PLACE: Patch-Level Agnostic Concept Extraction

Gabriele Onorato[1,2,*] **Fabrizio Silvestri**[2] **Nathaniel D. Bastian**[3] **Francesco Restuccia**[1]

[1]Northeastern University [2]Sapienza University of Rome [3]Syracuse University

Abstract

While Vision Foundation Models (VFMs) are increasingly used in several computer vision tasks, their internal representations remain substantially opaque. This mostly stems from *polysemanticity*, i.e., individual neurons encode mixtures of unrelated visual patterns. In stark contrast, human reasoning is inherently *monosemantic*, i.e., it naturally relies on distinct, isolated features to process information. Therefore, monosemanticity needs to be implemented within the model's activation space to obtain *interpretable concept representations*. Existing methods mostly operate on convolutional networks, so when applied to VFMs their "crop-and-resize" approach distorts the model's representations by disrupting global self-attention mechanisms and discarding spatial geometry. Furthermore, prior work requires downstream classification labels and is based on KL-divergence, which requires propagating gradients back from the classifier head, ultimately incurring excessive concept extraction time and making it hardly applicable to extract task-agnostic concepts. To overcome these issues, we propose PLACE, a *label-free* concept extraction framework that operates directly on the native patch-token activations of a VFM. We introduce a geometric Gram-matrix alignment loss that, jointly with reconstruction, upper-bounds the downstream KL divergence of any linear classifier, providing a label-free proxy for *faithfulness*, i.e., the notion that extracted concepts, although obtained without labels, are relevant to downstream classification tasks. Evaluations on image classification tasks (ImageNet and COCO) on DINOv2, ViT-MAE and ViT-B/16 show that PLACE ❶ extracts concepts about $20\times$ faster than KL-supervised extraction and $11\times$ faster than Sparse Autoencoders (SAEs) using our Gram alignment loss. Moreover, PLACE yields ❷ sparser concepts than SAEs in four of six backbone and dataset configurations (up to 0.678 vs. 0.276 Gini sparsity score), and ❸ concepts that align with those obtained under classifier supervision (maximum cosine similarity of 0.923 on ViT-MAE). Finally, when a single concept bank is learned from unlabeled multi-class data, the Gram loss improves this alignment over reconstruction loss alone.

1 Introduction

Vision Foundation Models (VFMs) [23] are currently seen as "black boxes", i.e., mostly lacking human-understandable representations [14, 46, 62]. This opacity has become a barrier to the safe deployment of VFMs into real-world scenarios such as AI-guided surgeries or medical imaging [3, 4, 44], where relying on an unexplainable decision process poses unacceptable risks to human life [62]. A VFM is difficult for humans to understand because it compresses the vast amount of visual information seen during training into its activation space. This results in *polysemanticity* [41], i.e., a single neuron fires in response to a mixture of unrelated visual patterns, which ultimately obscures their decision-making process [57]. This behavior stands in stark contrast to the *monosemantic* nature of human reasoning; when perceiving and describing the visual world, humans naturally rely on

*Corresponding author: onorato.g@northeastern.edu

Code available at https://gabrieleonorato.com/PLACE

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

distinct, isolated features [8]. Therefore, to make VFMs more interpretable, it is necessary to recover *monosemanticity* from their internal representations.

To achieve this alignment, one line of work, building on the linear representation hypothesis [58], defines the notion of *concept* as a *monosemantic* direction within the model's latent activation space [7, 24–26, 35]. While a monosemantic direction might visually align with a human-understandable or visually interpretable concept, *it does not guarantee that the model actively relies on it to execute its task*. For example, if the model is performing classification, a concept extraction method might identify a *rabbit ear* as a relevant concept, whereas the model could actually be using the *rabbit fur* to classify the image correctly. Without a mechanism to verify this alignment, extracted concepts risk being misleading, offering plausible but ultimately unreliable explanations of the model's behavior. As such, *faithfulness* addresses this risk by ensuring the concept influences the model's predictions (a formal definition of concept and faithfulness are provided in Section 3).

The key issue is that most prior methods were designed for convolutional architectures (CNNs) [25, 26, 35, 71]. Current attempts to interpret Vision Transformers (ViTs) rely on SAEs [33, 57, 61], architectural modifications [5, 9–11, 36, 50, 69], or external language models [15, 32, 40]. *These approaches share a critical issue: they require training auxiliary modules or fine-tuning the model's weights*. However, imposing a retraining phase in VFMs is computationally prohibitive and may degrade their generalization capabilities [39]. Consequently, extracting faithful concepts without relying on retraining with task-specific labels is a compelling necessity.

There are several technical challenges to achieve this goal. First, VFMs do not possess a classifier head, which removes the explicit task-driven signal that *all existing methods* rely on to enforce faithfulness [7]. Second, existing extraction methods rely on cropping and resizing images [7, 25, 71]. While effective for convolutional architectures, this approach disrupts the global self-attention mechanisms of ViTs [59], biasing concept extraction (formalized in Section 3). Finally, since prior approaches extract concepts from globally aggregated representations, they inherently discard the image's spatial geometry. Consequently, prior work is forced to rely on gradient-based attribution techniques to locate and identify concepts within the original input [25].



Figure 1: ***Example of* PLACE *applied to a picture of a church***. While current literature works on CNNs and yields image crops as the extracted concepts, PLACE returns a fine-grained heatmap localizing the relevant concepts in the whole image.

To address these issues, we propose PLACE, a *label-free* concept extraction framework exemplified in Figure 1 and specifically designed for ViTs and VFMs [13, 30, 56, 64]. In opposition with state of the art work CRAFT [25] and FACE [7], PLACE operates directly on patch-token activations of a frozen backbone and replaces classifier-based faithfulness supervision with a Gram-matrix alignment loss. This approach allows PLACE to overcome the limitations of prior work and yield spatially resolved concept heatmaps without the need for architectural modifications or class labels while achieving highly stable concept extraction (up to 0.95 W-Stab, see Appendix E.7) without demanding downstream labels.

Summary of Novel Contributions:

- **Sparsity & Efficiency:** PLACE yields sparser concepts than SAEs in four of six configurations – quantitatively measured via a high Gini sparsity of 0.933 – while accelerating the extraction pipeline by $11\times$ compared to SAEs and over $20\times$ compared to KL-supervised alternatives (Section 4.1, Appendix E.9).
- **Label-Free Faithfulness via Geometric Alignment:** We introduce an unsupervised Gram-matrix alignment loss and show (Section 3.1) that reconstruction error and Gram deviation are jointly sufficient to control downstream (D_{KL}) for any linear classifier, thus grounding faithfulness without labels.

- **Supervised Alignment without Labels:** We demonstrate via RSVC [37] that our extracted concepts align (up to 0.923 maximum cosine similarity) with those extracted under full classifier supervision, and that, when a single concept bank is learned from unlabeled multi-class data, the Gram loss improves this alignment over reconstruction alone (MCS $+0.12$ to $+0.55$; Section 4.4).
- **Comprehensive Benchmarking:** We systematically evaluate PLACE against state of the art concept extraction baselines CRAFT [25] and FACE [7], which were adapted for sliding-window ViTs [37], and SAEs [33, 57, 61]. Our experimental evaluation indicates that PLACE avoids the representational distortion of sliding-window methods (Section 4.2) while delivering spatially resolved concept heatmaps.

2 Related Work

Concept-based explanations. TCAV [35] pioneered concept-based interpretability by training linear classifiers on activation vectors of user-defined concepts and random counterexamples to measure a classifier's sensitivity to specific concept directions. However, TCAV relies on manually curated concept datasets, limiting scalability and introducing human bias. ACE [26] addresses this by segmenting images into superpixels via SLIC [1] and clustering segment-level activations to discover concepts automatically, at the cost of potential segmentation artifacts. ICE [71] replaces clustering with Non-negative Matrix Factorization (NMF) [42], offering cleaner concept dictionaries but operating at the level of convolutional kernels only. CRAFT [25] improves on ICE with a recursive factorization pipeline and Sobol-based relevance scoring [65], but does not enforce predictive consistency through the classifier. FACE [7] introduced a KL-regularized NMF framework that explicitly aligns discovered concepts with the classifier's predictions, at the cost of requiring class-label supervision. Concept Bottleneck Models (CBMs) [36] force predictions through an explicit concept layer, but require concept annotations at training time and architectural changes to the model. Concept Embedding Models [69] mitigate the predictive performance drop of standard CBMs by learning high-dimensional concept embeddings instead of scalar activations, though they still require architectural modifications and concept supervision during training.

Achtibat et al. [2] proposed Concept Relevance Propagation (CRP) and Relevance Maximization (RelMax). CRP propagates relevance scores to concept subspaces, identifying both *where* and *what* concepts contribute to individual predictions, while RelMax visualizes concepts by selecting representative training examples that maximize feature relevance during inference. However, CRP relies on layer-wise architecture-specific relevance propagation rules and RelMax requires access to the training data, which for VFMs like the DINO family [13, 56, 64] or Masked Autoencoders (ViT-MAE) [30] is typically either inaccessible or too large to navigate meaningfully. Sparse autoencoders (SAEs) [33, 57], instead, learn interpretable directions from model activations without label supervision. Recently, Gated SAEs [61] addressed an issue of SAEs, where the ℓ_1 penalty required to enforce sparsity inherently shrinks the magnitude of active features, degrading reconstruction fidelity. However, all SAE-based approaches require training a separate encoder-decoder architecture, thus adding significant computational overhead. A separate line of work leverages vision-language models (VLMs) to bypass manual concept curation entirely: LG-CAV [32] uses a VLM to score a shared pool of probe images according to textual concept descriptions, and uses those scores as supervision to learn concept directions in the target model without labeled concept images. CE-FAM [40] and SSG-CAM [15] learn to mimic VLM embeddings through a branched network with gradient-weighted activation-map fusion. While effective, these methods risk hallucinating explanations, directly inheriting the flaws of the external language models they rely on [43].

Concepts in Vision Transformers. Conversely from CNN feature maps, ViTs encode information through patch-token sequences and self-attention[23]. Dorszewski et al. [22] provide a layer-wise analysis of ViT concepts by labeling individual neurons via CLIP-Dissect [55], showing that ViTs develop a coarse-to-fine concept hierarchy despite lacking CNN inductive biases. B-cos ViTs [9–11] reformulate every ViT component as a dynamic linear mapping so that predictions can be faithfully summarized by a single linear transform, yet require retraining the model. Arya et al. [5] fine-tune existing pre-trained models into inherently interpretable networks. ProtoViT [50] extends prototype-based reasoning to ViT backbones with spatially deformable prototypes, but the prototype layer itself is an architectural addition trained jointly with the model. In contrast, *PLACE operates on the native patch-token activations of a frozen ViT*. This inherently preserves spatial structure, yielding high-quality heatmaps without introducing architectural overhead. Furthermore, we replace task-specific classifier supervision with an unsupervised Gram-matrix alignment. This allows us to guide the extraction toward faithful concepts without labels or external models.

3 The PLACE Framework

Prior concept extraction methods [7, 25, 71] assume a supervised classification setting in which a model $f : \mathcal{X} \to \mathcal{Y}$ decomposes into an encoder $g : \mathcal{X} \to \mathcal{G}$ and a classifier head $h : \mathcal{G} \to \mathcal{Y}$, so that $f(\mathbf{x}) = (h \circ g)(\mathbf{x})$. The classifier head (i) provides class-label supervision for concept discovery [7] and (ii) defines the output distribution against which faithfulness can be measured [25, 35]. However VFMs break this assumption; models such as DINO [13, 56, 64] or ViT-MAE [30] produce general-purpose embeddings $f : \mathcal{X} \to \mathcal{Z}$ with no classification head attached. As such, there is no h to supervise concept extraction, no class-label signal to enforce predictive consistency, and the representation space $\mathcal{Z}$ is not organized around decision boundaries. Consequently, the KL-regularized NMF objective of FACE [7], which aligns the factorized activations with the classifier's output distribution, can be hardly applied to ViTs.

Effect of crop-and-resize on ViTs. CRAFT [25], FACE [7], and their ViT adaptations [37] extract concepts by cropping sub-regions from an image, resizing each crop to the model's input resolution, feeding it to the model, and factorizing the resulting activation vectors. For CNNs, this is justified by translation equivariance [16, 70]. ViTs, however, process information in a different way [23, 27, 54, 59, 68]. This difference in input can be seen in Figure 2, and we discuss this mismatch formally in Appendix A. Due to this, our patch-token extraction and the sliding-window baseline are not directly comparable on classifier-dependent metrics (C-Ins, C-Del, D_{KL}), since they operate on activation spaces of different scale and address different input scenarios; we return to this in Section 4.2.

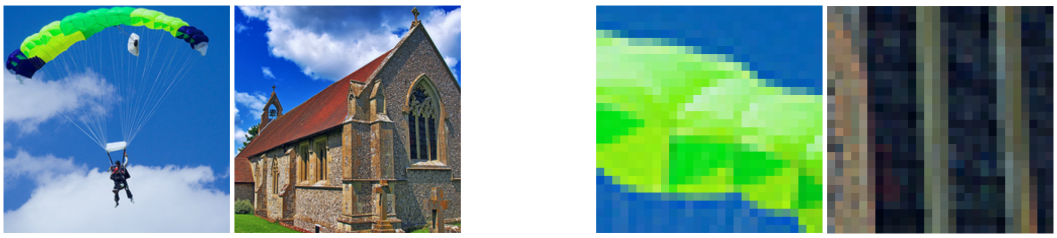


Figure 2: **The Crop-and-Resize Problem in ViTs.** While PLACE operates on the full-resolution image seen at training time (left), current crop-and-resize methods feed sub-resolution patches (right). We show in Section 4.2 this shift from the training distribution biases concept extraction.

Concept Extraction. We formulate concept extraction as a geometric problem on the backbone's internal representations. Let f be a ViT with L layers. As illustrated in Figure 3, we split f at a chosen layer ℓ into a prefix $g_{\leq \ell} : \mathcal{X} \to \mathbb{R}^{N \times d}$, which maps an input image to a sequence of N patch-token activations of dimension d, and a suffix $g_{>\ell}$, comprising the remaining $L - \ell$ transformer blocks. Given a dataset of n images, the prefix produces an activation tensor $\mathcal{A} \in \mathbb{R}^{n \times N \times d}$. Unlike CNNs, where a global average pool reduces spatial dimensions before factorization [7, 25], the patch-token structure of ViTs preserves a regular spatial grid (e.g. 14×14 for a 224×224 input with patch size 16), which we exploit directly for concept localization.

To decompose the activation space into visually interpretable components, we reshape the patch-token tensor $\mathcal{A}$ into an activation matrix $\mathbf{A} \in \mathbb{R}^{nN \times d}$, where each row corresponds to a single patch token. Since activations of transformer models (e.g., GELU, LayerNorm) contain negative values, we employ Semi-NMF [21]:

$$(\mathbf{U}, \mathbf{W}) = \arg \min_{\mathbf{U} \geq 0,\, \mathbf{W}} \tfrac{1}{2} \left\| \mathbf{A} - \mathbf{U}\mathbf{W}^\top \right\|_F^2 \tag{1}$$

where $\mathbf{W} \in \mathbb{R}^{d \times r}$ is the concept bank and $\mathbf{U} \in \mathbb{R}_{\geq 0}^{nN \times r}$ the coefficient matrix. This factorization naturally operationalizes our core interpretability primitives:

- **Concept:** A basis vector $\mathbf{w}_j \in \mathbb{R}^d$ (the j-th column of $\mathbf{W}$) representing a consistent, monosemantic direction in the latent representation space. The corresponding column $\mathbf{u}_j \in \mathbb{R}_{\geq 0}^{nN}$ encodes the magnitude of this concept across all tokens, i.e., *how intensely* the concept $\mathbf{w}_j$ appears in the scene.
- **Faithfulness:** A concept bank $\mathbf{W}$ is considered faithful if the reconstructed activations $\hat{\mathbf{A}} = \mathbf{U}\mathbf{W}^\top$ preserve the *predictive consistency* of the suffix $g_{>\ell}$ relative to the original activations $\mathbf{A}$.

Unlike NMF [7, 25, 42], Semi-NMF [21] leaves the basis $\mathbf{W}$ unconstrained in sign, so its columns do not form a non-subtractive, parts-based decomposition. A mixed-sign basis is necessary because transformer activations are mixed-sign: forcing $\mathbf{W} \geq 0$ would make it impossible to reconstruct $\mathbf{A}$.

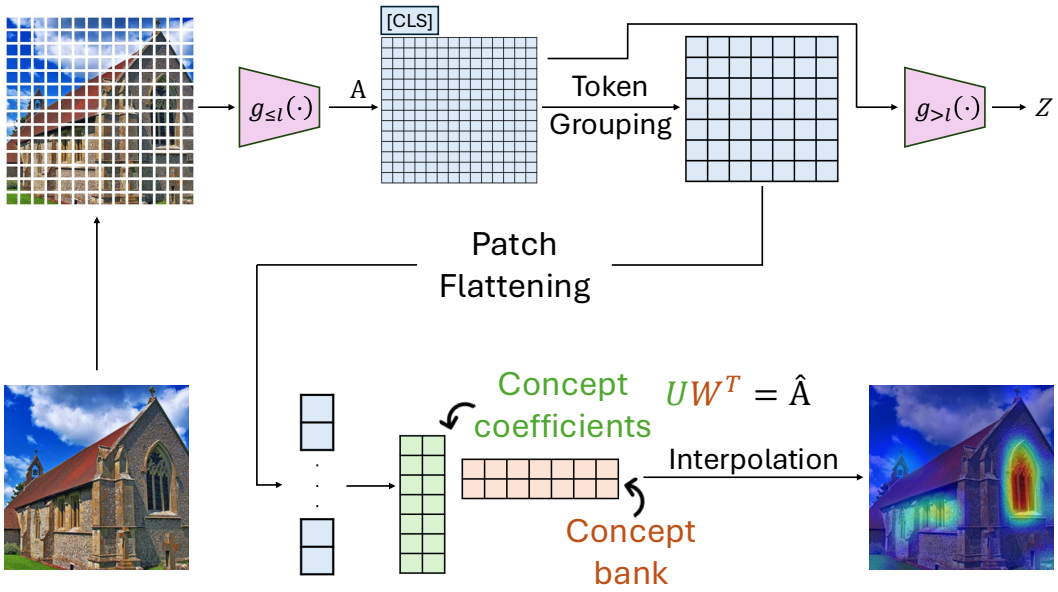


Figure 3: **PLACE Framework.** An input image is passed through the transformer prefix $g_{\leq \ell}$ to extract patch-token activations $\mathbf{A}$. To control granularity and computational cost, adjacent tokens are spatially grouped. The grouped representations are then dimensionally flattened and decomposed via Semi-NMF to discover a shared Concept bank and non-negative Concept coefficients. These coefficients are then upscaled to the original resolution to yield fine-grained concept heatmaps. The suffix $g_{>\ell}$ receives the original, ungrouped patch-token activations $\mathbf{A}$.

As such each patch-token activation is reconstructed as $\mathbf{a}_i \approx \sum_{j=1}^{r} u_{ij}\, \mathbf{w}_j$ with $u_{ij} \geq 0$, so for a fixed concept direction $\mathbf{w}_j$ every patch contributes to that concept monotonically and non-subtractively, and u_{ij} reads as the amount of concept j present at patch i. Since each row of $\mathbf{U}$ corresponds to a fixed patch position, reshaping the coefficients of a concept onto the token grid yields a per-patch map of its presence. Signed coefficients would instead allow negative values, i.e., an "anti-presence" with no coherent reading as a location in the image.

Token Grouping. Dense token grids (e.g., 37×37 for DINOv2) produce a large activation matrix $\mathbf{A} \in \mathbb{R}^{nN \times d}$, making Semi-NMF optimization expensive. To control this cost, we introduce a spatial grouping factor s: before factorization, we apply non-overlapping average pooling over $s \times s$ blocks of adjacent patch tokens, reducing the token count per image from N to $\lfloor \sqrt{N}/s \rfloor^2$. After optimization, the learned coefficients are mapped back to the original grid via nearest-neighbor upsampling (`repeat_interleave`). Setting s=2 accelerates optimization by $\sim$3$\times$ on both ImageNet and COCO (Appendix E.5). Detailed spatial arithmetic and grid boundary handling are described in Appendix B.

3.1 Geometric Faithfulness

Without a classifier head h, faithfulness cannot be measured as the divergence between predictive distributions. Instead, we preserve the *geometric structure* of activations $\mathbf{U}\mathbf{W}^\top = \hat{\mathbf{A}}$ to ensure predictive consistency. We optimize $\mathbf{U}$ and $\mathbf{W}$ by minimizing the joint objective:

$$\mathcal{L} = \underbrace{\tfrac{1}{2}\left\|\mathbf{A} - \hat{\mathbf{A}}\right\|_F^2}_{\mathcal{L}_{\text{rec}}} + \lambda \underbrace{\frac{\left\|\mathbf{A}\mathbf{A}^\top - \hat{\mathbf{A}}\hat{\mathbf{A}}^\top\right\|_F^2}{\left\|\mathbf{A}\mathbf{A}^\top\right\|_F^2}}_{\mathcal{L}_{\text{gram}}} \tag{2}$$

where $\mathcal{L}_{\text{rec}}$ is the reconstruction error (MSE) and $\mathcal{L}_{\text{gram}}$ aligns the global correlation structure $\mathbf{A}\mathbf{A}^\top$, capturing the pairwise inner products among all tokens. This metric is made invariant to the scale of representations by normalizing by $\|\mathbf{A}\mathbf{A}^\top\|_F^2$. The following proposition formalizes why jointly minimizing these two terms suffices to control downstream predictive divergence.

