE*	LE	CPE	Ours
ResNet-50 L2.1	$1.018_{\pm 0.031}$	$\mathbf{1.112}_{\pm 0.122}$	$0.996_{\pm 0.014}$	$1.092_{\pm 0.056}$
ResNet-50 L3.3	$0.999_{\pm 0.012}$	$1.108_{\pm 0.153}$	$0.995_{\pm 0.007}$	$\mathbf{1.160}_{\pm 0.081}$
ResNet-50 L4.2	$1.009_{\pm 0.071}$	$1.187_{\pm 0.221}$	$0.998_{\pm 0.011}$	$\mathbf{1.291}_{\pm 0.224}$
ViT-B B2	$1.002_{\pm 0.010}$	$\mathbf{1.199}_{\pm 0.181}$	$0.994_{\pm 0.009}$	$1.137_{\pm 0.062}$
ViT-B B6	$1.012_{\pm 0.013}$	$1.185_{\pm 0.153}$	$0.994_{\pm 0.009}$	$\mathbf{1.240}_{\pm 0.089}$
ViT-B B11	$1.035_{\pm 0.025}$	$1.357_{\pm 0.181}$	$0.998_{\pm 0.018}$	$\mathbf{1.577}_{\pm 0.089}$
DenseNet L2.1	$1.012_{\pm 0.028}$	$\mathbf{1.090}_{\pm 0.158}$	$0.994_{\pm 0.007}$	$1.071_{\pm 0.059}$
DenseNet L3.12	$1.053_{\pm 0.054}$	$1.062_{\pm 0.139}$	$0.994_{\pm 0.005}$	$\mathbf{1.114}_{\pm 0.060}$
DenseNet L4.16	$1.025_{\pm 0.035}$	$\mathbf{1.191}_{\pm 0.116}$	$0.994_{\pm 0.009}$	$1.033_{\pm 0.032}$
CLIP ViT-B B2	$1.007_{\pm 0.018}$	$1.088_{\pm 0.082}$	$0.994_{\pm 0.008}$	$\mathbf{1.090}_{\pm 0.051}$
CLIP ViT-B B6	$1.008_{\pm 0.016}$	$1.165_{\pm 0.148}$	$0.997_{\pm 0.009}$	$\mathbf{1.212}_{\pm 0.090}$
CLIP ViT-B B11	$1.021_{\pm 0.040}$	$1.244_{\pm 0.192}$	$0.998_{\pm 0.013}$	$\mathbf{1.410}_{\pm 0.102}$

cretely, for each neuron n with discovered clusters, we compute $\bar{s}^{(n)}_{\text{intra}}$ as the mean CLIP cosine similarity over all within-cluster image pairs and $\bar{s}^{(n)}_{\text{inter}}$ as the mean over *all* cross-cluster pairs (i.e., the full $i \times j$ pairwise comparisons, not centroid-based), and report a single dataset-level ratio of means,

$$\text{Sep.} = \frac{\frac{1}{N}\sum_{n=1}^{N} \bar{s}^{(n)}_{\text{intra}}}{\frac{1}{N}\sum_{n=1}^{N} \bar{s}^{(n)}_{\text{inter}}}. \tag{6}$$

We use the ImageNet validation set as the probing dataset for all separability evaluations. A higher separability score indicates a more successful and distinct concept separation.

As shown in Table 1, SPICE exhibits dominant separability scores across various networks and depths. While text-prior-based methods like LE show competitive performance in early layers (e.g., B2), their performance notably degrades in deeper layers (e.g., L4.2, B11) where concepts become highly abstract and polysemantic. In contrast, SPICE maintains robust disentanglement, significantly outperforming all baselines in the deep layers. The slight deviation in early layers occurs because they typically capture low-level features (e.g., textures, colors) that are less aligned with CLIP's high-level semantic space. We note that CPE consistently yields near-unit separability (≈ 0.99); this stems from an evaluation mismatch rather than a low cluster count: CPE groups samples by VLM-generated text concepts whereas separability is measured in CLIP image-embedding space, and under the CPE protocol each sample is linked to roughly three concepts per neuron so the same images recur across groups, pulling inter-cluster similarity toward intra-cluster similarity. This is further evidenced by PURE, which uses an even smaller fixed $K = 2$ yet attains a higher score (e.g.,

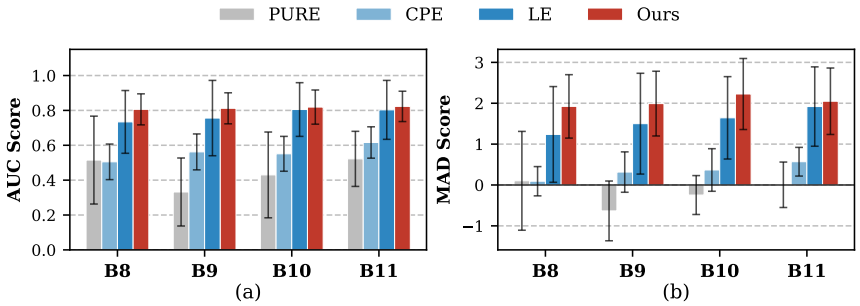


Fig. 2: Simulation-based Semantic Evaluation in ViT-B

1.035 at ViT-B B11), ruling out cluster count as the cause. This phenomenon is visually supported in Figure 3.

Simulation-based Evaluation. To further rigorously validate the semantic faithfulness of the discovered concepts beyond embedding-based metrics, we conduct a generative evaluation using the CoSy protocol [16]. Specifically, we generate one textual description per discovered cluster with GPT-4.1 [1] and synthesize verification images via SDXL, following the CoSy protocol. AUC (Alignment Under Curve) and MAD (Mean Absolute Difference) are computed per cluster and then averaged, so that the resulting score is independent of the number of clusters K each method assigns to a neuron.

As illustrated in Figure 2, SPICE consistently achieves the highest AUC and MAD scores across multiple blocks of ViT-B compared to PURE, CPE, and LE. It is worth noting that while inherent text-visual misalignment exists in deep vision models, SPICE achieves higher validity because it intrinsically discovers concept clusters from internal activations independent of text. By doing so, it effectively bypasses the bottleneck of text-defined priors that fundamentally limits baselines like LE and CPE, ensuring a more faithful recovery of the model's actual learned concepts.

4.2 Qualitative Comparison with Baselines

To visually demonstrate the effectiveness of our method in disentangling and explaining multiple concepts encoded within a single neuron, we perform a qualitative comparison against the aforementioned baselines (PURE [8], CPE [34], and LE [22]).

In Figure 3, each row illustrates representative images belonging to a single concept cluster discovered by each method. Across both ResNet-50 and ViT-B/16, SPICE discovers the most comprehensive set of polysemantic clusters, indicating its strong capability to fully recover the diverse concepts encoded within a neuron. Moreover, while several baselines exhibit notable visual inconsistency

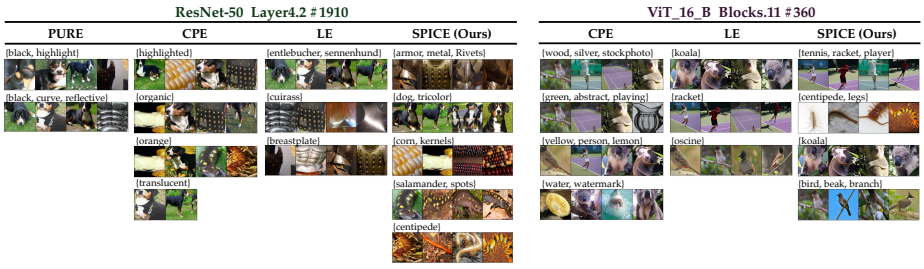


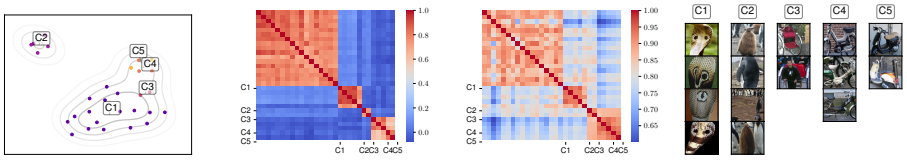
Fig. 3: Qualitative comparisons with various baselines. Each row shows four exemplar images representing the visual characteristics of a discovered cluster, accompanied by its corresponding textual description.