Proposition 1 (Informal; proof in Appendix C). *Let $h(z) = \mathbf{W}_c z + b$ be any linear classifier mapping the CLS token z to class logits, and let $\delta_{\mathrm{rec}} = \|\mathbf{A} - \hat{\mathbf{A}}\|_F$, $\delta_{\mathrm{gram}} = \sqrt{\mathcal{L}_{\mathrm{gram}}}$. Then:*

$$D_{\mathrm{KL}}\big(\sigma(h \circ g_{>\ell}(\mathbf{A})) \,\|\, \sigma(h \circ g_{>\ell}(\hat{\mathbf{A}}))\big) \;\le\; \tfrac{1}{2}\|\mathbf{W}_c\|_{\mathrm{op}}^2 L_{\mathrm{block}}^{2L_{\mathrm{suf}}} \Big(\sqrt{N}\,\delta_{\mathrm{rec}} + \alpha\,\|\mathbf{A}\mathbf{A}^\top\|_F \cdot \delta_{\mathrm{gram}}\Big)^2 \quad (3)$$

where L_{block} is the per-block Lipschitz constant, $L_{\mathrm{suf}} = L - \ell$ the number of suffix transformer blocks, α collects attention-projection constants, and $\|\cdot\|_{\mathrm{op}}$, $\|\cdot\|_F$ denote the spectral and Frobenius norms, respectively.

The bound identifies $\mathcal{L}_{\mathrm{rec}}$ and $\mathcal{L}_{\mathrm{gram}}$ as the two sufficient control variables for downstream predictive divergence (the precise mechanism is detailed in Appendix C). Being worst-case, the bound compounds multiplicatively across the L_{suf} suffix blocks and is therefore loose as a numerical predictor; its value is structural: it shows that the two losses control the KL divergence of any linear classifier without computing it, with $\mathcal{L}_{\mathrm{gram}}$ directly constraining the identity component of the attention logits, which $\mathcal{L}_{\mathrm{rec}}$ bounds only indirectly. Notably, the bound tightens as the extraction layer ℓ increases, since fewer suffix blocks reduce the compounding factor; at $\ell = L$ the composition term vanishes entirely (Appendix C). Moreover, optimizing $\mathcal{L}$ requires no forward pass and backpropagation through $g_{>\ell}$ or h, and no auxiliary network, requiring less computing than KL-supervised methods [7], which backpropagate through the full suffix at every step, and SAEs [33, 57, 61], which train a separate encoder-decoder (Appendix E.9).

3.2 Optimization and Visualization

We solve the objective in Equation (2) using AdamW [49], treating both $\mathbf{U}$ and $\mathbf{W}$ as free parameters updated via backpropagation. Unlike traditional Semi-NMF solvers [21], gradient-based optimization requires accommodating the non-convex geometric constraints introduced by $\mathcal{L}_{\mathrm{gram}}$, therefore the Semi-NMF constraint $\mathbf{U} \ge 0$ is enforced by clamping negative entries to zero, and gradient clipping is performed to ensure stability [51]. To ensure convergence toward interpretable solutions, we apply an ℓ_1 regularization on $\mathbf{U}$ (quantified in Appendix D.2) to enhance concept sparsity and an ℓ_2 penalty on $\mathbf{W}$ to stabilize the dictionary. The initialization of $\mathbf{U}$ is partially performed via NNDSVD [12], which accelerates convergence compared to random starting points (ablation in Appendix E.1), while the concept bank is initialized via the Moore–Penrose pseudoinverse $\mathbf{W} \leftarrow \mathbf{U}^\dagger \mathbf{A}$.

Heatmaps. A key advantage of PLACE is that concept heatmaps are obtained with no additional computational cost. Since each entry of $\mathbf{u}_j$ is spatially anchored to a specific patch token, the distribution of coefficients provides a direct activation map. To visualize a concept, the coefficients are reshaped into the original $\sqrt{N} \times \sqrt{N}$ grid and upscaled via bicubic interpolation. This process delivers exact *spatial localization* without requiring gradient-based attribution or implicit differentiation [25], preserving the structural integrity of the model's internal representations.

4 Experimental Results

Datasets. Following the evaluation protocol of FACE [7], we evaluate our method on matched subsets of ImageNet [20, 63] (10 classes: `cassette`, `chainsaw`, `church`, `english_springer`, `french_horn`, `garbage_truck`, `gas_pump`, `golf_ball`, `parachute`, `tench`; ≥ 800 images/class) and COCO [45] (5 classes: `hot_dog`, `teddy_bear`, `train`, `umbrella`, `zebra`).

Models and hyperparameters. We evaluate PLACE on DINOv2 [56] and ViT-MAE [30], alongside a supervised ViT-B/16 for parity. We compare against supervised baselines CRAFT [25] and FACE [7] on CNNs, and their ViT extensions [37], we train a surrogate linear probe to compute D_{KL}. For PLACE and SAE, we extract concepts from layer $\ell = 9$ and set the dictionary rank to $r = 25$ to align with FACE [7]. Hyperparameter configurations and detailed per-class ablations are provided in Appendices D and E. Crucially, evaluations confirm that PLACE passes the weight randomization sanity check (Appendix E.6), and maintains highly stable dictionaries (Appendix E.7) even under perturbations (Appendix E.13).

Metrics. Our primary evaluation relies on three metrics: KL-divergence for faithfulness, MSE for reconstruction quality, and the C-Gini index as a quantitative proxy for monosemanticity (i.e., concept sparsity). This choice aligns with our goal of task-agnostic evaluation, well suited for VFMs. Formal definitions for all metrics are deferred to Appendix F. Unless otherwise noted, results are mean $\pm$ std over 5 seeds, and significance against the SAE baselines is assessed with two-sided Welch's t-tests and Holm–Bonferroni correction; adjusted p-values are reported in Appendix E.15.



Figure 4: ***Visualizing the C-Deletion and C-Insertion protocols.*** Reading from left to right illustrates C-Deletion: the original input image undergoes progressive masking of patch-tokens based on concept activations, culminating in a fully obscured state. Conversely, reading from right to left depicts C-Insertion, where patches are progressively revealed starting from the fully masked baseline.

We additionally report C-Insertion and C-Deletion [7, 25], which measure confidence recovery as concept patches are progressively revealed or masked, respectively. These causal metrics require a classifier head and a downstream task (Appendix D.1), and we include them strictly for a more thorough comparison with prior work. We note, however, that they carry three well-known limitations: masking creates artifacts [28] (visualized in Figure 4), the sequential removal order confounds individual concept contributions [18], and the protocol implicitly assumes isolated single-concept reasoning, ignoring the multi-concept interactions typical of deep networks.

4.1 Benchmark Comparison

We focus on metrics aligned with our task-agnostic goal, organized by their dependency on a downstream classifier. First, two metrics require *no classifier*: MSE for reconstruction quality and C-Gini for concept sparsity (i.e., monosemanticity). Second, we report D_{KL} for predictive consistency; crucially, although D_{KL} requires a classification head, we use it purely for evaluation purposes (Appendix D.1). We also report C-Ins and C-Del to ensure comparison with prior work, even if they are incomparable across different architectures and classification heads. Consequently, the scores for ICE [71], CRAFT [25], and FACE [7] on ResNet-34 [29], as well as sliding-window adaptations on ViTs [37], are provided in Table 1 as methodological references rather than as direct comparisons. Focusing on the PLACE–SAE comparison, where both methods use matched dictionary rank $r{=}25$ for a controlled evaluation, Table 1 supports the following findings after Holm–Bonferroni correction.

❶ **Sparsity.** PLACE yields higher C-Gini than both SAE variants in four of six configurations: all three backbones on ImageNet, and ViT-B/16 on COCO. On DINOv2/ImageNet, PLACE reaches 0.933 (reconstruction) and 0.921 (+Gram), against 0.865 for SAEs and 0.724 for Gated SAEs; on ViT-MAE/ImageNet, PLACE (+Gram) reaches 0.678, against 0.238 and 0.276. On COCO with DINOv2 and ViT-MAE the differences are not significant. C-Gini is therefore the most consistent empirical advantage of PLACE.

❷ **Reconstruction and predictive consistency.** On DINOv2/ImageNet, Gated SAEs obtain lower MSE and D_{KL} than PLACE, while the corresponding comparisons with vanilla SAEs are not significant. In the remaining five configurations we detect no significant difference in MSE or D_{KL}. We thus observe a tradeoff between sparsity and reconstruction rather than uniform superiority of either method.

❸ **Causal metrics.** No C-Ins difference is significant in any configuration. PLACE improves C-Del on COCO/ViT-B/16 (0.979 vs. 0.115 and 0.033); the remaining C-Del differences are not significant.

Runtime. On DINOv2 ViT-B/14 (ImageNet, median over 10 classes, single RTX 5060 Ti), optimizing PLACE (+Gram) takes 5.6 s per class, against 118.0 s for PLACE (+KL) and 62.5 s and 71.4 s for training SAEs and Gated SAEs, i.e., about $20\times$ and $11\times$ faster (Appendix E.9). The Gram loss adds negligible overhead over reconstruction alone (5.3 s), since it requires no forward or backward pass through $g_{>\ell}$ or h.

4.2 Patch-Token Extraction vs. Crop-and-Resize

To address the input mismatch (Section 3), we compare PLACE against CRAFT [25] and FACE [7] adapted for ViTs by Kondapaneni, Aodha, and Perona [37]. This variation factorizes the final CLS

Table 1: Faithfulness, complexity, and reconstruction quality on IMAGENET and COCO. Prior art on ResNet-34 and sliding-window adaptations on ViTs are provided strictly as magnitude and methodological references. PLACE is evaluated on DINOv2 ViT-B/14, supervised ViT-B/16, and ViT-MAE B/16 (r=25, ℓ=9). Higher is better: C-Ins, C-Del, C-Gini; lower: MSE, D_{KL}. Mean and standard deviation over 5 runs. For each metric and configuration, the top-performing PLACE variant is compared with the strongest SAE baseline (two-sided Welch's t-test, Holm–Bonferroni correction across the complete family of comparisons, $\alpha = 0.05$): **bold** marks the significantly better method, on either a PLACE or an SAE row; unbolded cells are not significant after correction. Adjusted p-values are reported in Appendix E.15.

Dataset	Method	Backbone	C-Ins ↑	C-Del ↑	C-Gini ↑	MSE ↓	D_{KL} ↓
IMAGENET							
Context: Prior Art on Convolutional Networks							
	ICE [71]	ResNet-34	$0.908_{\pm 0.034}$	$0.484_{\pm 0.063}$	$0.537_{\pm 0.071}$	$0.296_{\pm 0.001}$	$0.359_{\pm 0.004}$
	CRAFT [25]		$0.932_{\pm 0.001}$	$0.752_{\pm 0.031}$	$0.835_{\pm 0.031}$	$0.451_{\pm 0.004}$	$0.240_{\pm 0.003}$
	FACE [7]		$0.969_{\pm 0.010}$	$0.891_{\pm 0.011}$	$0.895_{\pm 0.001}$	$0.497_{\pm 0.002}$	$0.220_{\pm 0.002}$
Context: Sliding-Window Adaptations on ViTs							
	CRAFT adapt. [25, 37]	DINOv2	$0.841_{\pm 0.009}$	$0.710_{\pm 0.008}$	$0.875_{\pm 0.004}$	$0.938_{\pm 0.001}$	$0.282_{\pm 0.001}$
	FACE adapt. [7, 37]		$0.914_{\pm 0.004}$	$0.829_{\pm 0.001}$	$0.898_{\pm 0.006}$	$1.015_{\pm 0.001}$	$0.371_{\pm 0.002}$
	CRAFT adapt. [25, 37]	ViT-B/16	$0.951_{\pm 0.003}$	$0.812_{\pm 0.012}$	$0.920_{\pm 0.002}$	$3.489_{\pm 0.001}$	$0.071_{\pm 0.001}$
	FACE adapt. [7, 37]		$0.955_{\pm 0.005}$	$0.825_{\pm 0.009}$	$0.926_{\pm 0.001}$	$3.510_{\pm 0.001}$	$0.062_{\pm 0.001}$
Comparison: PLACE *Framework vs. Sparse Autoencoders*							
	Gated SAE [61]	DINOv2	$0.472_{\pm 0.187}$	$0.620_{\pm 0.163}$	$0.724_{\pm 0.054}$	$\mathbf{0.073_{\pm 0.004}}$	$\mathbf{0.368_{\pm 0.161}}$
	SAE [33, 57]		$0.652_{\pm 0.212}$	$0.537_{\pm 0.188}$	$0.865_{\pm 0.012}$	$0.079_{\pm 0.004}$	$0.533_{\pm 0.235}$
	PLACE (reconstruction)		$0.467_{\pm 0.004}$	$0.895_{\pm 0.005}$	$\mathbf{0.933_{\pm 0.001}}$	$0.092_{\pm 0.001}$	$1.205_{\pm 0.044}$
	PLACE (+Gram)		$0.522_{\pm 0.014}$	$0.914_{\pm 0.003}$	$0.921_{\pm 0.003}$	$0.090_{\pm 0.001}$	$1.162_{\pm 0.099}$
	PLACE (+KL)		$0.509_{\pm 0.008}$	$0.914_{\pm 0.005}$	$0.924_{\pm 0.001}$	$0.092_{\pm 0.002}$	$1.233_{\pm 0.046}$
	Gated SAE [61]	ViT-B/16	$0.421_{\pm 0.262}$	$0.411_{\pm 0.214}$	$0.303_{\pm 0.051}$	$0.273_{\pm 0.017}$	$0.868_{\pm 0.428}$
	SAE [33, 57]		$0.528_{\pm 0.251}$	$0.497_{\pm 0.159}$	$0.434_{\pm 0.051}$	$0.284_{\pm 0.018}$	$1.101_{\pm 0.538}$
	PLACE (reconstruction)		$0.715_{\pm 0.005}$	$0.392_{\pm 0.015}$	$0.530_{\pm 0.003}$	$0.292_{\pm 0.001}$	$< 10^{-3}$
	PLACE (+Gram)		$0.701_{\pm 0.011}$	$0.401_{\pm 0.009}$	$\mathbf{0.614_{\pm 0.003}}$	$0.296_{\pm 0.001}$	$< 10^{-3}$
	PLACE (+KL)		$0.796_{\pm 0.002}$	$0.313_{\pm 0.011}$	$0.437_{\pm 0.003}$	$0.304_{\pm 0.001}$	$0.016_{\pm 0.002}$
	Gated SAE [61]	ViT-MAE	$0.734_{\pm 0.195}$	$0.477_{\pm 0.104}$	$0.276_{\pm 0.054}$	$0.980_{\pm 0.093}$	$0.285_{\pm 0.135}$
	SAE [33, 57]		$0.937_{\pm 0.090}$	$0.267_{\pm 0.183}$	$0.238_{\pm 0.042}$	$0.978_{\pm 0.093}$	$0.306_{\pm 0.144}$
	PLACE (reconstruction)		$0.911_{\pm 0.063}$	$0.586_{\pm 0.114}$	$0.438_{\pm 0.060}$	$0.995_{\pm 0.092}$	$0.302_{\pm 0.139}$
	PLACE (+Gram)		$0.989_{\pm 0.017}$	$0.161_{\pm 0.090}$	$\mathbf{0.678_{\pm 0.037}}$	$1.028_{\pm 0.090}$	$0.339_{\pm 0.153}$
	PLACE (+KL)		$0.989_{\pm 0.016}$	$0.189_{\pm 0.109}$	$0.663_{\pm 0.050}$	$1.023_{\pm 0.092}$	$0.325_{\pm 0.146}$
COCO							
Context: Prior Art on Convolutional Networks							
	ICE [71]	ResNet-34	$0.883_{\pm 0.029}$	$0.632_{\pm 0.020}$	$0.623_{\pm 0.086}$	$0.308_{\pm 0.003}$	$0.596_{\pm 0.002}$
	CRAFT [25]		$0.861_{\pm 0.029}$	$0.691_{\pm 0.029}$	$0.874_{\pm 0.035}$	$0.457_{\pm 0.002}$	$0.600_{\pm 0.004}$
	FACE [7]		$0.971_{\pm 0.013}$	$0.894_{\pm 0.010}$	$0.947_{\pm 0.002}$	$0.462_{\pm 0.003}$	$0.458_{\pm 0.001}$
Context: Sliding-Window Adaptations on ViTs							
	CRAFT adapt. [25, 37]	DINOv2	$0.974_{\pm 0.015}$	$0.487_{\pm 0.090}$	$0.883_{\pm 0.076}$	$1.237_{\pm 0.287}$	$0.011_{\pm 0.007}$
	FACE adapt. [7, 37]		$0.973_{\pm 0.016}$	$0.604_{\pm 0.099}$	$0.869_{\pm 0.083}$	$1.242_{\pm 0.288}$	$0.011_{\pm 0.006}$
	CRAFT adapt. [25, 37]	ViT-B/16	$0.977_{\pm 0.009}$	$0.596_{\pm 0.094}$	$0.874_{\pm 0.033}$	$1.023_{\pm 0.383}$	$0.020_{\pm 0.013}$
	FACE adapt. [7, 37]		$0.969_{\pm 0.014}$	$0.608_{\pm 0.097}$	$0.862_{\pm 0.040}$	$1.024_{\pm 0.383}$	$0.020_{\pm 0.013}$
Comparison: PLACE *Framework vs. Sparse Autoencoders*							
	Gated SAE [61]	DINOv2	$0.202_{\pm 0.194}$	$0.246_{\pm 0.137}$	$0.729_{\pm 0.068}$	$0.073_{\pm 0.005}$	$0.479_{\pm 0.318}$
	SAE [33, 57]		$0.143_{\pm 0.069}$	$0.276_{\pm 0.234}$	$0.859_{\pm 0.012}$	$0.080_{\pm 0.005}$	$0.589_{\pm 0.331}$
	PLACE (reconstruction)		$0.205_{\pm 0.022}$	$0.732_{\pm 0.019}$	$0.884_{\pm 0.001}$	$0.076_{\pm 0.002}$	$0.563_{\pm 0.006}$
	PLACE (+Gram)		$0.177_{\pm 0.014}$	$0.776_{\pm 0.009}$	$0.888_{\pm 0.001}$	$0.080_{\pm 0.001}$	$0.584_{\pm 0.015}$
	PLACE (+KL)		$0.173_{\pm 0.017}$	$0.768_{\pm 0.006}$	$0.887_{\pm 0.001}$	$0.080_{\pm 0.001}$	$0.576_{\pm 0.017}$
	Gated SAE [61]	ViT-B/16	$0.949_{\pm 0.063}$	$0.115_{\pm 0.103}$	$0.277_{\pm 0.041}$	$0.315_{\pm 0.038}$	$0.060_{\pm 0.028}$
	SAE [33, 57]		$0.993_{\pm 0.012}$	$0.033_{\pm 0.060}$	$0.415_{\pm 0.038}$	$0.328_{\pm 0.038}$	$0.068_{\pm 0.030}$
	PLACE (reconstruction)		$0.988_{\pm 0.001}$	$0.968_{\pm 0.004}$	$0.638_{\pm 0.005}$	$0.357_{\pm 0.004}$	$< 10^{-3}$
	PLACE (+Gram)		$0.957_{\pm 0.004}$	$\mathbf{0.979_{\pm 0.010}}$	$0.655_{\pm 0.014}$	$0.351_{\pm 0.001}$	$< 10^{-3}$
	PLACE (+KL)		$0.949_{\pm 0.004}$	$0.940_{\pm 0.004}$	$\mathbf{0.677_{\pm 0.005}}$	$0.359_{\pm 0.002}$	$0.003_{\pm 0.001}$
	Gated SAE [61]	ViT-MAE	$0.522_{\pm 0.239}$	$0.416_{\pm 0.304}$	$0.309_{\pm 0.068}$	$0.609_{\pm 0.164}$	$0.083_{\pm 0.043}$
	SAE [33, 57]		$0.692_{\pm 0.296}$	$0.272_{\pm 0.190}$	$0.200_{\pm 0.045}$	$0.595_{\pm 0.189}$	$0.076_{\pm 0.037}$
	PLACE (reconstruction)		$0.435_{\pm 0.242}$	$0.432_{\pm 0.279}$	$0.371_{\pm 0.062}$	$0.585_{\pm 0.191}$	$0.075_{\pm 0.038}$
	PLACE (+Gram)		$0.456_{\pm 0.231}$	$0.432_{\pm 0.286}$	$0.429_{\pm 0.126}$	$0.608_{\pm 0.210}$	$0.074_{\pm 0.038}$
	PLACE (+KL)		$0.454_{\pm 0.235}$	$0.425_{\pm 0.284}$	$0.424_{\pm 0.118}$	$0.602_{\pm 0.203}$	$0.074_{\pm 0.038}$

tokens of 64×64 sliding-window crops (stride 64, resized to 224×224), and optimizes via MSE and MSE+KL.

Discussion. The sliding-window baseline artificially inflates C-Ins metrics through two compounding artifacts. First, by factorizing the final CLS token immediately preceding the classifier head, the extracted concepts shortcut the predictive distribution, yielding high confidence scores that reflect architectural proximity to the logits. Second, as formally detailed in Appendix A, because each resized crop is processed independently, its CLS token aggregates context *as if the crop were a complete image*, violating the context of the original scene. The representation error gap is consistent with this mismatch: the sliding-window baseline yields an MSE of 3.489 on ViT-B/16, compared to 0.292 for PLACE, although the two values are computed on activation spaces of different scale. The final CLS token inherently aggregates highly entangled, global scene statistics, and forcing such a low-rank factorization leads to high reconstruction errors.

4.3 Semantic Alignment with Supervised Concepts

We validate the semantic quality of our unsupervised concepts by measuring their geometric alignment with the supervised "gold standard" (+KL). Because causal metrics inherently depend on the surrogate linear probe, we employ the RSVC framework [37] to compare the concept banks directly. We report Maximum Cosine Similarity (MCS) for one-to-one vector alignment, and Cross-Model Cosine Similarity i.e., Pearson correlation of Lasso/Ridge predictions (CMCS) for the functional equivalence of the overall subspace.

Table 2: Agreement of unsupervised concept banks with the supervised (+KL) reference on ImageNet, measured via RSVC (Lasso). Mean $\pm$ std over 5 runs.