and noise within their clusters, SPICE consistently yields semantically aligned and coherent groupings. Similar to LE and CPE, SPICE naturally integrates with vision-language models [1] to provide accurate textual descriptions for each cluster. Taken together, these qualitative results demonstrate that SPICE offers the most reliable visual characterization of neuron-level polysemanticity.

4.3 In-depth Analysis

Using our framework, we conduct a comprehensive analysis of polysemanticity. We first visually dissect a single neuron, and then trace the computational pathways that give rise to these diverse concepts. We then broaden our scope to explore architectural inductive biases across models, and finally validate the robustness of our findings through rigorous ablation studies.

Dissecting a Polysemantic Neuron. To understand how polysemanticity physically manifests, we conduct a detailed case study of Neuron #652 from



(a) **(b)** **(c)** **(d)**

Fig. 4: Dissecting a polysemantic neuron (#652) from ViT-B block 11. **(a)** UMAP embedding reveals distinct conceptual clusters within the neuron's representation footprints. **(b)** A similarity heatmap quantifies the low inter-cluster and high intra-cluster similarity. **(c)** A CLIP similarity heatmap quantitatively shows that concepts are semantically disparate. **(d)** Exemplar images confirm the diverse and disparate visual concepts represented by each cluster; such as snakeskin (C1), penguins (C2), and various two-wheeled vehicles (C3-C5).

the last block of ViT-B. As illustrated in Figure 4, we decompose its activations into distinct conceptual clusters. The UMAP projection (Figure 4a) and exemplar images (Figure 4d) reveal clear semantic heterogeneity, with the neuron responding to entirely unrelated concepts such as snakeskin textures (C1), penguins (C2), and two-wheeled vehicles (C3-C5). The heatmaps (Figure 4b, c) further validate that these clusters are structurally well-separated within the model's internal space and possess low semantic overlap, confirming the precise disentanglement by SPICE.

Formation Mechanisms and Concept Pathways Having observed that a single neuron can perfectly disentangle into distinct concepts, we next investigate *how* these specific concepts are formed. By measuring label co-occurrences in the final ViT-B block (Figure 5), we find that polysemantic neurons arise heterogeneously: some encode highly related concepts (e.g., variants of telephones in neuron #24), while others combine entirely unrelated categories (e.g., burritos and baseballs in neuron #157).

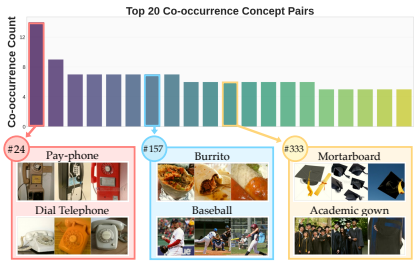


Fig. 5: Top 20 co-occurrence concept pairs in ViT-B.

To determine if these distinct forms of polysemanticity stem from different mechanisms, we trace their upstream computational pathways (Figure 6). For neuron #13, which encodes unrelated concepts, we observe largely disjoint upstream connections, with only an 18% overlap among the top-50 contributing neurons. Conversely, neuron #371, encoding related concepts, shares 42% of its attribution pathways, indicating a shared formation circuit. These human-aligned interpretations are further validated by VLM-generated descriptions [1, 34]. Ultimately, these findings reveal that polysemantic neurons are formed through diverse processes—ranging from overlapping semantic circuits to entirely disjoint pathways—which fundamentally dictate their interpretability.