Backbone	Comparison	MCS ↑	CMCS ↑
DINOv2 (ViT-B/14)	Reconstruction vs. +KL	$0.840_{\pm0.043}$	$0.732_{\pm0.040}$
	+Gram vs. +KL	$0.857_{\pm0.038}$	$0.722_{\pm0.041}$
ViT-MAE (B/16)	Reconstruction vs. +KL	$0.921_{\pm0.033}$	$0.863_{\pm0.021}$
	+Gram vs. +KL	$0.923_{\pm0.026}$	$0.856_{\pm0.021}$

Discussion. In this protocol both unsupervised variants align closely with the supervised reference (MCS ≥ 0.840, CMCS ≥ 0.722), but the differences between reconstruction and +Gram lie within one standard deviation: MCS changes from 0.840 to 0.857 on DINOv2 and from 0.921 to 0.923 on ViT-MAE, while CMCS decreases slightly on both (0.732 to 0.722 and 0.863 to 0.856). In this setting the Gram loss therefore preserves alignment with the supervised concepts. Its most consistent effect here is on dictionary stability: +Gram raises W-Stab over reconstruction on 8 of 10 ImageNet classes (e.g., `church` 0.897 to 0.953, `tench` 0.959 to 0.995; Appendix E.7), at about $20\times$ lower cost than the KL reference. CMCS relies on linear predictive agreement and thus gives a conservative estimate of concept agreement (Section 5). The task-agnostic setting of Section 4.4, where a single bank is learned from unlabeled multi-class data, provides the stronger test of the Gram loss.

4.4 Task Agnostic Experiments

To evaluate PLACE without any task information, we learn a single concept bank from unlabeled multi-class data: in each run we sample 200 images at random from the union of the evaluated classes of each dataset. Neither labels nor classifier outputs are used for data selection or by the reconstruction and Gram objectives. We increase the rank to $r = 100$ to accommodate multiple classes, and compute the +KL variant only as a reference to evaluate alignment. We run 20 seeds and compare reconstruction with +Gram using paired two-sided Wilcoxon signed-rank tests with Holm-Bonferroni correction for statistical significance.

Table 3 shows that the Gram loss *improves* matched concept alignment with the supervised reference on both backbones and datasets: MCS increases by 0.124 and 0.209 on DINOv2 and by 0.552 and 0.413 on ViT-MAE, on ImageNet and COCO respectively. CMCS improves on DINOv2 ($+0.034$ and $+0.017$), decreases on ViT-MAE/ImageNet (-0.027), and is unchanged on ViT-MAE/COCO. The Gram loss also raises dictionary stability over reconstruction in all four configurations (W-Stab from 0.813 to 0.871, 0.727 to 0.833, 0.854 to 0.881, and 0.747 to 0.865), close to the +KL reference (Appendix E.14). Compared with SAEs, PLACE (+Gram) attains higher C-Gini, by 0.06 to 0.07 on DINOv2 and 0.22 to 0.25 on ViT-MAE over the best SAE baseline, while SAEs obtain lower MSE

Table 3: Task-agnostic setting: agreement of unsupervised concept banks with the supervised (+KL) reference, measured via RSVC (Ridge). Mean $\pm$ std over 20 runs; p-values from paired two-sided Wilcoxon signed-rank tests, Holm-Bonferroni corrected.

Backbone	Dataset	Metric	Reconstruction	+Gram	Δ	p (Holm)
DINOv2	ImageNet	MCS ↑	0.262 ± 0.019	0.386 ± 0.078	$+0.124$	$< 10^{-4}$
		CMCS ↑	0.471 ± 0.014	0.505 ± 0.029	$+0.034$	0.0001
	COCO	MCS ↑	0.312 ± 0.016	0.521 ± 0.063	$+0.209$	$< 10^{-4}$
		CMCS ↑	0.348 ± 0.015	0.365 ± 0.021	$+0.017$	0.0018
ViT-MAE	ImageNet	MCS ↑	0.419 ± 0.008	0.971 ± 0.003	$+0.552$	$< 10^{-4}$
		CMCS ↑	0.781 ± 0.004	0.754 ± 0.004	-0.027	$< 10^{-4}$
	COCO	MCS ↑	0.556 ± 0.021	0.970 ± 0.012	$+0.413$	$< 10^{-4}$
		CMCS ↑	0.809 ± 0.004	0.809 ± 0.005	0.000	0.8182

in all four configurations and lower D_{KL} in three of them (Appendix E.14). On DINOv2, however, the Gram loss increases D_{KL} over reconstruction ($4.357 \to 4.903$ on ImageNet, $0.125 \to 0.326$ on COCO), as the MSE slightly rises and Proposition 1 bounds D_{KL} through both terms jointly.

5 Limitations and Future Work

While PLACE trivially extends to other ViT-based VFMs [13, 64], working on architectures with alternative tokenization schemes (Swin [47, 48], DeiT [67], JEPA [6, 52], and registers [19, 34]) is a natural next step. Our Gram-matrix alignment serves as a geometric proxy for faithfulness. We emphasize that our theoretical KL bound (Appendix C) is loose due to worst-case Lipschitz compounding and serves primarily as a mechanistic motivation rather than a strict numerical guarantee; tightening it via local Lipschitz estimates is an open direction. More broadly, complementing the Gram-matrix loss with alternative representational similarity measures [38, 53, 60] is a promising avenue. Nonetheless, in the task-agnostic setting the Gram loss improves RSVC agreement over reconstruction alone (Section 4.4). Performance on COCO is lower than on ImageNet; as demonstrated by our dataset size ablation (Appendix E.8), this is a data-driven limitation of NMF-based factorization on small sample sets (e.g., < 50 images), leading to under-determined systems, rather than a failure of the extraction pipeline. PLACE assumes that concepts are linearly factorizable in the activation space and requires pre-defining the dictionary rank r, which may not perfectly reflect the dynamic conceptual complexity of varied classes; adaptive rank selection via MDL [66] could relax this constraint. Additionally, spatial token grouping ($s > 1$) introduces spatial quantization error, as tokens within an $s \times s$ block share a reconstructed coefficient, masking intra-block variation (Appendix B). Regarding evaluation, while we assess concept agreement via the RSVC framework [37], its reliance on linear predictive models (i.e., Lasso regression) to correlate subspaces may not fully capture the highly non-linear transformations inherent to deep transformer architectures; developing non-linear semantic alignment metrics remains an open avenue for future research. Finally, the obtained concept heatmaps are evaluated qualitatively (Appendix H) rather than via human studies, and PLACE does not automatically assign textual labels to the discovered concepts.

6 Conclusion

In this work, we have proposed Patch-Level Agnostic Concept Extraction (PLACE), a label-free framework for concept extraction on VFMs and ViTs. PLACE adapts concept discovery directly to the native patch-token representations of ViTs by ❶ bypassing the distortive crop-and-resize pipelines of prior methods to preserve the exact spatial geometry; ❷ enforcing sparsity via localized Semi-NMF; and ❸ introducing a Gram-matrix alignment loss that, jointly with reconstruction, bounds the downstream predictive divergence without requiring *any* downstream classifier supervision. This way, PLACE overcomes the computational overhead of training auxiliary modules like Sparse Autoencoders (SAEs) and natively avoids resizing artifacts, delivering spatially resolved, visually interpretable concept heatmaps out of the box. Through comprehensive experiments on ImageNet and COCO using DINOv2, ViT-MAE, and ViT-B/16, we show that PLACE extracts concepts about $20\times$ faster than KL-supervised extraction and $11\times$ faster than SAEs. Furthermore, it yields sparser concepts than SAEs in four of six configurations (C-Gini up to 0.933, and up to 0.40 higher than the best SAE baseline), and extracts concepts that align with those obtained under classifier supervision (maximum cosine similarity up to 0.923). When a single concept bank is learned from unlabeled multi-class data, our Gram alignment loss improves this alignment over reconstruction alone.

Acknowledgements

This work was supported in part by U.S. National Science Foundation (NSF) under Grants CNS-2312875 and OAC-2530896, Defense Advanced Research Projects Agency (DARPA) under Cooperative Agreement D25AC00374-00, Office of Naval Research (ONR) under Award N00014-23-1-2221, and the Air Force Office of Scientific Research (AFOSR) under Award FA9550-23-1-0261.

References

[1] Radhakrishna Achanta et al. "SLIC Superpixels Compared to State-of-the-Art Superpixel Methods". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 34.11 (2012), pp. 2274–2282. DOI: 10.1109/TPAMI.2012.120.

[2] Reduan Achtibat et al. "From attribution maps to human-understandable explanations through concept relevance propagation". In: *Nature Machine Intelligence* 5.9 (2023), pp. 1006–1019. DOI: 10.1038/s42256-023-00711-8.

[3] Suhyun Ahn, Donggyu Lee, and Jinah Park. "Adaptability of Vision Foundation Models for 3D Medical Image Segmentation". In: *IEEE Open Journal of Signal Processing* 7 (2026), pp. 82–90. DOI: 10.1109/OJSP.2025.3650437.

[4] Julia Amann et al. "Explainability for artificial intelligence in healthcare: a multidisciplinary perspective". In: *BMC Medical Informatics and Decision Making* 20.1 (2020), p. 310. ISSN: 1472-6947. DOI: 10.1186/s12911-020-01332-6.

[5] Shreyash Arya et al. "B-cosification: Transforming deep neural networks to be inherently interpretable". In: *Advances in Neural Information Processing Systems* 37 (2024), pp. 62756–62786. DOI: 10.52202/079017-2007.

[6] Mahmoud Assran et al. "Self-Supervised Learning from Images with a Joint-Embedding Predictive Architecture". In: *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2023, pp. 15619–15629. DOI: 10.1109/CVPR52729.2023.01499.

[7] Dipkamal Bhusal et al. "FACE: Faithful Automatic Concept Extraction". In: *The Thirty-ninth Annual Conference on Neural Information Processing Systems*. 2025. DOI: 10.52202/085713-1462.

[8] Irving Biederman. "Recognition-by-components: A theory of human image understanding". In: *Psychological Review* 94.2 (Apr. 1987), pp. 115–147. DOI: 10.1037/0033-295X.94.2.115.

[9] Moritz Böhle, Mario Fritz, and Bernt Schiele. "B-cos Networks: Alignment is All We Need for Interpretability". In: *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2022, pp. 10319–10328. DOI: 10.1109/CVPR52688.2022.01008.

[10] Moritz Böhle, Mario Fritz, and Bernt Schiele. "Holistically explainable vision transformers". In: *arXiv preprint arXiv:2301.08669* (2023). DOI: 10.48550/arXiv.2301.08669.

[11] Moritz Böhle et al. "B-Cos Alignment for Inherently Interpretable CNNs and Vision Transformers". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 46.6 (2024), pp. 4504–4518. DOI: 10.1109/TPAMI.2024.3355155.

[12] C. Boutsidis and E. Gallopoulos. "SVD based initialization: A head start for nonnegative matrix factorization". In: *Pattern Recognition* 41.4 (2008), pp. 1350–1362. ISSN: 0031-3203. DOI: https://doi.org/10.1016/j.patcog.2007.09.010.

[13] Mathilde Caron et al. "Emerging Properties in Self-Supervised Vision Transformers". In: *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*. 2021, pp. 9630–9640. DOI: 10.1109/ICCV48922.2021.00951.

[14] Hila Chefer, Shir Gur, and Lior Wolf. "Transformer Interpretability Beyond Attention Visualization". In: *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2021, pp. 782–791. DOI: 10.1109/CVPR46437.2021.00084.

[15] Zhiqing Chen et al. "SSG–CAM: enhancing visual interpretability through refined second-order gradients and evolutionary multi-layer fusion". In: *Scientific Reports* 16 (2026), p. 6848. DOI: 10.1038/s41598-026-37278-4.

[16] Taco Cohen and Max Welling. "Group Equivariant Convolutional Networks". In: *Proceedings of The 33rd International Conference on Machine Learning*. Ed. by Maria Florina Balcan and Kilian Q. Weinberger. Vol. 48. Proceedings of Machine Learning Research. New York, New York, USA: PMLR, 2016, pp. 2990–2999. DOI: 10.5555/3045390.3045705.

[17] Jean-Baptiste Cordonnier, Andreas Loukas, and Martin Jaggi. "On the Relationship between Self-Attention and Convolutional Layers". In: *International Conference on Learning Representations*. 2020. DOI: 10.48550/arXiv.1911.03584.

[18] Ian C. Covert, Scott Lundberg, and Su-In Lee. "Explaining by removing: a unified framework for model explanation". In: *Journal of Machine Learning Research* 22.1 (Jan. 2021). ISSN: 1532-4435. DOI: 10.5555/3546258.3546467.

[19] Timothée Darcet et al. "Vision Transformers Need Registers". In: *The Twelfth International Conference on Learning Representations*. 2024. DOI: 10.48550/arXiv.2309.16588.

[20] Jia Deng et al. "ImageNet: A large-scale hierarchical image database". In: *2009 IEEE Conference on Computer Vision and Pattern Recognition*. 2009, pp. 248–255. DOI: 10.1109/CVPR.2009.5206848.

[21] Chris H.Q. Ding, Tao Li, and Michael I. Jordan. "Convex and Semi-Nonnegative Matrix Factorizations". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 32.1 (2010), pp. 45–55. DOI: 10.1109/TPAMI.2008.277.

[22] Teresa Dorszewski et al. "From Colors to Classes: Emergence of Concepts in Vision Transformers". In: *Explainable Artificial Intelligence*. Ed. by Riccardo Guidotti, Ute Schmid, and Luca Longo. Cham: Springer Nature Switzerland, 2026, pp. 28–47. ISBN: 978-3-032-08317-3. DOI: 10.1007/978-3-032-08317-3_2.

[23] Alexey Dosovitskiy et al. "An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale". In: *International Conference on Learning Representations*. 2021. DOI: 10.48550/arXiv.2010.11929.

[24] Thomas Fel et al. "A Holistic Approach to Unifying Automatic Concept Extraction and Concept Importance Estimation". In: *Thirty-seventh Conference on Neural Information Processing Systems*. 2023. DOI: 10.52202/075280-2391.

[25] Thomas Fel et al. "CRAFT: Concept Recursive Activation FacTorization for Explainability". In: *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2023, pp. 2711–2721. DOI: 10.1109/CVPR52729.2023.00266.

[26] Amirata Ghorbani et al. "Towards automatic concept-based explanations". In: *Proceedings of the 33rd International Conference on Neural Information Processing Systems*. Red Hook, NY, USA: Curran Associates Inc., 2019. DOI: 10.5555/3454287.3455119.

[27] Kai Han et al. "A Survey on Vision Transformer". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 45.1 (2023), pp. 87–110. DOI: 10.1109/TPAMI.2022.3152247.

[28] Peter Hase, Harry Xie, and Mohit Bansal. "The out-of-distribution problem in explainability and search methods for feature importance explanations". In: *Proceedings of the 35th International Conference on Neural Information Processing Systems*. NIPS '21. Red Hook, NY, USA: Curran Associates Inc., 2021. ISBN: 9781713845393. DOI: 10.5555/3540261.3540540.

[29] Kaiming He et al. "Deep Residual Learning for Image Recognition". In: *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2016, pp. 770–778. DOI: 10.1109/CVPR.2016.90.

[30] Kaiming He et al. "Masked Autoencoders Are Scalable Vision Learners". In: *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2022, pp. 15979–15988. DOI: 10.1109/CVPR52688.2022.01553.

[31] Dan Hendrycks and Thomas Dietterich. "Benchmarking Neural Network Robustness to Common Corruptions and Perturbations". In: *International Conference on Learning Representations*. 2019. DOI: 10.48550/arXiv.1903.12261.

[32] Qihan Huang et al. "LG-CAV: Train any concept activation vector with language guidance". In: *Advances in Neural Information Processing Systems* 37 (2024), pp. 39522–39551. DOI: 10.52202/079017-1249.

[33] Robert Huben et al. "Sparse Autoencoders Find Highly Interpretable Features in Language Models". In: *The Twelfth International Conference on Learning Representations*. 2024. DOI: 10.48550/arXiv.2309.08600.

[34] Nicholas Jiang et al. "Vision Transformers Don't Need Trained Registers". In: *The Thirty-ninth Annual Conference on Neural Information Processing Systems*. 2025. DOI: 10.52202/085713-1891.

[35] Been Kim et al. "Interpretability Beyond Feature Attribution: Quantitative Testing with Concept Activation Vectors (TCAV)". In: *Proceedings of the 35th International Conference on Machine Learning*. PMLR, 2018, pp. 2673–2682. DOI: 10.48550/arXiv.1711.11279.

[36] Pang Wei Koh et al. “Concept bottleneck models”. In: *International conference on machine learning*. PMLR. 2020, pp. 5338–5348. DOI: 10.5555/3524938.3525433.

[37] Neehar Kondapaneni, Oisin Mac Aodha, and Pietro Perona. “Representational Similarity via Interpretable Visual Concepts”. In: *The Thirteenth International Conference on Learning Representations*. 2025. DOI: 10.48550/arXiv.2503.15699.

[38] Simon Kornblith et al. “Similarity of Neural Network Representations Revisited”. In: *Proceedings of the 36th International Conference on Machine Learning*. Ed. by Kamalika Chaudhuri and Ruslan Salakhutdinov. Vol. 97. Proceedings of Machine Learning Research. PMLR, 2019, pp. 3519–3529. DOI: 10.48550/arXiv.1905.00414.

[39] Ananya Kumar et al. “Fine-Tuning can Distort Pretrained Features and Underperform Out-of-Distribution”. In: *International Conference on Learning Representations*. 2022. DOI: 10.48550/arXiv.2202.10054.

[40] Michihiro Kuroki and Toshihiko Yamasaki. “CE-FAM: Concept-Based Explanation via Fusion of Activation Maps”. In: *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*. 2025, pp. 1413–1422. DOI: 10.1109/ICCV51701.2025.00139.

[41] Victor Lecomte et al. “What Causes Polysemanticity? An Alternative Origin Story of Mixed Selectivity from Incidental Causes”. In: *ICLR 2024 Workshop on Representational Alignment*. 2024. DOI: 10.48550/arXiv.2312.03096.

[42] Daniel D Lee and H Sebastian Seung. “Learning the parts of objects by non-negative matrix factorization”. In: *Nature* 401.6755 (1999), pp. 788–791. DOI: 10.1038/44565.

[43] Yifan Li et al. “Evaluating Object Hallucination in Large Vision-Language Models”. In: *The 2023 Conference on Empirical Methods in Natural Language Processing*. 2023. DOI: 10.18653/v1/2023.emnlp-main.20.

[44] Pengchen Liang et al. “Vision Foundation Models in Medical Image Analysis: Advances and Challenges”. In: *Blockchain and Trustworthy Systems*. Ed. by Jianguo Chen, Xiaonan Luo, and Yuanlong Yu. Vol. 2638. Singapore: Springer Nature Singapore, 2026, pp. 170–181. ISBN: 978-981-95-3483-8. DOI: 10.1007/978-981-95-3483-8_13.

[45] Tsung-Yi Lin et al. “Microsoft COCO: Common Objects in Context”. In: *Computer Vision – ECCV 2014*. Ed. by David Fleet et al. Vol. 8693. Cham: Springer International Publishing, 2014, pp. 740–755. ISBN: 978-3-319-10602-1. DOI: 10.1007/978-3-319-10602-1_48.

[46] Zachary C. Lipton. “The mythos of model interpretability”. In: *Commun. ACM* 61.10 (Sept. 2018), pp. 36–43. ISSN: 0001-0782. DOI: 10.1145/3233231.

[47] Ze Liu et al. “Swin Transformer V2: Scaling Up Capacity and Resolution”. In: *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2022, pp. 11999–12009. DOI: 10.1109/CVPR52688.2022.01170.

[48] Ze Liu et al. “Swin Transformer: Hierarchical Vision Transformer using Shifted Windows”. In: *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*. 2021, pp. 9992–10002. DOI: 10.1109/ICCV48922.2021.00986.

[49] Ilya Loshchilov and Frank Hutter. “Decoupled Weight Decay Regularization”. In: *International Conference on Learning Representations*. 2019. DOI: 10.48550/arXiv.1711.05101.

[50] Chiyu Ma et al. “Interpretable image classification with adaptive prototype-based vision transformers”. In: *Advances in Neural Information Processing Systems* 37 (2024), pp. 41447–41493. DOI: 10.5555/3737916.3739228.

[51] Vien V. Mai and Mikael Johansson. “Stability and Convergence of Stochastic Gradient Clipping: Beyond Lipschitz Continuity and Smoothness”. In: *Proceedings of the 38th International Conference on Machine Learning*. Ed. by Marina Meila and Tong Zhang. Vol. 139. Proceedings of Machine Learning Research. PMLR, 2021, pp. 7325–7335. DOI: 10.48550/arXiv.2102.06489.

[52] Shentong Mo and Shengbang Tong. “Connecting Joint-Embedding Predictive Architecture with Contrastive Self-supervised Learning”. In: *The Thirty-eighth Annual Conference on Neural Information Processing Systems*. 2024. DOI: 10.5555/3737916.3737993.

[53] Alex Graeme Murphy, Joel Zylberberg, and Alona Fyshe. “Correcting Biased Centered Kernel Alignment Measures in Biological and Artificial Neural Networks”. In: *ICLR 2024 Workshop on Representational Alignment*. 2024. DOI: 10.48550/arXiv.2405.01012.

[54] Muzammal Naseer et al. “Intriguing properties of vision transformers”. In: *Proceedings of the 35th International Conference on Neural Information Processing Systems*. NIPS ’21. Red Hook, NY, USA: Curran Associates Inc., 2021. ISBN: 9781713845393. DOI: 10.5555/3540261.3542045.

[55] Tuomas Oikarinen and Tsui-Wei Weng. “CLIP-Dissect: Automatic Description of Neuron Representations in Deep Vision Networks”. In: *The Eleventh International Conference on Learning Representations*. 2023. DOI: https://doi.org/10.48550/arXiv.2204.10965.

[56] Maxime Oquab et al. “DINOv2: Learning Robust Visual Features without Supervision”. In: *Transactions on Machine Learning Research* (2024). Featured Certification. ISSN: 2835-8856.

[57] Mateusz Pach et al. “Sparse Autoencoders Learn Monosemantic Features in Vision-Language Models”. In: *The Thirty-ninth Annual Conference on Neural Information Processing Systems*. 2025. DOI: 10.52202/085713-3202.

[58] Kiho Park, Yo Joong Choe, and Victor Veitch. “The linear representation hypothesis and the geometry of large language models”. In: *Proceedings of the 41st International Conference on Machine Learning*. ICML’24. Vienna, Austria: JMLR.org, 2024. DOI: 10.5555/3692070.3693675.

[59] Maithra Raghu et al. “Do vision transformers see like convolutional neural networks?” In: *Proceedings of the 35th International Conference on Neural Information Processing Systems*. NIPS ’21. Red Hook, NY, USA: Curran Associates Inc., 2021. ISBN: 9781713845393. DOI: 10.5555/3540261.3541188.

[60] Maithra Raghu et al. “SVCCA: singular vector canonical correlation analysis for deep learning dynamics and interpretability”. In: *Proceedings of the 31st International Conference on Neural Information Processing Systems*. NIPS’17. Long Beach, California, USA: Curran Associates Inc., 2017, pp. 6078–6087. ISBN: 9781510860964. DOI: 10.5555/3295222.3295356.

[61] Senthooran Rajamanoharan et al. “Improving sparse decomposition of language model activations with gated sparse autoencoders”. In: *Proceedings of the 38th International Conference on Neural Information Processing Systems*. NIPS ’24. Vancouver, BC, Canada: Curran Associates Inc., 2024. ISBN: 9798331314385. DOI: 10.5555/3737916.3737940.

[62] Cynthia Rudin. “Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead”. In: *Nature Machine Intelligence* 1.5 (May 2019), pp. 206–215. ISSN: 2522-5839. DOI: 10.1038/s42256-019-0048-x.

[63] Olga Russakovsky et al. “ImageNet Large Scale Visual Recognition Challenge”. In: *International Journal of Computer Vision* 115.3 (Dec. 2015), pp. 211–252. ISSN: 1573-1405. DOI: 10.1007/s11263-015-0816-y.

[64] Oriane Siméoni et al. “DINOv3”. In: *Transactions on Machine Learning Research* (2026). Featured Certification. ISSN: 2835-8856. DOI: 10.48550/arXiv.2508.10104.

[65] Ilya M Sobol. “Sensitivity estimates for nonlinear mathematical models”. In: *Mathematical Modelling and Computational Experiments* 1.4 (1993), pp. 407–414.

[66] Vincent Y.F. Tan and Cédric Févotte. “Automatic Relevance Determination in Nonnegative Matrix Factorization with the /spl beta/-Divergence”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 35.7 (2013), pp. 1592–1605. DOI: 10.1109/TPAMI.2012.240.

[67] Hugo Touvron et al. “Training data-efficient image transformers; distillation through attention”. In: *Proceedings of the 38th International Conference on Machine Learning*. Ed. by Marina Meila and Tong Zhang. Vol. 139. Proceedings of Machine Learning Research. PMLR, 2021, pp. 10347–10357. DOI: 10.48550/arXiv.2012.12877.

[68] Shikhar Tuli et al. “Are Convolutional Neural Networks or Transformers more like human vision?” In: *43rd Annual Meeting of the Cognitive Science Society: Comparative Cognition: Animal Minds, CogSci 2021*. 2021.

[69] Mateo Espinosa Zarlenga et al. “Concept Embedding Models”. In: *Advances in Neural Information Processing Systems*. Ed. by Alice H. Oh et al. 2022. DOI: 10.5555/3600270.3601825.

[70] Richard Zhang. “Making Convolutional Networks Shift-Invariant Again”. In: *Proceedings of the 36th International Conference on Machine Learning*. Ed. by Kamalika Chaudhuri and Ruslan Salakhutdinov. Vol. 97. Proceedings of Machine Learning Research. PMLR, 2019, pp. 7324–7334. DOI: 10.48550/arXiv.1904.11486.

[71] Ruihan Zhang et al. “Invertible concept-based explanations for cnn models with non-negative concept activation vectors”. In: *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 35. 13. 2021, pp. 11682–11690. DOI: 10.1609/aaai.v35i13.17389.

PLACE: Patch-Level Agnostic Concept Extraction

Supplementary Material

Contents

A Why crop-and-resize concept extraction is sub-optimal for Vision Transformers **17**

A.1 Translation equivariance in CNNs 17

A.2 Why crop-and-resize exploits equivariance 18

A.3 Vision Transformers are not translation equivariant 18

A.4 What goes wrong with crop-and-resize in ViTs 19

A.5 Empirical evidence 19

A.6 Our approach: factorizing patch tokens in place 20

B Token grouping: detailed spatial arithmetic **20**

B.1 From pixels to patch tokens 20

B.2 Token grouping and resolution 20

B.3 Per-architecture spatial resolution 21

C Connecting Gram-matrix alignment to predictive consistency **21**

C.1 Preliminary: Gram deviation under reconstruction error 21

C.2 Step 1: Attention logit deviation 22

C.3 Step 2: Single-block error propagation 23

C.4 Step 3: Composition through suffix blocks 24

C.5 Step 4: From CLS-token deviation to KL divergence 24

C.6 Discussion 25

D Implementation Details **26**

D.1 Linear probes for validation 26

D.2 PLACE Hyperparameters 26

D.3 PCA Visualization Details 27

E Additional Analyses **27**

E.1 NNDSVD subsampling ratio for Semi-NMF initialization 27

E.2 Faithfulness loss weight sensitivity 28

E.3 Extraction layer ablation 29

E.4 Decomposition rank ablation 30

E.5 Token group size ablation 30

E.6 Sanity check: weight randomization 32

E.7 Stability across seeds 32

E.8 Effect of Dataset Size on Concept Stability 33

E.9 Runtime and memory 34
E.10 Computational Complexity and Scalability to Larger Architectures 34
E.11 Per-class faithfulness trends 36
E.12 Per-class agreement scores 39
E.13 Robustness to Image Perturbations 39
E.14 Task-agnostic setting: full results 40
E.15 Statistical significance of the benchmark comparison 40

F Evaluation Metrics Description **40**

G Broader Impact **43**

H Additional Qualitative Results **43**

A Why crop-and-resize concept extraction is sub-optimal for Vision Transformers

In this appendix we formalise the argument, sketched in Section 3, that the crop-and-resize concept-proposal pipeline used by CRAFT [25] and RSVC [37] is well-motivated for CNNs but is sub-optimal for Vision Transformers. The root cause is a structural property of convolutional networks, translation equivariance, that ViTs do not possess.

A.1 Translation equivariance in CNNs

Let $T_{\boldsymbol{\tau}}$ denote the operator that shifts an image $\mathbf{x} \in \mathbb{R}^{3\times H\times W}$ by a displacement $\boldsymbol{\tau} = (\tau_h, \tau_w)$:

$$\big[T_{\boldsymbol{\tau}}\,\mathbf{x}\big](c, i, j) = \mathbf{x}(c,\ i-\tau_h,\ j-\tau_w)\,. \tag{4}$$

A mapping $\phi : \mathbb{R}^{c_{\text{in}}\times H\times W} \to \mathbb{R}^{c_{\text{out}}\times H'\times W'}$ is *translation equivariant* if there exists a (possibly stride-adjusted) translation operator $T'_{\boldsymbol{\tau}}$ on the output space such that

$$\phi\big(T_{\boldsymbol{\tau}}\,\mathbf{x}\big) = T'_{\boldsymbol{\tau}}\,\phi(\mathbf{x}) \qquad \forall\,\mathbf{x},\ \boldsymbol{\tau}\,. \tag{5}$$

Convolution is equivariant. A convolutional layer with kernel $\mathbf{w} \in \mathbb{R}^{c_{\text{out}}\times c_{\text{in}}\times k\times k}$ computes

$$[\phi(\mathbf{x})](m, i, j) = \sum_{c=1}^{c_{\text{in}}} \sum_{p,q=0}^{k-1} w_{m,c,p,q}\ \mathbf{x}(c,\ i{+}p,\ j{+}q)\,. \tag{6}$$

Evaluating on the translated input gives

$$\begin{aligned}
[\phi(T_{\boldsymbol{\tau}}\,\mathbf{x})](m, i, j) &= \sum_{c,p,q} w_{m,c,p,q}\ [T_{\boldsymbol{\tau}}\,\mathbf{x}](c,\ i{+}p,\ j{+}q) \\
&= \sum_{c,p,q} w_{m,c,p,q}\ \mathbf{x}(c,\ (i{-}\tau_h){+}p,\ (j{-}\tau_w){+}q) \\
&= [\phi(\mathbf{x})](m,\ i{-}\tau_h,\ j{-}\tau_w) \\
&= [T_{\boldsymbol{\tau}}\,\phi(\mathbf{x})](m, i, j)\,,
\end{aligned} \tag{7}$$

where the third equality follows because the kernel weights $w_{m,c,p,q}$ are *shared across all spatial positions* and do not depend on (i, j). This is the key structural property: the same filter is applied everywhere, so shifting the input shifts the output by exactly the same amount. Pointwise nonlinearities (ReLU, GELU, etc.) act independently at each spatial position and therefore commute with translation trivially.

Spatial pooling: equivariance and local invariance. Average or max pooling with a fixed spatial grid of size $s \times s$ maps each non-overlapping $s \times s$ block to a single value. This operation is equivariant only to translations that are multiples of the stride s: shifting the input by s pixels shifts the pooled output by exactly one position. For shifts smaller than s, the pooled output remains unchanged, introducing *local translation invariance*: content that moves within the same pooling window produces identical activations. Crucially, even with pooling, each output position still depends on a bounded local neighborhood of the input.

Consequence: local receptive fields. By composing equivariant convolutions, pointwise nonlinearities, and local pooling, the feature map at any layer ℓ of a CNN satisfies

$$f^{(\ell)}_{\text{cnn}}\big(T_{\boldsymbol{\tau}}\,\mathbf{x}\big)\ =\ T'_{\boldsymbol{\tau}}\ f^{(\ell)}_{\text{cnn}}(\mathbf{x}) \tag{8}$$

up to boundary padding artifacts. This means that each spatial position (i, j) in $\mathbf{A} = f^{(\ell)}_{\text{cnn}}(\mathbf{x}) \in \mathbb{R}^{c\times h\times w}$ depends only on a bounded neighborhood of (i, j) in the input, the *receptive field* $\mathcal{R}^{(\ell)}_{i,j}$:

$$\mathbf{A}[:,\ i,\ j] = g\Big(\mathbf{x}\big|_{\mathcal{R}^{(\ell)}_{i,j}}\Big)\,. \tag{9}$$

A.2 Why crop-and-resize exploits equivariance

In CRAFT [25], FACE [7] and RSVC [37], a crop $\mathbf{x}_p$ from the spatial window $[i_0 : i_1,\ j_0 : j_1]$ of $\mathbf{x}$ is *resized* to the full input resolution $H \times W$ and passed through the network independently. CRAFT explicitly defines a "crop and resize function $\pi(\cdot)$" and justifies the resize by arguing that modern CNNs are trained with aggressive crop augmentation (RandomResizedCrop, CutMix, etc.), so resized crops lie within the training distribution.

For CNNs, this argument is reinforced by equivariance: thanks to the local receptive field property (Eq. (9)), the feature map of the resized crop satisfies

$$f_{\text{cnn}}^{(\ell)}\big(\texttt{resize}(\mathbf{x}_p)\big) \;\approx\; \mathbf{A}[:,\ i'_0 : i'_1,\ j'_0 : j'_1\,], \tag{10}$$

where $[i'_0 : i'_1,\ j'_0 : j'_1]$ is the corresponding region in the feature map. The approximation is imperfect because resizing changes the effective scale and boundary padding introduces edge effects, but the core property holds: the activation of the crop describes the *local visual content* of that image region. After global average pooling, each crop yields a vector $\bar{\mathbf{a}}_p \in \mathbb{R}^c$ that summarizes local content. Collecting these vectors into $\mathbf{A} \in \mathbb{R}^{n\times c}$ and factorizing via NMF discovers spatially localized concepts, which is the intended behavior.

A.3 Vision Transformers are not translation equivariant

A ViT [23] tokenizes the input into $N = (H/P)^2$ non-overlapping patches of size $P \times P$, linearly projects each patch into $\mathbb{R}^d$, prepends a CLS token, adds learnable positional embeddings, and processes the sequence through L transformer blocks. While Cordonnier, Loukas, and Jaggi [17] prove that self-attention layers *can* implement convolutional filters, they are not constrained to do so: the mapping is strictly more general and imposes no spatial locality by default. We show below that the ViT architecture violates translation equivariance at multiple stages.

1. Positional embeddings break equivariance. Let $\mathbf{e}_i^{(0)} = \mathbf{W}_{\text{proj}}\,\mathbf{x}_i + \mathbf{p}_i$ be the initial embedding of patch i, where $\mathbf{p}_i \in \mathbb{R}^d$ is the positional embedding. Translating the image by one patch width permutes the patch content ($\mathbf{x}_i \to \mathbf{x}_{\sigma(i)}$ for some permutation σ), but the positional embeddings remain tied to their grid positions:

$$\mathbf{e}_i^{(0)}\big(T_{\boldsymbol{\tau}}\,\mathbf{x}\big) = \mathbf{W}_{\text{proj}}\,\mathbf{x}_{\sigma(i)} + \mathbf{p}_i \;\neq\; \mathbf{W}_{\text{proj}}\,\mathbf{x}_{\sigma(i)} + \mathbf{p}_{\sigma(i)}\,. \tag{11}$$

The mismatch between content (which has shifted) and position (which has not) breaks equivariance before the first transformer block is even applied. This stands in direct contrast to CNNs, where spatial weight sharing guarantees that the same filter response is produced regardless of position [16].

2. Self-attention is global and content-dependent. Multi-head self-attention computes, for each token i, a weighted sum over *all* tokens:

$$\text{Attn}(\mathbf{Q},\mathbf{K},\mathbf{V})_i = \sum_{j=0}^{N} \frac{\exp\!\big(\mathbf{q}_i^\top \mathbf{k}_j/\sqrt{d_h}\big)}{\sum_{m=0}^{N}\exp\!\big(\mathbf{q}_i^\top \mathbf{k}_m/\sqrt{d_h}\big)}\;\mathbf{v}_j\,. \tag{12}$$

The attention weights a_{ij} depend on the *content* of all tokens through the queries and keys, whereas convolutional kernel weights are fixed parameters independent of the input (Eq. (6)). Two consequences follow. First, every token's representation after layer 1 is a function of all input patches: there is no bounded receptive field and no analogue of Eq. (9). Second, the attention pattern itself changes when the input is translated, because the queries and keys change. Even without positional embeddings, the softmax normalization couples all tokens, making the mapping globally contextual.

3. Empirical confirmation. The theoretical absence of equivariance has measurable consequences on how ViTs process visual information [27]. Raghu et al. [59] provide a systematic comparison of ViT and CNN internal representations, finding that (i) ViTs aggregate global spatial information from the earliest layers, whereas CNNs build local-to-global hierarchies gradually, (ii) ViT representations are more uniform across layers, with lower-layer features already encoding high-level structure, and (iii) spatial information is preserved throughout ViT layers but in a qualitatively different manner from CNNs, with self-attention enabling long-range feature propagation via residual connections. Complementing these representational findings, Tuli et al. [68] analyze the behavioral consequences of these architectural differences. They show that ViTs, precisely because they relax the translation-invariance constraint of CNNs, develop recognition strategies whose error patterns are more consistent

with human vision. This suggests that ViTs learn fundamentally different visual features, relying on global structure and shape rather than local texture. Naseer et al. [54] further corroborate this by demonstrating that ViTs exhibit markedly different robustness properties from CNNs: they are more resilient to severe occlusions, texture corruption, and adversarial perturbations, which is consistent with a recognition strategy that does not depend on local, spatially equivariant feature detection.

Taken together, these findings confirm that ViTs do not merely "relax" translation equivariance as a minor architectural choice: they leverage its absence to develop qualitatively different representations. Any concept extraction method that assumes CNN-like locality, whether through crop-and-resize proposals or spatial-slice approximations, is therefore methodologically mismatched with the ViT computation.

A.4 What goes wrong with crop-and-resize in ViTs

Let $\mathbf{x}$ be a full image ($H \times W$ pixels) and $\mathbf{x}_p$ a crop from region $[i_0 : i_1,\ j_0 : j_1]$, resized to $H \times W$ before being fed to the ViT. We identify four distinct failure modes.

1. The patch grid tiles different content at a different scale. When the full image is tokenised with patch size $P{=}16$, each of the $N{=}196$ patches covers a $16{\times}16$ pixel region of the scene. When a $64{\times}64$ crop is resized to $224{\times}224$ and tokenised with the same $P{=}16$, each patch now covers a $\sim 4.6{\times}4.6$ pixel region of the original crop. The visual content within each patch is at a completely different scale and resolution: fine texture that would fall within a single patch of the full image is now spread across multiple patches of the resized crop. The linear patch projection $\mathbf{W}_{\text{proj}}$ was not trained to process content at this magnification. While CRAFT argues that CNNs are trained with crop augmentation so resized crops are in-distribution, this argument is weaker for ViTs: the *patch-level* statistics of a heavily upsampled small crop differ from those of the random crops seen during training (which are typically $\geq 60\%$ of the image area for standard `RandomResizedCrop`).

2. Positional embeddings encode wrong locations. Patch i in the resized crop receives positional embedding $\mathbf{p}_i$, which the network learned to associate with a particular absolute position in the $14{\times}14$ grid. But the visual content now at position i of the crop occupied a completely different position in the original image. A patch of sky that sat in the top-left corner of the full image might land at the center of the resized crop, receiving a positional embedding the network associates with a different semantic context. This inconsistency has no CNN analogue: convolutional filters are spatially shared and carry no positional information.

3. Attention context is entirely different. In the full image, a patch token at position i attends to tokens covering the entire scene (background, foreground, other objects). In the resized crop, the same positional slot attends only to tokens originating from the crop region. The keys and values are functions of different input content, so

$$a_{ij}^{\text{crop}} \neq a_{ij}^{\text{full}} \qquad \forall\, l \geq 1, \quad \forall\, i, j\,. \tag{13}$$

The CLS token of the resized crop therefore summarizes the crop as if it were the entire world, incorporating attention patterns, LayerNorm statistics, and residual stream content determined exclusively by the crop.

4. No spatial correspondence to the full-image representation. In a CNN, Eq. (10) establishes an approximate correspondence between the crop activation and a spatial slice of the full-image feature map, grounded in equivariance. For the ViT CLS token, no such correspondence exists. The CLS token of a resized crop does not approximate any sub-vector, spatial slice, or linear projection of the full-image CLS token $\mathbf{z}_0^{(L)}(\mathbf{x})$. It is a fundamentally different representation of a fundamentally different input.

A.5 Empirical evidence

Kondapaneni, Aodha, and Perona [37] report (their Table A2) reconstruction errors after Semi-NMF concept extraction that are 5–8$\times$ higher for ViTs than for ResNets on the same ImageNet classes (e.g. 926.9 for ViT-S vs. 176.2 for ResNet-18 and 205.5 for ResNet-50, with $k{=}10$ concepts). While part of this gap reflects higher activation dimensionality ($d{=}384$ for ViT-S vs. $d{=}512$ for ResNet-18), the analysis above identifies a systematic, irreducible source of additional variance: two crops from the same image that share identical local content but differ in their remaining visible area produce

different CLS vectors, because global attention contextualizes each token by the entire input. This context-dependent variation inflates the effective rank of the activation matrix $\mathbf{A}$, making low-rank factorization inherently less faithful. RSVC does not investigate whether the extraction strategy itself (CLS tokens from resized crops) is appropriate for ViTs, treating the higher reconstruction error as a limitation of the decomposition method rather than a consequence of the extraction pipeline.

A.6 Our approach: factorizing patch tokens in place

Rather than extracting crops, resizing them, and factorizing CLS tokens, our method operates on the *patch-token activations* of the full image at an intermediate layer ℓ. For a single image, the prefix $g_{\leq \ell}$ produces N patch tokens $\mathbf{z}_1^{(\ell)}, \ldots, \mathbf{z}_N^{(\ell)} \in \mathbb{R}^d$, each anchored to a fixed position in the $\sqrt{N} \times \sqrt{N}$ grid. Collecting patch tokens across n images gives $\mathbf{A} \in \mathbb{R}^{nN \times d}$, which we factorize via Semi-NMF (Eq. 1).

This avoids all four failure modes:

1. **No artificial re-tokenization.** Each patch token is computed by the network in the context of the full image, at the scale and resolution the model was trained on. The patch projection $\mathbf{W}_{\text{proj}}$ operates on content at its native scale.
2. **Correct positional encoding.** Each token receives the positional embedding corresponding to its actual location in the image, consistent with training.
3. **Consistent attention context.** All patch tokens within an image share the same global attention context (the full image), so differences between their concept coefficients reflect genuine spatial variation in how concepts are activated across the scene, rather than artifacts of crop composition.
4. **Spatial anchoring is preserved.** Although self-attention mixes information globally, each token maintains its spatial identity. Each row of $\mathbf{U}$ corresponds to the token anchored at a specific grid position. Concept heatmaps are thus obtained natively by reshaping the N coefficients per image into a $\sqrt{N} \times \sqrt{N}$ grid and upscaling via bicubic interpolation, without requiring gradient-based attribution or implicit differentiation [25].

Each patch token at layer $\ell \geq 1$ is no longer a purely local descriptor due to self-attention. Consequently, the discovered concepts reflect a mixture of local visual evidence and global context. However, this is a reflection of how the ViT actually represents information, organically grounded in the token grid, rather than an artifact of an extraction pipeline designed for a different architecture.

B Token grouping: detailed spatial arithmetic

The spatial resolution of discovered concepts depends on the interaction between three quantities: the model's patch size P (in pixels), the input image resolution $H \times W$, and the token group size s used during factorisation. In this appendix, we make these relationships explicit for each architecture considered in our experiments.

B.1 From pixels to patch tokens

A ViT with patch size P and input resolution $H \times W$ produces a grid of

$$N = \frac{H}{P} \times \frac{W}{P} \tag{14}$$

non-overlapping patch tokens. Each token $\mathbf{z}_i \in \mathbb{R}^d$ corresponds to a $P \times P$ pixel region of the input. For a square input ($H = W$), the token grid has spatial dimensions $G_0 \times G_0$ where $G_0 = \sqrt{N}$. As detailed in Table 4, a standard ViT-B/16 at 224×224 resolution yields a 14×14 grid, whereas DINOv2 at 518×518 yields a 37×37 grid.

B.2 Token grouping and resolution

Given a token grid of size $G_0 \times G_0$, we apply non-overlapping average pooling with kernel and stride equal to s, producing a grouped grid of size $G_s = \lfloor G_0/s \rfloor$. Each grouped token is the average of

$s \times s$ adjacent patch tokens and therefore corresponds to a spatial region of $sP \times sP$ pixels in the input image.