Architectural Inductive Bias. Building on these neuron-level mechanisms, we broaden our analysis to examine how macroscopic architectural designs influence polysemantic behavior across the entire network (Figure 7). First, regarding the number of disentangled concepts (left), CNN-based models (ResNet, ConvNeXt) tend to maintain numerous concepts through most of the network, with ResNet-50 in particular showing a pronounced drop in the final blocks, whereas Vision Transformers (ViT) exhibit a more gradual decrease. Second, assessing internal cohesion via intra-cluster similarity (center), CNNs exhibit consistent stability interrupted only by spatial downsampling transitions. Conversely, Transformers display a characteristic U-shaped trend, exploring higher conceptual diversity in middle layers before converging.

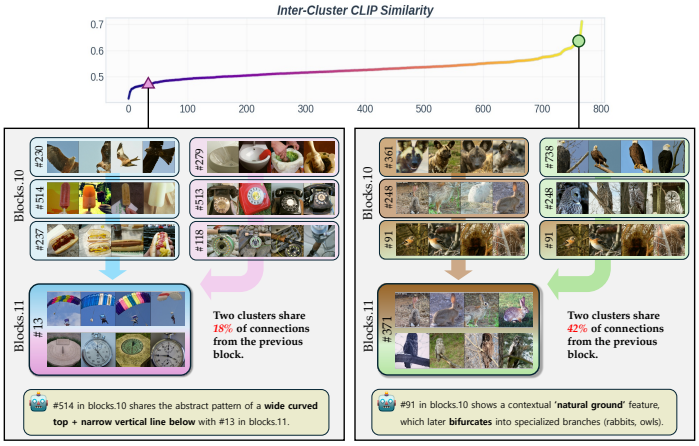


Fig. 6: (**Left**) Inter-cluster CLIP similarity, where the x-axis denotes neurons in ViT-B block 11 sorted by similarity. (**Middle**, **Right**) Example of a polysemantic neuron with low inter-cluster similarity, and a high inter-cluster similarity, respectively.

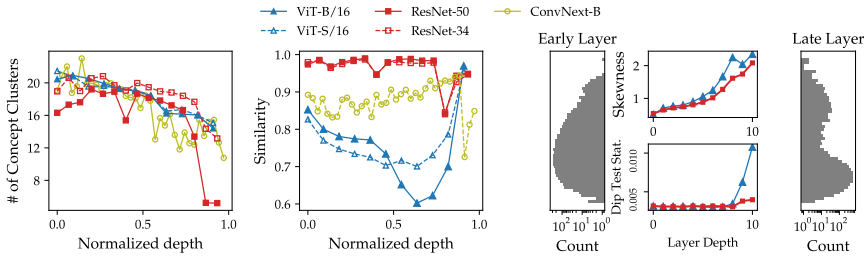


Fig. 7: Polysemanticity differs across network architectures. **(Left)** The number of concepts per neuron across layers. **(Center)** Intra-cluster similarity across layers. **(Right)** A structural evolution where the similarity distribution shifts from unimodal to bimodal.

These behavioral differences stem from the evolution of feature-similarity distributions (Figure 7, right). While all models begin with unimodal distributions indicating limited separation in early layers, they transition to distinctly bimodal distributions later, reflecting clear concept splits. This multimodality shift, quantified via Hartigan's Dip Test is most pronounced in ViTs. Ultimately, these contrasting trajectories highlight distinct inductive biases: CNNs preserve cohesive concepts driven by local processing, while ViTs undergo mid-layer diversification and late-stage reconsolidation via global self-attention.

Table 2: Separability of ViT-B/16 across activation ranges (Top/Mid/Bottom), showing the same depth-wise trend as Table 1.

Method	B2			B4			B6			B8			B10		
	T	M	B	T	M	B	T	M	B	T	M	B	T	M	B
PURE	1.02	1.00	1.02	1.02	1.00	1.02	1.03	1.00	1.01	1.03	1.00	1.01	1.07	1.00	1.01
CPE	0.99	0.99	0.99	1.01	0.99	0.99	0.99	0.99	1.00	1.01	0.99	0.99	1.01	0.99	0.99
LE	**1.20**	**1.11**	1.12	1.15	**1.15**	1.13	1.19	**1.18**	1.13	**1.21**	1.14	1.13	1.33	1.13	1.04
Ours	1.14	1.09	**1.12**	**1.19**	1.14	**1.16**	**1.24**	1.18	**1.21**	1.20	**1.31**	**1.29**	**1.48**	**1.27**	**1.41**

Ablation Studies The validity of the aforementioned scientific insights hinges on the robustness of the SPICE framework. Here, we thoroughly evaluate our core design choices.

Robustness across Activation Ranges. Standard interpretability evaluation protocols often restrict their analysis to the highly activated range (e.g., Top-10%) of a neuron. To ensure our method's robustness and comprehensively address the evaluation scope, we expand our quantitative analysis to cover the **Top**, **Middle**, and **Bottom** activation ranges (100 samples each) across varying depths. As shown in Table 2, SPICE consistently maintains high separability scores significantly above the PURE and CPE baselines across all depths and ranges. Notably, while LE achieves competitive scores in early layers (e.g., B2), its performance degrades substantially in deeper layers (B10) and lower activation ranges (Bottom: 1.04). In contrast, SPICE demonstrates superior robustness, achieving a dominant score of 1.41 even in the B10 Bottom range. This confirms that SPICE effectively disentangles concepts regardless of activation magnitude, successfully operating even in highly polysemantic and less activated spaces where existing text-prior-based methods fail.

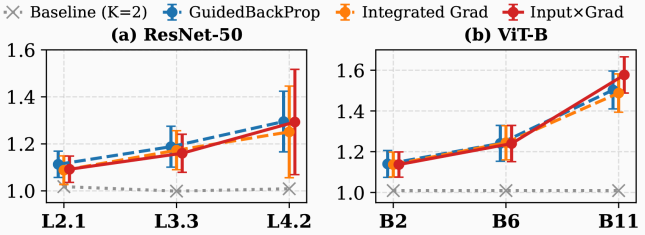


Fig. 8: Robustness to attribution methods. SPICE consistently outperforms PURE across Integrated Gradients, GradientShap, Guided Backpropagation, and Input × Gradient.

Robustness to Attribution Methods. To validate our core design choices, we first examine whether SPICE's disentanglement relies on the specific attribution method used to compute the representational footprints. As shown in Figure 8,

Table 3: Adaptive vs. static thresholding. For each variant we report the separability score and the coverage (# of 100 randomly sampled neurons yielding at least one cohesive cluster). *Fixed* $K = 2$ accepts every cluster (no cohesion threshold $\Rightarrow$ 100% coverage by construction). *Static* τ applies a global cosine-cohesion threshold; high τ inflates separability but drops coverage to near zero because most neurons cannot meet the threshold. *Adaptive (SPICE)* sets τ per neuron, achieving near-100% coverage with strong separability.

	ResNet-50 (L3.3)		**ViT-B (B6)**	
Variant	Score	Coverage	Score	Coverage
Fixed $K = 2$ (no τ)	1.00	100/100	1.03	100/100
Static $\tau \geq 0.9$	1.56	6/100	2.18	7/100
Static $\tau \geq 0.8$	1.56	9/100	2.18	10/100
Static $\tau \geq 0.7$	1.39	27/100	1.88	21/100
Adaptive τ (SPICE)	**1.15**	**100/100**	**1.25**	**100/100**

SPICE consistently outperforms the PURE baseline regardless of the attribution method employed—including Integrated Gradients [31], GradientShap [20], Guided Backpropagation [30], and Input $\times$ Gradient. This confirms that the significant performance gain stems from our framework itself, rather than an artifact of a specific metric. We adopt Input $\times$ Gradient as our default choice due to its efficiency.