After factorization, the grouped coefficient matrix $\mathbf{U}$ is mapped back to the original token resolution via `repeat_interleave`, which replicates each grouped coefficient across the $s \times s$ tokens it represents. This effectively quantizes the spatial resolution of the concept heatmap to $sP \times sP$ pixels. We note that when G_0 is not perfectly divisible by s (as occurs with DINOv2 where $G_0 = 37$ is prime), the floor operation in G_s implies that a small portion of the original tokens at the right and bottom boundaries is excluded from the grouping. Consequently, when applying `repeat_interleave`, the resulting $sG_s \times sG_s$ tensor is strictly smaller than the original $G_0 \times G_0$ activation map. To resolve this shape mismatch during optimization, we compute all losses ($\mathcal{L}_{\text{rec}}$, $\mathcal{L}_{\text{gram}}$) by cropping the original activation grid $\mathbf{A}$ to match the truncated $sG_s \times sG_s$ dimensions, explicitly discarding these marginal boundary tokens.

B.3 Per-architecture spatial resolution

Table 4 reports the effective spatial resolution for each architecture. While for $P = 16$ models we choose s to be divisors of $G_0 = 14$ to ensure perfect coverage, for DINOv2 a marginal approximation error is introduced for $s > 1$, which is quantified in Appendix E.5.

Table 4: Effective spatial resolution of concept heatmaps. P: patch size; H: input resolution; G_0: native grid dimension (H/P); s: grouping factor; G_s: grouped grid dimension; sP: pixel resolution.

Architecture	P	H	G_0	s	G_s	sP (px)
ViT-B/16	16	224	14	1	14	16
(ImageNet-1k / MAE)				2	7	32
				7	2	112
				14	1	224
DINOv2 ViT-B/14	14	518	37	1	37	14
				2	18	28
				7	5	98
				37	1	518

C Connecting Gram-matrix alignment to predictive consistency

We formalise the claim, stated informally in Section 3.1, that jointly minimising the reconstruction loss $\mathcal{L}_{\text{rec}}$ and the Gram-matrix alignment loss $\mathcal{L}_{\text{gram}}$ controls the KL divergence through a downstream linear classifier without ever computing it. The argument proceeds in five steps: we first establish a preliminary relationship between reconstruction error and Gram deviation (Preliminary), then bound the attention logit deviation (Step 1), propagate the error through a single transformer block (Step 2), compose the bound across the $L_{\text{suf}} = L - \ell$ suffix blocks (Step 3), and convert the final CLS-token deviation into a KL divergence bound (Step 4). The resulting bound identifies $\mathcal{L}_{\text{rec}}$ and $\mathcal{L}_{\text{gram}}$ as the two quantities whose joint minimization is sufficient, in principle, to control predictive consistency without access to the classifier. The empirical correspondence is stronger than the worst-case composition below would predict, and is evaluated in Sections 4.3 and 4.4.

Notation. Let $\mathbf{A} \in \mathbb{R}^{N \times d}$ denote the patch-token activations at layer ℓ and $\hat{\mathbf{A}} = \mathbf{U}\mathbf{W}^\top$ their Semi-NMF reconstruction. Define the residual $\mathbf{E} = \mathbf{A} - \hat{\mathbf{A}}$, so that $\|\mathbf{E}\|_F^2 = 2\mathcal{L}_{\text{rec}}$. Let $\mathbf{G} = \mathbf{A}\mathbf{A}^\top$ and $\hat{\mathbf{G}} = \hat{\mathbf{A}}\hat{\mathbf{A}}^\top$ be the original and reconstructed Gram matrices. We write $\delta_{\text{rec}} = \|\mathbf{E}\|_F$ and $\delta_{\text{gram}} = \left(\mathcal{L}_{\text{gram}}\right)^{1/2} = \|\mathbf{G} - \hat{\mathbf{G}}\|_F / \|\mathbf{G}\|_F$ for the two error terms controlled by our objective.

C.1 Preliminary: Gram deviation under reconstruction error

Before analyzing the attention mechanism, we establish a direct relationship between the reconstruction error and the Gram-matrix deviation. This clarifies why $\mathcal{L}_{\text{rec}}$ alone provides *indirect* control over the Gram structure, and motivates the need for the explicit $\mathcal{L}_{\text{gram}}$ term.

Lemma 1. *Let* $\mathbf{A} \in \mathbb{R}^{N\times d}$ *and* $\hat{\mathbf{A}} = \mathbf{A} - \mathbf{E}$*. Then*

$$\|\mathbf{A}\mathbf{A}^\top - \hat{\mathbf{A}}\hat{\mathbf{A}}^\top\|_F \le 2\,\|\mathbf{A}\|_F \cdot \delta_{\text{rec}} + \delta_{\text{rec}}^2. \tag{15}$$

Proof. Substituting $\hat{\mathbf{A}} = \mathbf{A} - \mathbf{E}$ and expanding:

$$\mathbf{A}\mathbf{A}^\top - \hat{\mathbf{A}}\hat{\mathbf{A}}^\top = \mathbf{A}\mathbf{E}^\top + \mathbf{E}\mathbf{A}^\top - \mathbf{E}\mathbf{E}^\top.$$

Applying the triangle inequality and sub-multiplicativity ($\|\mathbf{X}\mathbf{Y}^\top\|_F \le \|\mathbf{X}\|_F\|\mathbf{Y}\|_F$):

$$\|\mathbf{A}\mathbf{A}^\top - \hat{\mathbf{A}}\hat{\mathbf{A}}^\top\|_F \le \|\mathbf{A}\|_F\|\mathbf{E}\|_F + \|\mathbf{E}\|_F\|\mathbf{A}\|_F + \|\mathbf{E}\|_F^2 = 2\,\|\mathbf{A}\|_F \cdot \delta_{\text{rec}} + \delta_{\text{rec}}^2. \qquad \square$$

Interpretation. Lemma 1 shows that reconstruction fidelity alone exerts some control over the Gram structure. However, this indirect control scales with $\|\mathbf{A}\|_F$, which can be large for high-dimensional transformer activations. The explicit $\mathcal{L}_{\text{gram}}$ term provides a *direct* constraint on $\|\mathbf{G} - \hat{\mathbf{G}}\|_F/\|\mathbf{G}\|_F$ that is normalized and independent of the activation scale. This distinction explains why reconstruction alone nearly suffices when $\|\mathbf{A}\|_F \cdot \delta_{\text{rec}}$ is already small, while the Gram loss adds measurable value precisely when the indirect control is loose.

C.2 Step 1: Attention logit deviation

Consider a single self-attention head with query, key, and value projections $W_Q, W_K, W_V \in \mathbb{R}^{d\times d_h}$. The pre-softmax attention logits are:

$$\mathbf{S} = \frac{1}{\sqrt{d_h}}\,\mathbf{A}\,W_Q W_K^\top\,\mathbf{A}^\top \in \mathbb{R}^{N\times N}. \tag{16}$$

Define $M = W_Q W_K^\top \in \mathbb{R}^{d\times d}$. The reconstructed logits are $\hat{\mathbf{S}} = \hat{\mathbf{A}}\,M\,\hat{\mathbf{A}}^\top/\sqrt{d_h}$. We decompose the deviation:

$$\begin{aligned}\mathbf{S} - \hat{\mathbf{S}} &= \frac{1}{\sqrt{d_h}}\big(\mathbf{A}M\mathbf{A}^\top - \hat{\mathbf{A}}M\hat{\mathbf{A}}^\top\big)\\ &= \frac{1}{\sqrt{d_h}}\big(\mathbf{E}\,M\,\mathbf{A}^\top + \mathbf{A}\,M\,\mathbf{E}^\top - \mathbf{E}\,M\,\mathbf{E}^\top\big),\end{aligned} \tag{17}$$

where the identity follows from substituting $\hat{\mathbf{A}} = \mathbf{A} - \mathbf{E}$ and expanding. Taking the Frobenius norm and applying the triangle inequality yields:

$$\sqrt{d_h}\|\mathbf{S} - \hat{\mathbf{S}}\|_F \le \|\mathbf{E}\,M\,\mathbf{A}^\top\|_F + \|\mathbf{A}\,M\,\mathbf{E}^\top\|_F + \|\mathbf{E}\,M\,\mathbf{E}^\top\|_F.$$

We bound each term using the sub-multiplicativity of matrix norms (specifically the mixed bound $\|XYZ\|_F \le \|X\|_F\|Y\|_{\text{op}}\|Z\|_F$) and the property that transposition preserves the Frobenius norm ($\|\mathbf{A}^\top\|_F = \|\mathbf{A}\|_F$). Recalling our definition $\delta_{\text{rec}} = \|\mathbf{E}\|_F$, the three terms are bounded respectively by:

$$\begin{aligned}\|\mathbf{E}\,M\,\mathbf{A}^\top\|_F &\le \|\mathbf{E}\|_F\|M\|_{\text{op}}\|\mathbf{A}\|_F = \|M\|_{\text{op}}\|\mathbf{A}\|_F\delta_{\text{rec}},\\ \|\mathbf{A}\,M\,\mathbf{E}^\top\|_F &\le \|\mathbf{A}\|_F\|M\|_{\text{op}}\|\mathbf{E}\|_F = \|M\|_{\text{op}}\|\mathbf{A}\|_F\delta_{\text{rec}},\\ \|\mathbf{E}\,M\,\mathbf{E}^\top\|_F &\le \|\mathbf{E}\|_F\|M\|_{\text{op}}\|\mathbf{E}\|_F = \|M\|_{\text{op}}\delta_{\text{rec}}^2.\end{aligned}$$

Summing these upper bounds and factoring out $\|M\|_{\text{op}}$ gives the final deviation bound:

$$\|\mathbf{S} - \hat{\mathbf{S}}\|_F \le \frac{\|M\|_{\text{op}}}{\sqrt{d_h}}\big(2\,\|\mathbf{A}\|_F \cdot \delta_{\text{rec}} + \delta_{\text{rec}}^2\big). \tag{18}$$

To connect the Gram-matrix alignment loss to this deviation, we decompose the projection matrix into an identity component and a residual: $M = I + (M - I)$. When $M = I$, the bilinear deviation reduces to $\|\mathbf{G} - \hat{\mathbf{G}}\|_F = \|\mathbf{G}\|_F \cdot \delta_{\text{gram}}$ by definition of δ_{gram}, which is directly controlled by $\mathcal{L}_{\text{gram}}$. Note that by Lemma 1, reconstruction alone would only bound this quantity indirectly as $2\|\mathbf{A}\|_F \cdot \delta_{\text{rec}} + \delta_{\text{rec}}^2$, underscoring the value of the explicit Gram constraint. Splitting the full deviation accordingly:

$$\mathbf{A}M\mathbf{A}^\top - \hat{\mathbf{A}}M\hat{\mathbf{A}}^\top = \underbrace{(\mathbf{G} - \hat{\mathbf{G}})}_{\text{Gram error}} + \underbrace{\mathbf{A}(M-I)\mathbf{A}^\top - \hat{\mathbf{A}}(M-I)\hat{\mathbf{A}}^\top}_{\text{residual}}. \tag{19}$$

The first term is bounded by $\|\mathbf{G}\|_F \cdot \delta_{\text{gram}}$. The second term has the same structure as Eq. (17) with M replaced by $(M - I)$, yielding:

$$\|\mathbf{S} - \hat{\mathbf{S}}\|_F \leq \frac{1}{\sqrt{d_h}} \Big(\|\mathbf{G}\|_F \cdot \delta_{\text{gram}} + \|M - I\|_{\text{op}} \big(2\|\mathbf{A}\|_F \cdot \delta_{\text{rec}} + \delta_{\text{rec}}^2\big) \Big). \tag{20}$$

Equation (20) reveals the complementary roles of the two losses. $\mathcal{L}_{\text{gram}}$ directly controls the dominant identity component of the attention deviation, while $\mathcal{L}_{\text{rec}}$ handles the projection-specific residual. When $M \approx I$ (i.e. the head computes near-identity queries and keys), the Gram term dominates.

Crucially, this decomposition also suggests why purely reconstruction-based methods (like SAEs or standard NMF) may be insufficient for predictive faithfulness in ViTs. By Lemma 1, reconstruction alone (δ_{rec}) provides only an *indirect* bound on the Gram deviation that scales with $\|\mathbf{A}\|_F$, leaving the identity component of the attention logits loosely constrained. Our explicit Gram-matrix alignment plugs this gap with a scale-normalized, direct constraint.

C.3 Step 2: Single-block error propagation

A standard transformer block applies multi-head self-attention (MHSA) followed by a feed-forward network (FFN), each wrapped in a residual connection and layer normalization (LN):

$$\mathbf{A}' = \text{LN}\big(\mathbf{A} + \text{MHSA}(\mathbf{A})\big), \tag{21}$$

$$\mathbf{A}'' = \text{LN}\big(\mathbf{A}' + \text{FFN}(\mathbf{A}')\big). \tag{22}$$

We analyze the error propagation through each sub-layer.

MHSA output deviation. For a single head, the output is $\mathbf{O} = \text{softmax}(\mathbf{S})\,\mathbf{A}W_V \in \mathbb{R}^{N \times d_h}$, where $\mathbf{S}$ is defined in Eq. (16). Define the attention matrices $\mathbf{P} = \text{softmax}(\mathbf{S})$ and $\hat{\mathbf{P}} = \text{softmax}(\hat{\mathbf{S}})$. The output deviation decomposes as:

$$\begin{aligned} \mathbf{O} - \hat{\mathbf{O}} &= \mathbf{P}\,\mathbf{A}W_V - \hat{\mathbf{P}}\,\hat{\mathbf{A}}W_V \\ &= \mathbf{P}\,\mathbf{A}W_V - \hat{\mathbf{P}}\,(\mathbf{A} - \mathbf{E})W_V \\ &= (\mathbf{P} - \hat{\mathbf{P}})\,\mathbf{A}W_V + \hat{\mathbf{P}}\,\mathbf{E}\,W_V. \end{aligned} \tag{23}$$

Taking the Frobenius norm:

$$\|\mathbf{O} - \hat{\mathbf{O}}\|_F \leq \|\mathbf{P} - \hat{\mathbf{P}}\|_F \cdot \|\mathbf{A}\|_F \cdot \|W_V\|_{\text{op}} + \sqrt{N} \cdot \|W_V\|_{\text{op}} \cdot \delta_{\text{rec}}, \tag{24}$$

where we used $\|\hat{\mathbf{P}}\|_{\text{op}} \leq \sqrt{N}$ since the spectral norm of an $N \times N$ row-stochastic matrix is bounded by $\sqrt{N}$.

The first term in Eq. (24) depends on the deviation of the attention probability matrix $\mathbf{P}$. Because the attention operation applies the softmax function independently to each row of the logit matrix $\mathbf{S}$, we first bound the error for a single row. Let $\mathbf{s}, \hat{\mathbf{s}} \in \mathbb{R}^N$ be corresponding row vectors extracted from $\mathbf{S}$ and $\hat{\mathbf{S}}$. The softmax function is 1-Lipschitz continuous:

$$\|\text{softmax}(\mathbf{s}) - \text{softmax}(\hat{\mathbf{s}})\|_2 \leq \|\mathbf{s} - \hat{\mathbf{s}}\|_2, \tag{25}$$

because its Jacobian matrix, $J = \text{diag}(\mathbf{p}) - \mathbf{p}\mathbf{p}^\top$, has a spectral norm strictly bounded by 1. Applying this 1-Lipschitz property independently across all N rows and summing the squared errors yields the matrix-level bound in the Frobenius norm:

$$\|\mathbf{P} - \hat{\mathbf{P}}\|_F \leq \|\mathbf{S} - \hat{\mathbf{S}}\|_F. \tag{26}$$

Residual connection and LayerNorm. The residual connection produces $\mathbf{A} + \mathbf{O} = \mathbf{A} + \text{MHSA}(\mathbf{A})$, so its deviation is bounded by $\delta_{\text{rec}} + \|\mathbf{O} - \hat{\mathbf{O}}\|_F$. LayerNorm is locally Lipschitz: for bounded inputs there exists a constant L_{LN} such that

$$\|\text{LN}(\mathbf{x}) - \text{LN}(\hat{\mathbf{x}})\| \leq L_{\text{LN}} \|\mathbf{x} - \hat{\mathbf{x}}\|. \tag{27}$$

The FFN is a two-layer MLP with GELU activation, which is $\sqrt{2/\pi}$-Lipschitz, so the FFN sub-layer is Lipschitz with constant $L_{\text{FFN}} = \|W_1\|_{\text{op}} \cdot \|W_2\|_{\text{op}} \cdot \sqrt{2/\pi}$.

Composing all sub-layers of a single block, we obtain a per-block Lipschitz constant:

$$L_{\text{block}} = L_{\text{LN}}\big(1 + L_{\text{FFN}}\big) \cdot L_{\text{LN}}\big(1 + L_{\text{MHSA}}\big), \tag{28}$$

where L_{MHSA} absorbs the per-head bounds from Eq. (24) summed across all H heads.

C.4 Step 3: Composition through suffix blocks

Let $\mathbf{A}^{(0)} = \mathbf{A}$, $\hat{\mathbf{A}}^{(0)} = \hat{\mathbf{A}}$, and define $\mathbf{A}^{(t)}, \hat{\mathbf{A}}^{(t)}$ as the token sequences after t suffix blocks. Applying the per-block bound recursively:

$$\|\mathbf{A}^{(t)} - \hat{\mathbf{A}}^{(t)}\|_F \le L_{\text{block}}^{t} \, \|\mathbf{A}^{(0)} - \hat{\mathbf{A}}^{(0)}\|_F + \sum_{k=0}^{t-1} L_{\text{block}}^{t-1-k} \, \eta_{\text{gram}}^{(k)}, \tag{29}$$

where $\eta_{\text{gram}}^{(k)}$ is the additional error injected at block k due to the Gram-matrix deviation of $\mathbf{A}^{(k)}$. In the worst case, $\eta_{\text{gram}}^{(k)} \le c_k \cdot \|\mathbf{G}^{(k)}\|_F \cdot \delta_{\text{gram}}^{(k)}$ for constants c_k depending on the attention projections. At $k = 0$ this is directly controlled by our loss; at subsequent blocks it is controlled indirectly through the accumulated deviation.

After all $L_{\text{suf}} = L - \ell$ suffix blocks, the CLS token deviation satisfies:

$$\|z_{\text{cls}} - \hat{z}_{\text{cls}}\|_2 \le \|\mathbf{A}^{(L_{\text{suf}})} - \hat{\mathbf{A}}^{(L_{\text{suf}})}\|_F \le L_{\text{block}}^{L_{\text{suf}}} \left(\sqrt{N}\, \delta_{\text{rec}} + \alpha \, \|\mathbf{G}\|_F \cdot \delta_{\text{gram}}\right), \tag{30}$$

where α collects the constants from the attention projections and the geometric series, and $\sqrt{N}$ carries forward the sequence-length scaling introduced by the stochastic attention matrix in Eq. (24). This makes explicit that both δ_{rec} and δ_{gram} contribute to controlling the final representation.

C.5 Step 4: From CLS-token deviation to KL divergence

Let $h(z) = W_c\, z + b$ be a linear classifier mapping the CLS token to C-class logits, which we denote as $\mathbf{u}$. The logit deviation is bounded by:

$$\|\mathbf{u} - \hat{\mathbf{u}}\|_2 = \|W_c(z_{\text{cls}} - \hat{z}_{\text{cls}})\|_2 \le \|W_c\|_{\text{op}} \, \|z_{\text{cls}} - \hat{z}_{\text{cls}}\|_2. \tag{31}$$

To rigorously convert this logit deviation into a KL divergence bound without introducing dependencies on logit magnitudes, we exploit the connection between KL divergence and Bregman divergences.

Let $A(\mathbf{u}) = \log \sum_{c=1}^{C} \exp(\mathbf{u}_c)$ be the log-partition function. Its gradient yields the softmax probabilities, $\nabla A(\mathbf{u}) = p$, and its Hessian is the covariance matrix of the categorical distribution, $\nabla^2 A(\mathbf{u}) = \operatorname{diag}(p) - pp^\top$. Expressing the KL divergence $D_{\text{KL}}(p \,\|\, \hat{p})$ in terms of the logits gives:

$$\begin{aligned} D_{\text{KL}}(p \,\|\, \hat{p}) &= \sum_{c=1}^{C} p_c \log \frac{\exp(\mathbf{u}_c)/\exp(A(\mathbf{u}))}{\exp(\hat{\mathbf{u}}_c)/\exp(A(\hat{\mathbf{u}}))} \\ &= \sum_{c=1}^{C} p_c(\mathbf{u}_c - \hat{\mathbf{u}}_c) - A(\mathbf{u}) + A(\hat{\mathbf{u}}) \\ &= A(\hat{\mathbf{u}}) - A(\mathbf{u}) - \nabla A(\mathbf{u})^\top(\hat{\mathbf{u}} - \mathbf{u}). \end{aligned} \tag{32}$$

Equation (32) reveals that the KL divergence is exactly the Bregman divergence $D_A(\hat{\mathbf{u}}, \mathbf{u})$ generated by the log-partition function. By the multivariate Taylor's theorem, this error is precisely the second-order remainder:

$$D_{\text{KL}}(p \,\|\, \hat{p}) = \frac{1}{2}(\mathbf{u} - \hat{\mathbf{u}})^\top H(\xi)\,(\mathbf{u} - \hat{\mathbf{u}}), \tag{33}$$

where $H(\xi) = \operatorname{diag}(q) - qq^\top$ is the Hessian evaluated at an intermediate point $\xi = t\mathbf{u} + (1-t)\hat{\mathbf{u}}$ (for some $t \in [0,1]$) lying on the line segment connecting the two logit vectors, with corresponding softmax probabilities $q = \sigma(\xi)$.

Because $qq^\top$ is positive semi-definite, we have $H(\xi) \preceq \operatorname{diag}(q)$. Since all probabilities $q_c \in (0,1)$, the maximum eigenvalue of $\operatorname{diag}(q)$ is strictly bounded by 1. Consequently, the spectral norm of the Hessian satisfies $\|H(\xi)\|_{\text{op}} \le 1$ globally, regardless of the magnitude of the intermediate logits. Applying this to Equation (33) yields a strict, magnitude-independent quadratic bound:

$$D_{\text{KL}}(p \,\|\, \hat{p}) \le \frac{1}{2} \, \|\mathbf{u} - \hat{\mathbf{u}}\|_2^2. \tag{34}$$

Substituting Eq. (31) and Eq. (30) into Eq. (34):

$$\boxed{D_{\text{KL}}\big(\sigma(h \circ g_{>\ell}(\mathbf{A})) \,\big\|\, \sigma(h \circ g_{>\ell}(\hat{\mathbf{A}}))\big) \le \frac{1}{2} \, \|W_c\|_{\text{op}}^2 \, L_{\text{block}}^{2L_{\text{suf}}} \left(\sqrt{N}\, \delta_{\text{rec}} + \alpha \, \|\mathbf{G}\|_F \cdot \delta_{\text{gram}}\right)^2.} \tag{35}$$

C.6 Discussion

What the bound identifies. Equation (35) attributes predictive consistency jointly to δ_{rec} and δ_{gram}. The decomposition $M = I + (M - I)$ in Section C.2 makes the roles of the two terms explicit: the identity component of every query-key product is controlled directly by $\mathcal{L}_{\text{gram}}$, since preserving $\mathbf{A}\mathbf{A}^\top$ is exactly preserving the bilinear form at $M = I$, while the residual $(M - I)$ is controlled by $\mathcal{L}_{\text{rec}}$ through Eq. (20). The two losses are therefore not redundant in the worst case: $\mathcal{L}_{\text{gram}}$ supplies a direct constraint on the attention-logit deviation that $\mathcal{L}_{\text{rec}}$ bounds only indirectly via $\|\mathbf{A}\|_F \cdot \delta_{\text{rec}}$ (Lemma 1), while $\mathcal{L}_{\text{rec}}$ supplies the pointwise activation fidelity that the Gram loss, being invariant under any orthogonal transformation acting on the right of $\mathbf{A}$, cannot constrain at all. This complementarity is the formal content of the claim that geometric preservation plus reconstruction fidelity substitute for classifier supervision.

Reconciling backbone-dependent effects. Lemma 1 may also explain why the empirical benefit of $\mathcal{L}_{\text{gram}}$ varies across architectures (Tables 1 and 2). The indirect Gram control provided by reconstruction scales as $2\|\mathbf{A}\|_F \cdot \delta_{\text{rec}} + \delta_{\text{rec}}^2$. For ViT-MAE, the masked autoencoder pretraining objective produces activations whose pairwise geometry is already well-captured by reconstruction alone: the product $\|\mathbf{A}\|_F \cdot \delta_{\text{rec}}$ is small enough that the indirect bound is tight, consistent with the near-identical MCS of reconstruction and +Gram in Table 2 (0.921 and 0.923). Conversely, DINOv2's contrastive pretraining yields activations with richer pairwise structure and larger $\|\mathbf{A}\|_F$, so the indirect bound is looser and the explicit, scale-normalized Gram constraint can provide a tighter handle, although in the class-conditioned setting the observed MCS difference (0.840 to 0.857) lies within one standard deviation. However this argument applies to the class-conditioned protocol; in the task-agnostic setting of Section 4.4, where a single bank must cover multiple classes, the ordering reverses and ViT-MAE shows the larger Gram gain.

What the bound does not characterise. The operator norm $\|W_Q W_K^\top - I\|_{\text{op}}$ and its per-head distribution determine how much of the logit deviation is absorbed by the Gram term versus the reconstruction term in practice; we do not measure it. We also do not track how $\mathcal{L}_{\text{gram}}$ at layer ℓ propagates to the Gram matrices of subsequent layers: the bound preserves only the ℓ-th Gram matrix and invokes Lipschitz composition (Eq. (28)) for everything beyond.

Worst-case bounds vs. empirical tightness. As is standard in the theoretical analysis of deep neural networks, our worst-case bound compounds multiplicatively across layers ($L_{\text{block}}^{2L_{\text{suf}}}$). Because it assumes adversarial alignment of sublayer errors, the theoretical bound is looser than the empirical divergences observed in Table 1. LayerNorm, in particular, acts as a strong empirical contractor for low-rank signals, a dynamic not captured by worst-case Lipschitz constants.

Therefore, Equation (35) should be interpreted not as a strict numerical predictor, but as a structural identification of the sufficient control variables: it formally shows that $\mathcal{L}_{\text{gram}}$ and $\mathcal{L}_{\text{rec}}$ are the quantities whose joint minimization controls downstream predictive divergence in a task-agnostic manner.

Dependence on extraction depth (ℓ). The looseness of the bound is strictly governed by the number of suffix blocks, $L_{\text{suf}} = L - \ell$. Extracting concepts closer to the classifier (increasing ℓ) tightens the theoretical bound by reducing the exponent in $L_{\text{block}}^{2L_{\text{suf}}}$. In the extreme case of extracting at the final layer ($\ell = L$, so $L_{\text{suf}} = 0$), the transformer composition term vanishes entirely ($L_{\text{block}}^0 = 1$), and the KL divergence bound collapses to:

$$D_{\text{KL}}\big(\sigma(h(\mathbf{A})) \,\big\|\, \sigma(h(\hat{\mathbf{A}}))\big) \leq \frac{1}{2}\,\|W_c\|_{\text{op}}^2 \left(\sqrt{N}\,\delta_{\text{rec}} + \alpha\,\|\mathbf{G}\|_F \cdot \delta_{\text{gram}}\right)^2. \tag{36}$$

However, this introduces a representation trade-off: while the predictive consistency bound becomes tighter, representations at deeper layers become increasingly task-specific, potentially degrading the spatial localization of the Semi-NMF concepts compared to intermediate layers.

What the bound does, and does not, justify about the method. Eq. (35) is a sufficiency statement: it identifies $\mathcal{L}_{\text{rec}}$ and $\mathcal{L}_{\text{gram}}$ as two quantities whose joint minimization controls predictive consistency through any linear classifier attached downstream, without requiring that classifier during concept extraction. It is not a necessity statement, and does not exclude alternative activation-space losses (for example, higher-order moment matching or layer-wise CKA) from achieving comparable control. The empirical question of whether $\mathcal{L}_{\text{gram}}$ recovers the same concept subspace as full classifier supervision, and whether it does so more faithfully than reconstruction alone, is addressed in Sections 4.3 and 4.4

via the RSVC framework. The empirical agreement is much stronger than this worst-case bound predicts. However, as expected from our analysis, the bound's true value lies in identifying the correct mechanism rather than providing tight numerical guarantees.

D Implementation Details

D.1 Linear probes for validation

To evaluate our unsupervised concept banks against supervised ones via the RSVC framework, we attach a lightweight linear probe on top of each frozen backbone. The probe consists of a single linear layer ($d \to C$) applied to the CLS token. We train for 10 epochs using the Adam optimizer with a learning rate of 10^{-4}, no weight decay, and a batch size of 64. Table 5 reports the probe accuracy and macro-F1 scores on COCO for each backbone. The probes for DINOv2, ViT, and ViT-MAE on ImageNet are not finetuned; rather, we utilize the classification heads distributed with the respective checkpoints: for ViT we leverage `timm` (i.e. for `vit_base_patch16_224` we use `augreg2_in21k_ft_in1k`), we get `dinov2_vitb14_lc` from `torch.hub`, and for ViT-MAE we use the Meta AI version available from `https://dl.fbaipublicfiles.com/mae/finetune/mae_finetuned_vit_base.pth` that uses global average pooling + fully connected norm instead of the CLS token as the classifier-head input (this because the CLS token is not trained on ViT-MAE).

Table 5: Linear probe performance per backbone on COCO. All probes share the same architecture (CLS pooling, d=768) and training schedule.

Backbone	Accuracy (%)	Macro F1 (%)
DINOv2 ViT-B/14	95.5	95.5
ViT-B/16	93.8	93.8
ViT-MAE B/16	90.0	90.0

D.2 PLACE Hyperparameters

Table 6 lists all hyperparameters used by PLACE for concept extraction. These are shared across all three loss variants (reconstruction, +Gram, +KL) and across all backbones and datasets, unless otherwise noted. We use five seeds ($s \in \{42, 43, 44, 45, 46\}$) on all reported results; tables report mean $\pm$ standard deviation across seeds.

Table 6: PLACE concept extraction hyperparameters.

Parameter	Value	Description
`extract_layer`	9	Transformer block from which patch tokens are extracted
`group_size` s	2	Spatial grouping factor ($s \times s$ patches $\to$ 1 token)
`rank` r	25	Number of concepts per class
`steps`	22,000	Maximum optimization steps (early stopping on plateau)
`learning_rate`	5×10^{-4}	AdamW learning rate
`batch_size`	4	Mini-batch size for optimization
λ	1.0	Weight of the faithfulness loss ($\mathcal{L}_{\text{gram}}$ or $\mathcal{L}_{\text{KL}}$)
ϵ	10^{-6}	Early stopping criterion
$\ell_1(U)$	10^{-6}	ℓ_1 regularization on the coefficient matrix U
$\ell_2(W)$	10^{-6}	ℓ_2 regularization on the basis matrix W
NNDSVD %	80	Fraction of rows subsampled for NNDSVD initialization

The regularization terms $\ell_1(U)$ and $\ell_2(W)$ are set to small values (10^{-6}) to mildly encourage sparsity in the concept activations and prevent unbounded growth of the basis vectors, respectively, without materially affecting reconstruction quality. The faithfulness weight λ=1.0 gives equal importance to the reconstruction and geometric/KL loss terms; a sweep over λ is reported in Appendix E.2. It is worth noting that the `batch_size` is used for every loss (Gram, KL and MSE).

Sparse Autoencoders. SAEs baseline comparisons follow the same parameters where applicable, with a learning rate of 10^{-3} and a ℓ_1 penalty of 10^{-3}.

D.3 PCA Visualization Details

To visualize the raw patch-token activations in Appendix H, we project the high-dimensional token representations into RGB space. We follow a two-step principal component analysis (PCA) approach to separate the foreground object from the background and highlight intra-object feature variance. This process is applied independently to each image.

First, we extract the raw patch tokens $\mathbf{A} \in \mathbb{R}^{N \times d}$ and fit a 1D PCA over all N tokens. The first principal component typically captures the highest variance direction in the representation space, effectively acting as a foreground-background discriminator. Tokens with a positive projection score are designated as foreground. If this produces a foreground mask covering less than 30% of the patches, we assume the polarity is inverted and flip the sign of the scores.

Second, to capture the feature structure without interference from background variance, we fit a 3D PCA strictly on the foreground tokens. We then project all N tokens through these three components. To generate the final RGB values, we normalize each of the three component channels to $[0, 1]$ using the minimum and maximum values computed exclusively from the foreground tokens. Finally, the background patches are dimmed by a factor of 0.3, and the resulting $\sqrt{N} \times \sqrt{N}$ grid is bilinearly upsampled to the original input resolution $H \times W$.

E Additional Analyses

In this section, we present comprehensive analyses to justify our architectural choices and hyperparameter settings. Unless otherwise specified, all ablations are performed on the ImageNet dataset using the *church* class. We examine the impact of the NNDSVD subsampling ratio (Appendix E.1), the faithfulness loss weight λ (Appendix E.2), the extraction layer ℓ (Appendix E.3), and the decomposition rank r (Appendix E.4). Furthermore, we include a weight randomization sanity check (Appendix E.6), an analysis of the stability of our factorization across different random seeds (Appendix E.7) and runtime measurements (Appendix E.9). Finally we discuss the computational complexity and scalability of PLACE (Appendix E.10), the robustness to image perturbations (Appendix E.13), and the complete, per-class results (Appendix E.12). The default configuration used throughout the main experiments (Section 4) is listed in Appendix D.2.

E.1 NNDSVD subsampling ratio for Semi-NMF initialization

Computing the full SVD of the activation matrix $\mathbf{A}$ is expensive for large datasets; a natural question is whether computing NNDSVD on a random subset of rows degrades the final solution.

We run concept extraction on the ImageNet *church* class (878 correctly-classified images) using ViT-B/16 (`timm`, `augreg2_in21k_ft_in1k`) at layer ℓ=9 with rank r=25 and group size s=2, varying the fraction of rows subsampled for NNDSVD initialization: 100%, 80%, 60%, 40%, and 20%. All runs use the MSE+KL loss with a budget of 22,000 steps and early stopping.

Table 7: Effect of NNDSVD subsampling ratio on Semi-NMF concept extraction (ViT-B/16, ImageNet *church*, r=25, s=2). Mean ± std over 5 seeds.

NNDSVD %	MSE ↓	D_{KL} ↓	Steps ↓	C-Ins ↑	C-Del ↑	C-Gini ↑
100	0.306 ± 0.002	0.011 ± 0.001	12100 ± 7477	0.903 ± 0.032	0.162 ± 0.014	0.458 ± 0.014
80	0.302 ± 0.002	0.015 ± 0.003	4847 ± 2265	0.911 ± 0.003	0.147 ± 0.023	0.439 ± 0.014
60	0.309 ± 0.002	0.013 ± 0.002	6012 ± 4814	0.906 ± 0.048	0.138 ± 0.022	0.460 ± 0.016
40	0.303 ± 0.004	0.015 ± 0.007	3896 ± 2252	0.913 ± 0.031	0.123 ± 0.063	0.462 ± 0.013
20	0.303 ± 0.003	0.013 ± 0.003	5587 ± 4162	0.912 ± 0.008	0.130 ± 0.055	0.452 ± 0.010
0	0.304 ± 0.003	0.014 ± 0.002	6992 ± 141	0.764 ± 0.073	0.127 ± 0.026	0.312 ± 0.039

Discussion. The results show that Semi-NMF is remarkably robust to the NNDSVD subsampling ratio in terms of final reconstruction accuracy. Reconstruction error (MSE) varies marginally across all settings, hovering between 0.302 and 0.309, while the KL divergence remains tightly bounded in

the range $[0.011, 0.015]$. Even when entirely abandoning NNDSVD in favor of random initialization (0%), the model achieves comparable reconstruction metrics.

However, logging the optimization steps required for convergence reveals a high degree of variance in the optimization trajectory. Full 100% NNDSVD initialization required the highest average number of steps (12100 ± 7477) before triggering early stopping. Subsampling the initialization matrix reduced the mean convergence time: 80% took 4847 ± 2265 steps, and 40% took 3896 ± 2252 steps. Random initialization (0%) required a moderate 6992 ± 141 steps, notably exhibiting the lowest variance in convergence time across seeds.

Crucially, while reconstruction quality is agnostic to the initialization strategy, the interpretability and faithfulness of the extracted concepts are highly dependent on NNDSVD. The 0% setting suffers a degradation in concept quality: C-Ins drops precipitously to 0.764 ± 0.073 (compared to > 0.900 for all NNDSVD runs), and C-Gini falls to 0.312 ± 0.039, indicating less sparse and interpretable concepts. As long as NNDSVD is used, even at just 20% subsampling, faithfulness metrics remain strong and stable (e.g., C-Ins of 0.912 ± 0.008). C-Del is highest (best) at 100% (0.162), with a monotonic but mild degradation as subsampling decreases, though random initialization performs comparably to the lower NNDSVD ratios.

In practice, subsampling to 40%–80% offers a superior trade-off. It avoids the decrease in interpretability observed with random initialization, while reducing both the initial NNDSVD compute requirements and the number of subsequent optimization steps needed to converge, all while retaining full-data solution quality.

E.2 Faithfulness loss weight sensitivity

We sweep the faithfulness loss weight λ in $\mathcal{L} = \mathcal{L}_{\text{MSE}} + \lambda \mathcal{L}_{\text{faith}}$ across seven values from 0.01 to 10 for both the KL and Gram variants, and report the effect on reconstruction quality, faithfulness, and concept sparsity. All experiments use ViT-B/16 (`timm`, `augreg2_in21k_ft_in1k`), ImageNet *church* class, layer ℓ=9, r=25, s=2; values are mean $\pm$ std over five seeds.

Table 8: Effect of λ on concept extraction with the KL loss (ViT-B/16, ImageNet *church*, r=25, s=2). Mean $\pm$ std over 5 seeds.

λ	MSE $\downarrow$	D_{KL} $\downarrow$	C-Ins $\uparrow$	C-Del $\uparrow$	C-Gini $\uparrow$
0.01	0.296 ± 0.002	0.106 ± 0.015	0.882 ± 0.019	0.088 ± 0.023	0.402 ± 0.005
0.1	0.298 ± 0.003	0.033 ± 0.010	0.879 ± 0.026	0.110 ± 0.028	0.463 ± 0.011
0.5	0.300 ± 0.002	0.017 ± 0.003	0.887 ± 0.031	0.136 ± 0.026	0.445 ± 0.009
1.0	0.302 ± 0.002	0.015 ± 0.003	0.911 ± 0.003	0.147 ± 0.023	0.439 ± 0.014
2.0	0.302 ± 0.002	0.011 ± 0.002	0.917 ± 0.037	0.156 ± 0.031	0.433 ± 0.013
5.0	0.306 ± 0.001	0.008 ± 0.001	0.933 ± 0.009	0.149 ± 0.029	0.425 ± 0.007
10.0	0.312 ± 0.003	0.007 ± 0.002	0.932 ± 0.011	0.143 ± 0.044	0.423 ± 0.012

Table 9: Effect of λ on concept extraction with the Gram loss (ViT-B/16, ImageNet *church*, r=25, s=2). Mean $\pm$ std over 5 seeds.

λ	MSE $\downarrow$	$\mathcal{L}_{\text{gram}}$ $\downarrow$	C-Ins $\uparrow$	C-Del $\uparrow$	C-Gini $\uparrow$
0.01	0.287 ± 0.001	0.018 ± 0.001	0.710 ± 0.020	0.309 ± 0.087	0.600 ± 0.010
0.1	0.292 ± 0.002	0.023 ± 0.003	0.710 ± 0.020	0.300 ± 0.077	0.580 ± 0.010
0.5	0.292 ± 0.010	0.024 ± 0.020	0.710 ± 0.020	0.319 ± 0.087	0.580 ± 0.010
1.0	0.293 ± 0.003	0.022 ± 0.010	0.707 ± 0.028	0.300 ± 0.011	0.571 ± 0.017
2.0	0.281 ± 0.006	0.010 ± 0.002	0.710 ± 0.020	0.300 ± 0.087	0.560 ± 0.020
5.0	0.282 ± 0.007	0.010 ± 0.002	0.700 ± 0.030	0.309 ± 0.058	0.560 ± 0.020
10.0	0.282 ± 0.014	0.012 ± 0.007	0.690 ± 0.020	0.338 ± 0.077	0.520 ± 0.060

Discussion. The two losses respond to λ in qualitatively different ways, reflecting the distinction between a supervised and an unsupervised faithfulness objective.

For the KL variant (Table 8), increasing λ produces a clear MSE–faithfulness trade-off: MSE rises monotonically from 0.296 to 0.312, while D_{KL} drops by more than an order of magnitude ($0.106 \rightarrow 0.007$). C-Ins improves steadily from 0.882 to 0.933, with the sharpest gain between λ=0.5 and λ=5. C-Del improves rapidly from 0.088 at λ=0.01 to $\sim$0.147 at λ=1 and then plateaus, indicating that moderate KL pressure suffices for faithful concept ordering.

The Gram variant (Table 9) tells a different story. Because $\mathcal{L}_{\text{gram}}$ constrains the pairwise geometry of the reconstruction without reference to the downstream classifier, its influence on C-Ins and C-Del is second-order: C-Ins varies only between 0.69 and 0.71, and C-Del remains in the 0.30–0.34 range across all λ values. MSE is similarly stable (0.28–0.29), and C-Gini decreases mildly from 0.60 to 0.52. The Gram loss itself saturates around $\sim$0.010 for $\lambda \geq 2$, with diminishing returns beyond. The main practical difference is convergence speed: at λ=0.01 the optimizer converges in $\sim$300 steps, while $\lambda \geq 2$ requires $1{,}000$–$3{,}000$ steps.

Comparing the two, the KL loss is the more responsive lever for controlling faithfulness but introduces a genuine trade-off with reconstruction quality. The Gram loss is robust to λ across a wide range, making it the safer default for unsupervised settings where the practitioner cannot tune λ against a classifier. We select λ=1 for both variants: for the KL loss this balances low D_{KL}, competitive C-Ins, and minimal MSE penalty; for the Gram loss the choice is largely inconsequential but ensures consistency across the experimental setup.

E.3 Extraction layer ablation

We extract concepts at layers $\ell \in \{4, 6, 8, 9, 10\}$ of a 12-block ViT and report the effect on concept quality metrics. For ViT-B/16 (`timm`), which has a classification head, we use the MSE+KL loss and report C-Ins and C-Del alongside reconstruction error. For DINOv2 ViT-B/14, which lacks a classification head, we use the MSE+Gram loss and report $\mathcal{L}_{\text{gram}}$ and Gini sparsity. All runs use r=25, s=2, λ=1; values are mean $\pm$ std over five seeds.

Table 10: Layer ablation on ViT-B/16 (`timm`, MSE+KL, ImageNet *church*, r=25, s=2). † Layer 10 is degenerate: with only two transformer blocks remaining, representations are highly aggregated, and the factorization heavily overfits the nearly-formed class prediction rather than discovering spatially localized concepts.

ℓ	MSE ↓	D_{KL} ↓	C-Ins ↑	C-Del ↑	C-Gini ↑
4	0.053 ± 0.001	0.040 ± 0.013	0.403 ± 0.035	0.848 ± 0.004	0.361 ± 0.008
6	0.086 ± 0.000	0.022 ± 0.002	0.702 ± 0.018	0.889 ± 0.003	0.370 ± 0.002
8	0.222 ± 0.001	0.014 ± 0.002	0.696 ± 0.022	0.641 ± 0.016	0.413 ± 0.015
9	0.302 ± 0.002	0.015 ± 0.003	0.911 ± 0.003	0.147 ± 0.023	0.439 ± 0.014
$10^{\dagger}$	0.621 ± 0.004	0.010 ± 0.001	0.948 ± 0.006	0.161 ± 0.042	0.434 ± 0.011

Table 11: Layer ablation on DINOv2 ViT-B/14 (MSE+Gram, ImageNet *church*, r=25, s=2).