Adaptive vs. Static Thresholding. Finally, we evaluate the necessity of our adaptive clustering algorithm by comparing it against two natural alternatives: (i) accepting every cluster from a fixed $K = 2$ partition (no cohesion threshold), and (ii) applying a single *global* cohesion threshold τ shared across all neurons. We measure both separability and *coverage*, defined as the number of neurons (out of 100 randomly sampled) for which at least one cluster passes the cohesion test; fixed-K trivially achieves 100% coverage by construction since it imposes no threshold. As shown in Table 3, a strict global τ (e.g., $\tau \geq 0.9$) yields high raw separability but applies to only 6–7 out of 100 neurons, since after attribution normalization the pairwise similarities concentrate in a narrow range well below 0.9 and most neurons have no cluster cohesive enough to clear the threshold. Loosening τ raises coverage but degrades the score. SPICE's per-neuron adaptive τ instead achieves near-100% coverage while maintaining strong separability over the same population, demonstrating that the threshold must be calibrated per neuron to generalize interpretation across the entire network. We further statistically confirm in

5 Limitations

While SPICE generally yields superior separability, the utility of more advanced attribution methods warrants further investigation. Moreover, the lower separa-

bility observed in the early layers (e.g., B2) of ViT-B highlights a structural constraint of the CLIP verification protocol itself as early layers capture low-level features that are not easily distinguishable within CLIP's high-level semantic space. This underscores that numerical assessments of concept quality heavily depend on the external semantic space used for verification. Despite its generality, our method presents practical constraints: generating human-understandable labels relies on external VLMs, and the iterative clustering process is more computationally intensive than single-pass methods. These challenges motivate clear avenues for future work, such as developing strategies to harness VLM knowledge while compensating for representational misalignment. Ultimately, the framework can be extended toward circuit-level disentanglement, moving beyond single neurons to systematically trace the formation mechanisms of concepts along their pathways.

6 Conclusion

In this work, we addressed the challenge of polysemanticity—the tendency of a single neuron to encode multiple, often unrelated concepts—which obscures the functional interpretation in deep models. We introduce SPICE (Simple Polysemantic Feature Interpretation via Clustering-based Explanation), a simple and generalizable framework that overcomes the architectural constraints and heuristic assumptions (e.g., fixed K) of prior approaches. SPICE effectively disentangles the visual concepts encoded within individual neurons, enabling the first systematic comparison of polysemanticity across both CNNs and Transformers. By automatically determining the number of concept clusters per neuron, it eliminates reliance on manual hyperparameters and provides a scalable analysis pipeline suitable for large modern architectures. Using SPICE, we conduct a comprehensive analysis across diverse deep vision architectures. We believe this work highlights the importance of shifting interpretability from single-neuron analyses to the concept clusters they encode, providing a more faithful lens through which to understand the internal mechanisms of deep visual models.

Acknowledgements

This work was supported by the Institute for Information & Communications Technology Planning & Evaluation (IITP) grant funded by the Korea government (MSIT) (RS-2019-II190075, Artificial Intelligence Graduate School Support Program (KAIST); RS-2024-00457882, AI Research Hub Project; and RS-2022-II220984, Development of Artificial Intelligence Technology for Personalized Plug-and-Play Explanation and Verification of Explanation).

References

1. Achiam, J., Adler, S., Agarwal, S., Ahmad, L., Akkaya, I., Aleman, F.L., Almeida, D., Altenschmidt, J., Altman, S., Anadkat, S., et al.: Gpt-4 technical report. arXiv preprint arXiv:2303.08774 (2023)