ℓ	MSE ↓	$\mathcal{L}_{\text{gram}}$ ↓	C-Gini ↑
4	0.013 ± 0.001	0.042 ± 0.029	0.459 ± 0.053
6	0.012 ± 0.001	0.019 ± 0.003	0.605 ± 0.005
8	0.017 ± 0.001	0.020 ± 0.002	0.803 ± 0.008
9	0.093 ± 0.001	0.017 ± 0.004	0.923 ± 0.004
10	0.104 ± 0.001	0.019 ± 0.000	0.588 ± 0.009

Discussion. The two models exhibit complementary trends across depth. For ViT-B/16 with the KL loss, C-Ins increases sharply with depth, jumping from 0.403 at ℓ=4 to 0.911 at ℓ=9. This is expected: deeper layers encode increasingly class-discriminative features, so concepts extracted there are more faithful to the downstream prediction. Layer 10 achieves the highest raw C-Ins (0.948) but is functionally degenerate (marked †): at this depth (with only two transformer blocks remaining before the classifier), the representations are highly aggregated and task-specific. The CLS token has already accumulated global evidence, and reconstructing the patch tokens at this stage allows the Semi-NMF to trivially overfit the nearly-formed class logits. This inflates both

C-Ins and classification accuracy (0.987 vs. a 0.984 ungrouped baseline) without reflecting genuine, spatially-grounded concept structure.

Conversely, C-Del exhibits a decreasing pattern: it is exceptionally high at layers 4-6 (0.889 at ℓ=6), but drops to 0.147 at layer 9. High C-Del at early layers indicates that these concepts capture fundamental, low-level visual features; removing them completely breaks the model's ability to process the image. At layer 9, concepts are more semantically distinct (higher C-Gini, 0.439) but the model exhibits higher redundancy, meaning it can maintain its prediction even when individual high-level concepts are deleted. We select ℓ=9 as the default extraction layer for ViT-B/16 to prioritize the recovery of semantic, class-specific concepts (C-Ins) and sparsity (C-Gini), while acknowledging the trade-off in deletion sensitivity.

For DINOv2 ViT-B/14 with the Gram loss, MSE increases with depth (from 0.013 at ℓ=4 to 0.104 at ℓ=10), reflecting the growing complexity of later-layer representations. Gini sparsity increases with depth, peaking at ℓ=9 (0.923). Notably, at ℓ=10, the factorization begins to collapse: sparsity drops sharply to 0.588. Similar to the supervised ViT, this indicates that representations at $\ell \geq 10$ are too globally aggregated to be meaningfully factorized into sparse, localized concepts. Layer 9, used as the default throughout the paper, sits at the optimal point where representations are most distinct and specialized before this architectural collapse.

E.4 Decomposition rank ablation

We analyze the effect of the decomposition rank $r \in \{5, 10, 15, 20, 25, 30\}$ on faithfulness and concept quality. Table 12 and Table 13 report metrics averaged over five seeds on the `church` class.

Faithfulness (Table 12). C-Del decreases from 0.230 at r=10 to 0.125 at r=30, while C-Ins remains stable at $\sim$0.90 across all ranks, indicating that even a small number of concepts suffices to recover most of the target-class logit under progressive insertion. C-Gini increases monotonically, reflecting greater per-concept specialization at higher ranks. We note that at r=5 two of five seeds converge to a degenerate solution (high C-Del, low C-Ins), inflating the variance; this instability disappears at $r \geq 10$, suggesting that very low rank under-determines the KL-constrained factorization. Gains plateau between r=25 and r=30, motivating our choice of r=25 for the main experiments.

Reconstruction and stability (Table 13). MSE decreases monotonically from 0.099 to 0.092 as rank increases, confirming that additional concepts capture progressively finer structure. This trade-off between reconstruction fidelity and interpretability is expected: at higher rank, the decomposition captures more nuanced features, but risks fragmenting cohesive visual concepts. Gains plateau between r=25 and r=30, validating our choice of r=25 to balance granularity and conciseness.

Table 12: Rank ablation for PLACE (+KL) on the `church` class (ViT-B/16, s=2). Mean $\pm$ std over 5 seeds. $\downarrow$ = lower is better, $\uparrow$ = higher is better.

r	C-Ins $\uparrow$	C-Del $\uparrow$	C-Gini $\uparrow$
5	0.88 ± 0.13	0.419 ± 0.258	0.34 ± 0.03
10	0.90 ± 0.02	0.230 ± 0.036	0.45 ± 0.01
15	0.90 ± 0.02	0.131 ± 0.056	0.48 ± 0.01
20	0.87 ± 0.05	0.174 ± 0.038	0.52 ± 0.01
25	0.91 ± 0.01	0.147 ± 0.023	0.44 ± 0.01
30	0.91 ± 0.04	0.125 ± 0.058	0.58 ± 0.01

E.5 Token group size ablation

We evaluate the effect of the token group size $s \in \{1, 2, 4, 7, 14\}$ on faithfulness, reconstruction quality, and runtime, on ImageNet and COCO. As discussed in Section 3 and Appendix B.3, larger s reduces computational cost but coarsens the spatial resolution of concept heatmaps and introduces quantization error through the `repeat_interleave` operation.

Discussion. The results in Tables 14 and 15 reveal a clear trade-off between computational efficiency and spatial resolution.

Table 13: Rank ablation for PLACE (+Gram) on the `church` class (DINOv2 ViT-B/14, s=2). Mean $\pm$ std over 5 seeds.

r	MSE ↓	$\mathcal{L}_{\text{gram}}$ ↓	C-Gini ↑
5	0.099 ± 0.002	0.048 ± 0.002	0.64 ± 0.04
10	0.096 ± 0.001	0.031 ± 0.002	0.74 ± 0.03
15	0.094 ± 0.002	0.028 ± 0.006	0.83 ± 0.03
20	0.094 ± 0.000	0.034 ± 0.013	0.91 ± 0.00
25	0.093 ± 0.001	0.017 ± 0.004	0.923 ± 0.004
30	0.092 ± 0.001	0.017 ± 0.011	0.93 ± 0.02

Table 14: Effect of token group size s on ViT-B/16 (ImageNet, r=25, PLACE (+KL)). Grouped and ungrouped MSE disentangle quantization error from factorization error. Values are reported as mean $\pm$ standard deviation.

s	C-Ins ↑	C-Del ↑	MSE (grouped) ↓	MSE (ungrouped) ↓	Acc (Rec) ↑	Acc (Ungrouped) ↑	Time (s) ↓
1	0.913 ± 0.063	0.177 ± 0.044	0.193 ± 0.001	-	0.980 ± 0.003	-	348.538 ± 53.164
2	$\mathbf{0.962 \pm 0.013}$	0.132 ± 0.008	0.093 ± 0.002	0.089 ± 0.002	$\mathbf{0.990 \pm 0.002}$	$\mathbf{0.945 \pm 0.002}$	125.419 ± 6.041
4	0.918 ± 0.025	0.112 ± 0.015	0.126 ± 0.005	0.093 ± 0.004	0.801 ± 0.024	0.731 ± 0.001	118.969 ± 2.400
7	0.869 ± 0.059	0.165 ± 0.020	0.077 ± 0.001	0.077 ± 0.001	0.875 ± 0.006	0.733 ± 0.003	117.602 ± 2.102
14	0.917 ± 0.057	0.101 ± 0.020	0.036 ± 0.000	0.036 ± 0.000	0.709 ± 0.003	0.615 ± 0.004	$\mathbf{115.301 \pm 2.059}$

Table 15: Effect of token group size s on ViT-B/16 (COCO, r=25, PLACE (+KL)). Grouped and ungrouped MSE disentangle quantization error from factorization error. Values are reported as mean $\pm$ standard deviation.

s	C-Ins ↑	C-Del ↑	MSE (grouped) ↓	MSE (ungrouped) ↓	Acc (Rec) ↑	Acc (Ungrouped) ↑	Time (s) ↓
1	0.925 ± 0.033	0.975 ± 0.042	0.520 ± 0.017	-	$\mathbf{0.997 \pm 0.002}$	-	345.234 ± 43.513
2	$\mathbf{0.945 \pm 0.005}$	0.933 ± 0.231	0.353 ± 0.074	0.343 ± 0.074	0.993 ± 0.004	$\mathbf{0.994 \pm 0.004}$	118.942 ± 5.119
4	0.921 ± 0.028	0.915 ± 0.231	0.278 ± 0.047	0.231 ± 0.035	0.969 ± 0.014	0.970 ± 0.014	115.452 ± 2.941
7	0.815 ± 0.081	0.966 ± 0.207	0.215 ± 0.029	0.205 ± 0.027	0.912 ± 0.081	0.912 ± 0.081	113.305 ± 2.292
14	0.705 ± 0.192	0.909 ± 0.274	0.187 ± 0.013	0.167 ± 0.012	0.788 ± 0.218	0.788 ± 0.217	$\mathbf{110.601 \pm 1.911}$

On ImageNet, dense factorization ($s = 1$) preserves the native token grid but incurs a computational bottleneck ($\sim$ 348s). Setting $s = 2$ emerges as the optimal configuration: it reduces the optimization runtime by nearly $3\times$ ($\sim$ 125s) while simultaneously achieving the highest concept insertion score (C-Ins 0.962) and the highest post-reconstruction target accuracy (0.990). This indicates that a moderate grouping of 2×2 tokens effectively regularises the factorization, mitigating local variance among adjacent tokens and improving the extraction of predictively relevant concepts. While C-Del sees a slight drop compared to $s = 1$, the gains in C-Ins, accuracy, and speed make $s = 2$ the preferred choice. On COCO, $s = 2$ similarly reduces the runtime by $\sim 2.9\times$ ($\sim$ 345s to $\sim$ 119s) and achieves the highest C-Ins (0.945), while post-reconstruction accuracy remains close to the dense setting (0.993 vs. 0.997 at $s = 1$).

For larger group sizes ($s \geq 4$), spatial quantization error becomes detrimental. Grouping 4×4 or larger blocks of tokens aggressively smooths out fine-grained visual features, causing a sharp collapse in target-class accuracy (dropping to 0.801 at $s = 4$ and further deteriorating at larger sizes). On COCO the degradation is more gradual but follows the same trend (accuracy 0.969 at $s = 4$, 0.912 at $s = 7$, and 0.788 at $s = 14$).

Notably, the reconstruction error (MSE) tends to decrease at extreme grouping levels on both datasets (e.g., 0.036 on ImageNet and 0.187 on COCO at $s = 14$). We emphasize that this does not indicate superior factorization quality; rather, heavy spatial pooling creates a smaller, lower-variance activation matrix $\mathbf{A}$, which is trivially easier to fit with a low-rank approximation but devoid of meaningful spatial semantics. The consistency of these trends across datasets agrees with Appendix B.3: the cost and resolution of token grouping are set by the architecture rather than by the data. Based on these findings, we adopt $s = 2$ as the default hyperparameter, providing the ideal balance between fine-grained localization, predictive faithfulness, and computational tractability.

E.6 Sanity check: weight randomization

Following Fel et al. [25], we verify that our method's explanations degrade when the model's weights are randomized. We progressively randomize transformer blocks from the first to the last ($n \in \{0, 2, 4, 6, 8, 10\}$ out of 12 blocks) and re-extract concepts at layer ℓ=9. If the discovered concepts genuinely reflect learned representations rather than artifacts of the factorization, concept quality should degrade as more blocks are randomized.

Table 16: Weight randomization sanity check. n denotes the number of randomized blocks (from input side). C-Ins and C-Del collapse to 0 for $n \geq 2$ because the randomized model has 0% target-class accuracy. We report results on ViT-B/16 (`timm`, MSE+KL, ImageNet *church*, r=25, s=2), mean $\pm$ std over five seeds.

n	MSE ↓	D_{KL} ↓	C-Gini ↑	C-Ins ↑	C-Del ↑
0	0.302 ± 0.002	0.015 ± 0.003	0.439 ± 0.014	0.911 ± 0.003	0.147 ± 0.023
2	0.181 ± 0.001	0.022 ± 0.003	0.332 ± 0.021	0.0	0.0
4	0.150 ± 0.002	0.020 ± 0.005	0.310 ± 0.014	0.0	0.0
6	0.136 ± 0.002	0.048 ± 0.015	0.271 ± 0.021	0.0	0.0
8	0.101 ± 0.003	0.029 ± 0.012	0.371 ± 0.031	0.0	0.0
10	0.065 ± 0.001	0.004 ± 0.000	0.502 ± 0.011	0.0	0.0

Discussion. MSE decreases monotonically with randomization ($0.302 \rightarrow 0.065$), which is expected: random features have lower effective dimensionality and are more compressible by a rank-25 factorization; low reconstruction error under a randomized model should not be confused with explanation quality. The sanity check confirms that our method's explanations reflect learned representations. Randomizing even two blocks reduces target-class accuracy to 0%, collapsing C-Ins and C-Del to zero: the concepts extracted from a corrupted model carry no predictive information. C-Gini drops from 0.439 (intact model) to 0.271 at n=6, indicating that the coefficient matrix becomes uniformly diffuse rather than selectively activating distinct concepts. At n=8 and n=10, Gini partially recovers (0.371, 0.502) as one or two concepts absorb the dominant random direction, yielding a degenerate form of sparsity that does not correspond to meaningful visual concepts.

E.7 Stability across seeds

Since Semi-NMF, like NMF, yields non-unique solutions due to permutation and scaling ambiguities, we verify that the discovered concepts are stable across random seeds. We run concept extraction five times with different seeds on the same data. To account for permutation invariance, we measure stability by computing the mean cosine similarity between the columns of the concept dictionaries W across runs, after determining the optimal column alignment via the Hungarian algorithm (W-Stab).

Discussion. All three Semi-NMF variants achieve high W-Stab on ImageNet (mean $\geq$0.94), confirming that the method converges to consistent concept dictionaries despite the non-convexity of the factorization. Adding a faithfulness loss further improves stability: both +Gram and +KL outperform the reconstruction-only baseline on most classes, with the largest gains on *cassette* ($0.890 \rightarrow 0.931$), *church* ($0.897 \rightarrow 0.953$), and *tench* ($0.959 \rightarrow 0.995$). This is consistent with the geometric constraints reducing the effective solution space and biasing Semi-NMF toward more determined factorizations. +KL and +Gram reach nearly identical mean stability on both datasets (0.960 vs. 0.958 on ImageNet, 0.818 vs. 0.819 on COCO), suggesting that faithfulness metrics remain the primary basis for differentiating the two.

The inclusion of Sparse Autoencoders (SAEs) reveals a divergence in architectural stability. **Vanilla SAEs** [33, 57] exhibit high stability, particularly on the low-data COCO regime (mean 0.911), where they outperform all Semi-NMF variants. This suggests that the SAE's architectural priors and fixed initialization strategies may provide a more constrained optimization landscape than Semi-NMF. Conversely, **Gated SAEs** [61] show the lowest stability across all tested models (ImageNet mean 0.764, COCO mean 0.664). The additional gating parameters likely introduce a more complex, non-convex objective that is more sensitive to seed initialization, leading to more varied dictionary solutions across runs.

Table 17: W-Stab (Mean matched cosine similarity of W across seeds) per class and model variant (DINOv2 ViT-B/14).

Dataset	Class	Reconstruction	+Gram	+KL	Gated SAE	SAE
ImageNet	cassette	0.8904	0.9310	0.9309	0.7667	0.9561
	chainsaw	0.9013	0.9043	0.9052	0.7452	0.9260
	church	0.8969	0.9535	0.9531	0.7224	0.9038
	English springer	0.9330	0.9874	0.9919	0.7477	0.8906
	French horn	0.9426	0.9437	0.9550	0.7609	0.9403
	garbage truck	0.9367	0.9359	0.9365	0.7913	0.9459
	gas pump	0.9829	0.9926	0.9944	0.7791	0.9038
	golf ball	0.9895	0.9905	0.9923	0.8040	0.9046
	parachute	0.9436	0.9420	0.9449	0.8154	0.9223
	tench	0.9589	0.9953	0.9959	0.7041	0.9227
	Mean	**0.9376**	**0.9576**	**0.9600**	**0.7637**	**0.9216**
COCO	hot dog	0.8317	0.8242	0.8246	0.6459	0.8913
	teddy bear	0.8282	0.8515	0.8502	0.6612	0.8919
	train	0.8706	0.9053	0.9055	0.7329	0.9518
	umbrella	0.6871	0.6900	0.6877	0.5859	0.9088
	zebra	0.7805	0.8249	0.8232	0.6950	0.9121
	Mean	**0.7996**	**0.8192**	**0.8183**	**0.6642**	**0.9112**

Stability remains noticeably lower for most methods on COCO (mean ≤ 0.82 for Semi-NMF), driven by *umbrella* (22 images) and *zebra* (43 images). With very few images, the patch-token activation matrix is lower-rank and noisier, leaving more freedom for the factorization to settle in different local minima across seeds. While Vanilla SAEs seem more robust to this data-regime limitation, for Semi-NMF, this remains a challenge, we thoroughly isolate and quantify this limitation in Appendix E.8.

E.8 Effect of Dataset Size on Concept Stability

Because Semi-NMF relies on statistical co-occurrences of features across the dataset, its stability depends heavily on the sample size. To quantify this, we extract concepts ($r = 25, s = 2$) from the ImageNet `church` class using PLACE (+Gram) on the ViT-B/16 backbone at layer $\ell = 9$, progressively restricting the dataset size to 1%, 5%, 10%, 25%, 50%, 75%, and 100% of the available images. We report the matched cosine similarity of the dictionary $\mathbf{W}$ computed across 5 random seeds (W-Stab).

Table 18: Effect of dataset size on dictionary stability (W-Stab). Evaluated on the ImageNet `church` class across 5 random seeds using PLACE (+Gram). Images refer to the correctly classified subset used for extraction.

Data Fraction	Images	W-Stab ↑
1%	9	0.587
5%	47	0.799
10%	91	0.767
25%	217	0.877
50%	439	0.842
75%	659	0.873
100%	878	0.906

Discussion. As shown in Table 18, stability is highly sensitive to the sample regime. In the extreme low-data regime (1%, corresponding to just 9 images), W-Stab collapses to 0.587. In such settings, the activation matrix $\mathbf{A}$ is severely under-determined, allowing the factorization to settle into different local minima across seeds. As the number of images increases, stability climbs rapidly, surpassing

0.87 with approximately 200 images, and approaching 0.906 when the full 878-image subset is utilized.

This ablation provides crucial empirical context for the COCO results observed in Appendix E.7: classes like `umbrella` (22 images) and `zebra` (43 images) naturally exhibit lower stability (~ 0.69 for `umbrella`, ~ 0.78 to 0.82 for `zebra`). These datasets fall exactly into the extreme low-data regime demonstrated by our 1% and 5% splits, confirming that the performance drop is a fundamental structural limitation of NMF-based factorization on small sample sets, rather than a shortcoming of the geometric loss formulation.

E.9 Runtime and memory

We report wall-clock time and peak GPU memory for concept extraction on ImageNet using DINOv2 ViT-B/14 (r=25, s=2) measured on a single NVIDIA RTX 5060 Ti (16 GB VRAM, CUDA 12.8). All timings exclude data loading and preprocessing.

Table 19: Runtime and memory for concept extraction (DINOv2 ViT-B/14, ImageNet). Median across 10 classes.

Phase	Time (s)	Peak VRAM (GB)
Activation extraction + NNDSVD init	3.9	10.9
Optimization (PLACE, reconstruction)	5.3	10.9
Optimization (PLACE, +Gram)	5.6	10.9
Optimization (PLACE, +KL)	118.0	10.6
Training (SAE)	62.5	8.0
Training (Gated SAE)	71.4	8.0

Discussion. Activation extraction and NNDSVD initialization are shared across all variants and complete in under 5 s per class. Both the reconstruction-only and the Gram variants converge in comparable time ($\sim$5 s), indicating that the Gram-matrix alignment loss adds negligible overhead to the optimization loop. The full concept extraction pipeline thus completes in under 10 s per class for these two variants. The KL variant requires substantially longer optimization due to the per-step cost of computing the divergence over the full patch-token distribution and its gradients. Training for Sparse Autoencoders (SAEs) and Gated SAEs also requires notably more time than the baseline reconstruction variants (62.5 s and 71.4 s, respectively), though they remain faster than the KL approach. The gating mechanism in Gated SAEs introduces only a modest runtime overhead ($\sim$9 s) over standard SAEs. Notably, both SAE variants benefit from a lower peak VRAM footprint (8.0 GB). For the other variants, peak VRAM is reported for batch extraction over the entire class subset ($\sim$1,000 images), where the dominant cost is storing the activation matrix $\mathbf{A}$ in memory. At inference time, when projecting a single image onto a precomputed concept bank, the memory footprint reduces to a single forward pass through the backbone plus a matrix multiplication with W, requiring less than 1 GB of additional VRAM beyond the model weights.

E.10 Computational Complexity and Scalability to Larger Architectures

We analyze the computational cost and memory footprint of PLACE as a function of the model and extraction parameters, and empirically validate its scalability on DINOv2 ViT-L/14.

Notation. Let P denote the patch size, H the input resolution, $G_0 = H/P$ the native grid dimension, s the token grouping factor, $G_s = \lfloor G_0/s \rfloor$ the grouped grid dimension, $N_s = G_s^2$ the number of grouped tokens per image, n the number of images, d the embedding dimension, and r the dictionary rank.

Activation extraction. A single forward pass through the prefix $g_{\leq \ell}$ costs $\mathcal{O}(n \cdot G_0^2 \cdot d^2)$, dominated by the self-attention computation. This cost is shared across all methods (PLACE, SAEs, CRAFT, FACE) and is *not* part of the concept extraction itself. Token grouping via non-overlapping average pooling is $\mathcal{O}(n \cdot G_0^2 \cdot d)$ and negligible in practice.

NNDSVD initialization. The initialization requires a truncated SVD of a subsampled activation matrix $\tilde{\mathbf{A}} \in \mathbb{R}^{m \times d}$, where $m = \rho \cdot nN_s$ and $\rho \in (0, 1]$ is the subsampling ratio. Using randomized

SVD, this costs $\mathcal{O}(m \cdot d \cdot r)$. Since $r \ll d \ll m$ in all configurations, this step is dominated by the optimization phase only when ρ is large.