2. Arrieta, A.B., Díaz-Rodríguez, N., Del Ser, J., Bennetot, A., Tabik, S., Barbado, A., García, S., Gil-López, S., Molina, D., Benjamins, R., et al.: Explainable artificial intelligence (xai): Concepts, taxonomies, opportunities and challenges toward responsible ai. Information fusion **58**, 82–115 (2020)
3. Bansal, N., Chen, X., Wang, Z.: Can we gain more from orthogonality regularizations in training deep networks? Advances in Neural Information Processing Systems **31** (2018)
4. Bau, D., Zhou, B., Khosla, A., Oliva, A., Torralba, A.: Network dissection: Quantifying interpretability of deep visual representations. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 6541–6549 (2017)
5. Bereska, L., Gavves, E.: Mechanistic interpretability for ai safety–a review. arXiv preprint arXiv:2404.14082 (2024)
6. Cao, T.M., Hoang-Xuan, N., Pham, H.H., Nguyen, P.L., Thai, M.T.: Neurflow: Interpreting neural networks through neuron groups and functional interactions. In: The Thirteenth International Conference on Learning Representations, ICLR 2025, Singapore, April 24-28, 2025. OpenReview.net (2025), https://openreview.net/forum?id=GdbQyFOUlJ
7. Cunningham, H., Ewart, A., Riggs, L., Huben, R., Sharkey, L.: Sparse autoencoders find highly interpretable features in language models. arXiv preprint arXiv:2309.08600 (2023)
8. Dreyer, M., Purelku, E., Vielhaben, J., Samek, W., Lapuschkin, S.: Pure: Turning polysemantic neurons into pure features by identifying relevant circuits. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 8212–8217 (2024)
9. Elhage, N., Hume, T., Olsson, C., Schiefer, N., Henighan, T., Kravec, S., Hatfield-Dodds, Z., Lasenby, R., Drain, D., Chen, C., et al.: Toy models of superposition. arXiv preprint arXiv:2209.10652 (2022)
10. Elhage, N., Lasenby, R., Olah, C.: Privileged bases in the transformer residual stream. Transformer Circuits Thread p. 24 (2023)
11. Elhage, N., Nanda, N., Olsson, C., Henighan, T., Joseph, N., Mann, B., Askell, A., Bai, Y., Chen, A., Conerly, T., et al.: A mathematical framework for transformer circuits. Transformer Circuits Thread **1**(1), 12 (2021)
12. Fel, T., Lubana, E.S., Prince, J.S., Kowal, M., Boutin, V., Papadimitriou, I., Wang, B., Wattenberg, M., Ba, D., Konkle, T.: Archetypal SAE: Adaptive and stable dictionary learning for concept extraction in large vision models. arXiv preprint arXiv:2502.12892 (2025)
13. Godey, N., de la Clergerie, É., Sagot, B.: Anisotropy is inherent to self-attention in transformers. arXiv preprint arXiv:2401.12143 (2024)
14. Hesse, R., Fischer, J., Schaub-Meyer, S., Roth, S.: Disentangling polysemantic channels in convolutional neural networks. In: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 4799–4803 (2025)
15. Kalibhat, N., Bhardwaj, S., Bruss, C.B., Firooz, H., Sanjabi, M., Feizi, S.: Identifying interpretable subspaces in image representations. In: International Conference on Machine Learning. pp. 15623–15638. PMLR (2023)
16. Kopf, L., Bommer, P.L., Hedström, A., Lapuschkin, S., Höhne, M.M., Bykov, K.: Cosy: Evaluating textual explanations of neurons. In: The Thirty-eighth Annual Conference on Neural Information Processing Systems (2024), https://openreview.net/forum?id=R0bnWrpIeN
17. Kowal, M., Wildes, R.P., Derpanis, K.G.: Visual concept connectome (vcc): Open world concept discovery and their interlayer connections in deep models. In: Pro-

ceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 10895–10905 (2024)
18. Kwon, D., Lee, S., Choi, J.: Granular concept circuits: Toward a fine-grained circuit discovery for concept representations. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 2356–2365 (2025)
19. LeCun, Y., Bengio, Y., Hinton, G.: Deep learning. Nature **521**(7553), 436–444 (2015)
20. Lundberg, S.M., Lee, S.: A unified approach to interpreting model predictions. In: Guyon, I., von Luxburg, U., Bengio, S., Wallach, H.M., Fergus, R., Vishwanathan, S.V.N., Garnett, R. (eds.) Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA. pp. 4765–4774 (2017), https://proceedings.neurips.cc/paper/2017/hash/8a20a8621978632d76c43dfd28b67767-Abstract.html
21. Marshall, S.C., Kirchner, J.H.: Understanding polysemanticity in neural networks through coding theory. arXiv preprint arXiv:2401.17975 (2024)
22. Oikarinen, T., Weng, T.W.: Linear explanations for individual neurons. In: International Conference on Machine Learning (2024)
23. Oikarinen, T.P., Weng, T.: Clip-dissect: Automatic description of neuron representations in deep vision networks. In: The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023. OpenReview.net (2023), https://openreview.net/forum?id=iPWiwWHc1V
24. Olah, C., Cammarata, N., Schubert, L., Goh, G., Petrov, M., Carter, S.: Zoom in: An introduction to circuits. Distill **5**(3), e00024–001 (2020)
25. Olah, C., Mordvintsev, A., Schubert, L.: Feature visualization. Distill **2**(11), e7 (2017)
26. Razzhigaev, A., Mikhalchuk, M., Goncharova, E., Oseledets, I., Dimitrov, D., Kuznetsov, A.: The shape of learning: Anisotropy and intrinsic dimensions in transformer-based models. In: Findings of the Association for Computational Linguistics: EACL 2024. pp. 868–874 (2024)
27. Selvaraju, R.R., Cogswell, M., Das, A., Vedantam, R., Parikh, D., Batra, D.: Grad-cam: Visual explanations from deep networks via gradient-based localization. In: Proceedings of the IEEE international conference on computer vision. pp. 618–626 (2017)
28. Shrikumar, A., Greenside, P., Kundaje, A.: Learning important features through propagating activation differences. In: International conference on machine learning. pp. 3145–3153. PMlR (2017)
29. Simonyan, K., Vedaldi, A., Zisserman, A.: Visualising image classification models and saliency maps. Deep Inside Convolutional Networks **2**(2) (2014)
30. Springenberg, J.T., Dosovitskiy, A., Brox, T., Riedmiller, M.A.: Striving for simplicity: The all convolutional net. In: Bengio, Y., LeCun, Y. (eds.) 3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Workshop Track Proceedings (2015), http://arxiv.org/abs/1412.6806
31. Sundararajan, M., Taly, A., Yan, Q.: Axiomatic attribution for deep networks. In: Precup, D., Teh, Y.W. (eds.) Proceedings of the 34th International Conference on Machine Learning, ICML 2017, Sydney, NSW, Australia, 6-11 August 2017. Proceedings of Machine Learning Research, vol. 70, pp. 3319–3328. PMLR (2017), http://proceedings.mlr.press/v70/sundararajan17a.html
32. Thasarathan, H., Forsyth, J., Fel, T., Kowal, M., Derpanis, K.G.: Universal sparse autoencoders: Interpretable cross-model concept alignment. In: Forty-second International Conference on Machine Learning (2025)

33. Wang, K., Variengien, A., Conmy, A., Shlegeris, B., Steinhardt, J.: Interpretability in the wild: a circuit for indirect object identification in gpt-2 small. arXiv preprint arXiv:2211.00593 (2022)
34. Yu, W., Wang, Q., Liu, C., Li, D., Hu, Q.: Coe: Chain-of-explanation via automatic visual concept circuit description and polysemanticity quantification. In: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 4364–4374 (2025)
35. Zeiler, M.D., Fergus, R.: Visualizing and understanding convolutional networks. In: ECCV. pp. 818–833. Springer (2014)
36. Zhai, X., Mustafa, B., Kolesnikov, A., Beyer, L.: Sigmoid loss for language image pre-training (2023), https://arxiv.org/abs/2303.15343