Per-step optimization cost. Each AdamW step involves:

1. **Reconstruction:** Computing $\hat{\mathbf{A}} = \mathbf{U}\mathbf{W}^\top$ for a mini-batch of b images costs $\mathcal{O}(b \cdot N_s \cdot d \cdot r)$, and $\mathcal{L}_{\text{rec}} = \frac{1}{2}\|\mathbf{A}_b - \hat{\mathbf{A}}_b\|_F^2$ is $\mathcal{O}(b \cdot N_s \cdot d)$.
2. **Gram loss:** Computing the Gram matrices $\mathbf{A}_b\mathbf{A}_b^\top$ and $\hat{\mathbf{A}}_b\hat{\mathbf{A}}_b^\top \in \mathbb{R}^{bN_s \times bN_s}$ costs $\mathcal{O}((bN_s)^2 \cdot d)$, and the Frobenius norm of their difference is $\mathcal{O}((bN_s)^2)$. This is the dominant cost per step.
3. **KL loss:** Requires a full forward pass through the suffix $g_{>\ell}$ and classifier h, costing $\mathcal{O}(b \cdot G_0^2 \cdot d^2 \cdot L_{\text{suf}})$ where $L_{\text{suf}} = L - \ell$. This is one reason for the $\sim$20$\times$ slowdown of the +KL variant.

The total per-step cost of PLACE (+Gram) is therefore:

$$\mathcal{O}\big(b \cdot N_s \cdot d \cdot r + (b \cdot N_s)^2 \cdot d\big), \tag{37}$$

which is *independent* of the suffix depth L_{suf} and requires no forward pass through $g_{>\ell}$ or h.

Memory footprint. Peak GPU memory is dominated by storing the full activation matrix $\mathbf{A} \in \mathbb{R}^{nN_s \times d}$ and the two Gram matrices per mini-batch (each $bN_s \times bN_s$). In float32:

$$\text{VRAM} \approx 4 \cdot \big(n \cdot N_s \cdot d + 2 \cdot (b \cdot N_s)^2\big) \text{ bytes}, \tag{38}$$

plus model weights for the prefix (loaded once for activation extraction, then discarded during optimization).

Scaling properties across ViT scales. Table 20 reports the projected computational parameters for architectures from ViT-B to ViT-G.

Table 20: Projected computational parameters for PLACE (+Gram) across ViT scales. N_s: grouped tokens per image; $|\mathbf{A}|$: activation matrix size. All estimates assume n=800, b=4, r=25, s=2.

Architecture	P	H	d	N_s	$\lvert\mathbf{A}\rvert$ (M floats)	Gram size
ViT-B/16	16	224	768	49	30.0	196×196
DINOv2 ViT-B/14	14	518	768	324	199.1	1296×1296
DINOv2 ViT-L/14	14	518	1024	324	265.4	1296×1296
DINOv2 ViT-G/14	14	518	1536	324	398.1	1296×1296

A crucial observation emerges: since DINOv2 ViT-B/14, ViT-L/14, and ViT-G/14 share the same patch size (P=14) and input resolution (H=518), they produce *identical* token grids (37×37, grouped to 18×18). Consequently, the Gram matrix computation – the dominant per-step cost – is identical across all three scales. Only the linear reconstruction term $\mathcal{O}(bN_s \cdot d \cdot r)$ and the activation storage $\mathcal{O}(nN_s \cdot d)$ grow with the embedding dimension d. Furthermore, unlike the +KL variant, whose per-step cost scales as $\mathcal{O}(b \cdot G_0^2 \cdot d^2 \cdot L_{\text{suf}})$, the +Gram variant is entirely independent of the number of remaining transformer blocks. This advantage compounds with model depth: for ViT-G (L=40), extracting at ℓ=30 would require backpropagating through 10 blocks per step under +KL, while +Gram incurs zero additional cost.

Empirical validation on DINOv2 ViT-L/14. To confirm these projections, we run the full PLACE pipeline on DINOv2 ViT-L/14 (d=1024, L=24) for two representative ImageNet classes (`church` and `tench`), using (r=25, s=2, ℓ=21, λ=1.0).

The runtime confirms the complexity analysis: the +Gram optimization cost increases only modestly from ViT-B (5.6 s, Table 19) to ViT-L (11–24 s), consistent with the linear scaling in d, while the Gram bottleneck remains unchanged. The +KL variant, by contrast, suffers an increase (118 s $\rightarrow$ 202–578 s) due to the larger suffix with d=1024.

Table 22 reports the full suite of faithfulness, sparsity, reconstruction, and stability metrics on ViT-L/14.

The results confirm that all metrics transfer consistently from ViT-B to ViT-L. The +Gram variant achieves the highest dictionary stability on both classes (W-Stab 0.938 and 0.957), confirming that the geometric constraint continues to regularize the factorization at larger scale. On `tench`, the +Gram variant attains a D_{KL} comparable to the supervised +KL reference (0.118 vs. 0.123). The

Table 21: Wall-clock runtime breakdown on DINOv2 ViT-L/14 (median over runs). The +Gram variant completes in 23–38 s total, achieving a 5–25× speedup over +KL.

Class	Method	Extract (s)	Init (s)	Optimize (s)	**Total (s)**
church	Reconstruction	9.6	2.2	13.6	25.4
	+Gram	9.4	2.1	11.1	**22.7**
	+KL	9.5	2.1	566.6	578.5
tench	Reconstruction	11.7	2.5	15.4	29.9
	+Gram	11.6	2.5	23.9	**38.0**
	+KL	11.6	2.5	188.4	202.5

Table 22: Faithfulness, reconstruction, sparsity, and dictionary stability on DINOv2 ViT-L/14 (ImageNet, r=25, ℓ=21). Higher is better: C-Ins, C-Del, C-Gini, W-Stab; lower: MSE, D_{KL}. Mean $\pm$ std over 5 runs.

Class	Method	C-Ins ↑	C-Del ↑	C-Gini ↑	MSE ↓	D_{KL} ↓	W-Stab ↑
church	Reconstruction	0.590 ± 0.023	0.784 ± 0.022	0.667 ± 0.004	0.372 ± 0.000	0.356 ± 0.005	0.894
	+Gram	0.721 ± 0.068	0.783 ± 0.032	0.704 ± 0.023	0.373 ± 0.003	0.361 ± 0.009	0.938
	+KL	0.775 ± 0.118	0.831 ± 0.025	0.720 ± 0.026	0.377 ± 0.002	0.374 ± 0.011	0.929
tench	Reconstruction	0.857 ± 0.028	0.660 ± 0.030	0.632 ± 0.002	0.328 ± 0.001	0.123 ± 0.002	0.941
	+Gram	0.894 ± 0.045	0.763 ± 0.110	0.654 ± 0.033	0.327 ± 0.004	0.118 ± 0.002	0.957
	+KL	0.937 ± 0.006	0.889 ± 0.005	0.666 ± 0.013	0.335 ± 0.001	0.123 ± 0.003	0.941

MSE–faithfulness trade-off is consistent: reconstruction-only achieves the lowest MSE while +KL achieves the best causal metrics, with +Gram positioned between the two – matching its unsupervised character.

Table 23 reports the semantic alignment between unsupervised and supervised concept banks, measured via RSVC.

Table 23: Agreement of unsupervised concept banks with the supervised (+KL) reference on DINOv2 ViT-L/14 (ImageNet, r=25, ℓ=21). Mean $\pm$ std over 5 runs.

Class	Comparison	MCS ↑	CMCS ↑
church	Reconstruction vs. +KL	0.921 ± 0.018	0.740 ± 0.006
	+Gram vs. +KL	$\mathbf{0.947 \pm 0.015}$	0.736 ± 0.010
tench	Reconstruction vs. +KL	0.918 ± 0.012	0.695 ± 0.010
	+Gram vs. +KL	$\mathbf{0.935 \pm 0.018}$	0.697 ± 0.008

Both unsupervised variants achieve higher individual concept alignment on ViT-L than on ViT-B (church: MCS 0.947 and 0.921 for +Gram and reconstruction, against 0.866 and 0.841; tench: 0.935 and 0.918, against 0.829 and 0.805; cf. Table 30), so the increase reflects model scale rather than the Gram loss. On both classes, +Gram remains above reconstruction by 0.026 and 0.017 MCS.

Summary. PLACE's computational cost is governed primarily by the *spatial* resolution of the token grid (controlled by P, H, and s) rather than the *model* scale (d, L). The empirical validation on ViT-L/14 confirms that the method scales gracefully: extraction completes in under 40 s on consumer hardware, concept alignment with the supervised reference increases at larger scale, and the +Gram variant achieves the highest dictionary stability while closely matching the supervised reference across all metrics.

E.11 Per-class faithfulness trends

We report detailed, per-class faithfulness and reconstruction metrics for the three PLACE variants across both ImageNet and COCO datasets. Table 24 presents the results for the self-supervised DINOv2 ViT-B/14 backbone, while Table 25 reports the metrics for the supervised ViT-B/16 architecture, and Table 26 for ViT-MAE B/16. Tables 27 to 29 report the SAE baselines on the same backbones.

Table 24: Per-class faithfulness and reconstruction quality for all three PLACE variants on ImageNet and COCO (DINOv2 ViT-B/14, $\ell = 9$, r=25). Mean $\pm$ std across 5 seeds.

	PLACE (rec.)					PLACE (+Gram)					PLACE (+KL)				
Class	C-Ins ↑	C-Del ↑	C-Gini ↑	MSE ↓	D_{KL} ↓	C-Ins ↑	C-Del ↑	C-Gini ↑	MSE ↓	D_{KL} ↓	C-Ins ↑	C-Del ↑	C-Gini ↑	MSE ↓	D_{KL} ↓
ImageNet															
cassette	0.422 ± 0.021	0.910 ± 0.015	0.917 ± 0.001	0.082 ± 0.000	0.591 ± 0.012	0.701 ± 0.025	0.940 ± 0.004	0.910 ± 0.003	0.083 ± 0.000	0.651 ± 0.010	0.682 ± 0.029	0.936 ± 0.005	0.913 ± 0.003	0.083 ± 0.000	0.658 ± 0.006
chainsaw	0.380 ± 0.038	0.784 ± 0.026	0.939 ± 0.001	0.096 ± 0.000	2.265 ± 0.407	0.401 ± 0.055	0.790 ± 0.039	0.926 ± 0.008	0.093 ± 0.002	1.763 ± 0.477	0.404 ± 0.061	0.805 ± 0.036	0.930 ± 0.005	0.095 ± 0.002	1.776 ± 0.460
church	0.113 ± 0.020	0.952 ± 0.004	0.930 ± 0.001	0.093 ± 0.000	1.277 ± 0.031	0.153 ± 0.041	0.962 ± 0.000	0.923 ± 0.004	0.093 ± 0.001	1.275 ± 0.094	0.130 ± 0.036	0.962 ± 0.000	0.928 ± 0.004	0.095 ± 0.001	1.326 ± 0.082
english springer	0.690 ± 0.018	0.957 ± 0.004	0.931 ± 0.001	0.101 ± 0.001	0.813 ± 0.012	0.746 ± 0.022	0.958 ± 0.002	0.918 ± 0.011	0.100 ± 0.003	0.765 ± 0.073	0.745 ± 0.023	0.958 ± 0.002	0.925 ± 0.006	0.102 ± 0.002	0.800 ± 0.052
french horn	0.076 ± 0.003	0.905 ± 0.011	0.936 ± 0.001	0.095 ± 0.000	1.196 ± 0.028	0.067 ± 0.025	0.927 ± 0.006	0.926 ± 0.011	0.094 ± 0.002	1.988 ± 0.425	0.052 ± 0.010	0.928 ± 0.004	0.928 ± 0.004	0.096 ± 0.001	2.264 ± 0.157
garbage truck	0.438 ± 0.010	0.960 ± 0.001	0.938 ± 0.000	0.091 ± 0.000	1.605 ± 0.038	0.704 ± 0.083	0.956 ± 0.001	0.923 ± 0.011	0.088 ± 0.003	1.067 ± 0.381	0.658 ± 0.021	0.955 ± 0.001	0.928 ± 0.004	0.090 ± 0.001	1.185 ± 0.183
gas pump	0.275 ± 0.017	0.712 ± 0.019	0.948 ± 0.000	0.094 ± 0.000	3.112 ± 0.122	0.286 ± 0.110	0.780 ± 0.022	0.932 ± 0.009	0.090 ± 0.003	2.540 ± 0.828	0.293 ± 0.036	0.760 ± 0.031	0.934 ± 0.003	0.091 ± 0.001	2.491 ± 0.294
golf ball	0.925 ± 0.001	0.960 ± 0.000	0.938 ± 0.000	0.092 ± 0.000	0.216 ± 0.009	0.915 ± 0.005	0.957 ± 0.002	0.927 ± 0.006	0.089 ± 0.002	0.221 ± 0.026	0.911 ± 0.003	0.957 ± 0.002	0.926 ± 0.006	0.089 ± 0.002	0.219 ± 0.030
parachute	0.831 ± 0.003	0.950 ± 0.001	0.934 ± 0.000	0.088 ± 0.000	0.539 ± 0.017	0.838 ± 0.003	0.960 ± 0.001	0.925 ± 0.005	0.086 ± 0.002	0.563 ± 0.037	0.839 ± 0.012	0.960 ± 0.000	0.921 ± 0.010	0.085 ± 0.003	0.551 ± 0.116
tench	0.521 ± 0.016	0.856 ± 0.006	0.924 ± 0.001	0.090 ± 0.001	0.435 ± 0.017	0.405 ± 0.038	0.910 ± 0.005	0.898 ± 0.014	0.087 ± 0.002	0.785 ± 0.332	0.376 ± 0.026	0.922 ± 0.004	0.911 ± 0.007	0.090 ± 0.002	1.055 ± 0.199
COCO															
hot dog	0.132 ± 0.024	0.894 ± 0.025	0.889 ± 0.001	0.077 ± 0.000	0.212 ± 0.007	0.096 ± 0.009	0.913 ± 0.018	0.899 ± 0.002	0.081 ± 0.001	0.300 ± 0.011	0.089 ± 0.013	0.915 ± 0.017	0.901 ± 0.002	0.083 ± 0.002	0.317 ± 0.023
teddy bear	0.474 ± 0.029	0.691 ± 0.044	0.908 ± 0.001	0.084 ± 0.000	1.215 ± 0.027	0.274 ± 0.028	0.776 ± 0.034	0.906 ± 0.003	0.087 ± 0.001	1.045 ± 0.063	0.285 ± 0.040	0.772 ± 0.039	0.905 ± 0.003	0.087 ± 0.002	1.032 ± 0.094
train	0.020 ± 0.007	0.852 ± 0.077	0.897 ± 0.001	0.073 ± 0.000	0.332 ± 0.014	0.041 ± 0.004	0.946 ± 0.009	0.899 ± 0.002	0.077 ± 0.001	0.354 ± 0.008	0.045 ± 0.003	0.948 ± 0.012	0.899 ± 0.002	0.077 ± 0.001	0.348 ± 0.006
umbrella	0.149 ± 0.027	0.898 ± 0.015	0.853 ± 0.001	0.077 ± 0.000	0.648 ± 0.026	0.158 ± 0.042	0.927 ± 0.020	0.853 ± 0.001	0.077 ± 0.000	0.666 ± 0.016	0.144 ± 0.032	0.910 ± 0.020	0.852 ± 0.001	0.077 ± 0.000	0.646 ± 0.020
zebra	0.251 ± 0.108	0.327 ± 0.026	0.873 ± 0.002	0.071 ± 0.000	0.408 ± 0.019	0.316 ± 0.082	0.317 ± 0.045	0.882 ± 0.002	0.076 ± 0.000	0.554 ± 0.017	0.303 ± 0.099	0.296 ± 0.051	0.879 ± 0.006	0.075 ± 0.002	0.539 ± 0.050

Table 25: Per-class faithfulness and reconstruction quality for all three PLACE variants on ImageNet and COCO (ViT-B/16, $\ell = 9$, r=25). Mean $\pm$ std across 5 seeds.

	PLACE (rec.)					PLACE (+Gram)					PLACE (+KL)				
Class	C-Ins ↑	C-Del ↑	C-Gini ↑	MSE ↓	D_{KL} ↓	C-Ins ↑	C-Del ↑	C-Gini ↑	MSE ↓	D_{KL} ↓	C-Ins ↑	C-Del ↑	C-Gini ↑	MSE ↓	D_{KL} ↓
ImageNet															
cassette	0.614 ± 0.024	0.133 ± 0.015	0.514 ± 0.008	0.265 ± 0.000	0.000 ± 0.000	0.533 ± 0.034	0.268 ± 0.024	0.579 ± 0.014	0.271 ± 0.003	0.000 ± 0.000	0.730 ± 0.021	0.234 ± 0.032	0.448 ± 0.028	0.281 ± 0.006	0.012 ± 0.007
chainsaw	0.685 ± 0.014	0.362 ± 0.079	0.618 ± 0.006	0.314 ± 0.000	0.000 ± 0.000	0.654 ± 0.026	0.432 ± 0.047	0.656 ± 0.009	0.317 ± 0.004	0.000 ± 0.000	0.765 ± 0.014	0.116 ± 0.018	0.459 ± 0.012	0.327 ± 0.001	0.027 ± 0.004
church	0.606 ± 0.032	0.434 ± 0.084	0.474 ± 0.003	0.289 ± 0.000	0.000 ± 0.000	0.707 ± 0.028	0.300 ± 0.011	0.571 ± 0.017	0.293 ± 0.003	0.000 ± 0.000	0.911 ± 0.003	0.147 ± 0.023	0.439 ± 0.014	0.302 ± 0.002	0.015 ± 0.003
english springer	0.747 ± 0.012	0.233 ± 0.045	0.536 ± 0.009	0.304 ± 0.001	0.000 ± 0.000	0.762 ± 0.007	0.104 ± 0.021	0.587 ± 0.006	0.306 ± 0.002	0.000 ± 0.000	0.736 ± 0.019	0.211 ± 0.040	0.328 ± 0.011	0.314 ± 0.001	0.016 ± 0.004
french horn	0.746 ± 0.030	0.451 ± 0.099	0.510 ± 0.013	0.306 ± 0.001	0.000 ± 0.000	0.694 ± 0.027	0.406 ± 0.073	0.605 ± 0.013	0.309 ± 0.003	0.000 ± 0.000	0.822 ± 0.056	0.363 ± 0.088	0.452 ± 0.035	0.316 ± 0.002	0.019 ± 0.009
garbage truck	0.463 ± 0.015	0.139 ± 0.047	0.394 ± 0.006	0.278 ± 0.000	0.000 ± 0.000	0.506 ± 0.007	0.090 ± 0.003	0.559 ± 0.004	0.280 ± 0.001	0.000 ± 0.000	0.503 ± 0.015	0.314 ± 0.069	0.303 ± 0.021	0.289 ± 0.003	0.020 ± 0.009
gas pump	0.691 ± 0.009	0.536 ± 0.061	0.571 ± 0.008	0.298 ± 0.000	0.000 ± 0.000	0.708 ± 0.037	0.452 ± 0.037	0.615 ± 0.017	0.301 ± 0.003	0.000 ± 0.000	0.928 ± 0.010	0.195 ± 0.036	0.479 ± 0.015	0.310 ± 0.001	0.023 ± 0.008
golf ball	0.900 ± 0.008	0.441 ± 0.035	0.620 ± 0.011	0.316 ± 0.000	0.000 ± 0.000	0.693 ± 0.012	0.812 ± 0.017	0.689 ± 0.005	0.320 ± 0.003	0.000 ± 0.000	0.764 ± 0.039	0.702 ± 0.067	0.497 ± 0.005	0.325 ± 0.001	0.012 ± 0.003
parachute	0.915 ± 0.009	0.476 ± 0.028	0.633 ± 0.008	0.258 ± 0.000	0.000 ± 0.000	0.900 ± 0.009	0.314 ± 0.016	0.698 ± 0.008	0.263 ± 0.003	0.000 ± 0.000	0.964 ± 0.006	0.165 ± 0.009	0.588 ± 0.012	0.267 ± 0.001	0.012 ± 0.003
tench	0.787 ± 0.033	0.716 ± 0.100	0.431 ± 0.014	0.296 ± 0.000	0.000 ± 0.000	0.856 ± 0.016	0.828 ± 0.020	0.587 ± 0.005	0.300 ± 0.003	0.000 ± 0.000	0.839 ± 0.013	0.682 ± 0.060	0.379 ± 0.036	0.304 ± 0.002	0.007 ± 0.003
COCO															
hot dog	0.983 ± 0.002	0.990 ± 0.005	0.577 ± 0.005	0.351 ± 0.000	0.000 ± 0.000	0.948 ± 0.005	0.990 ± 0.002	0.628 ± 0.013	0.351 ± 0.003	0.000 ± 0.000	0.925 ± 0.001	0.934 ± 0.015	0.646 ± 0.005	0.356 ± 0.001	0.003 ± 0.002
teddy bear	0.970 ± 0.001	0.965 ± 0.018	0.657 ± 0.006	0.369 ± 0.000	0.000 ± 0.000	0.959 ± 0.006	0.950 ± 0.047	0.664 ± 0.038	0.362 ± 0.002	0.000 ± 0.000	0.961 ± 0.003	0.893 ± 0.012	0.692 ± 0.027	0.369 ± 0.002	0.002 ± 0.001
train	0.996 ± 0.000	0.958 ± 0.003	0.576 ± 0.002	0.368 ± 0.000	0.000 ± 0.000	0.965 ± 0.010	0.986 ± 0.002	0.606 ± 0.047	0.359 ± 0.002	0.000 ± 0.000	0.956 ± 0.004	0.946 ± 0.010	0.684 ± 0.036	0.373 ± 0.003	0.003 ± 0.000
umbrella	0.991 ± 0.003	0.947 ± 0.005	0.704 ± 0.005	0.415 ± 0.001	0.000 ± 0.000	0.927 ± 0.017	0.985 ± 0.002	0.664 ± 0.038	0.403 ± 0.003	0.000 ± 0.000	0.933 ± 0.019	0.945 ± 0.022	0.646 ± 0.017	0.416 ± 0.003	0.006 ± 0.001
zebra	0.999 ± 0.000	0.979 ± 0.003	0.673 ± 0.021	0.282 ± 0.001	0.000 ± 0.000	0.985 ± 0.001	0.986 ± 0.001	0.711 ± 0.008	0.279 ± 0.000	0.000 ± 0.000	0.969 ± 0.004	0.981 ± 0.003	0.719 ± 0.005	0.283 ± 0.000	0.001 ± 0.000

Table 26: Per-class faithfulness and reconstruction quality for all three PLACE variants on ImageNet and COCO (ViT-MAE B/16, $\ell = 9$, r=25). Mean $\pm$ std across 5 seeds.

	PLACE (rec.)					PLACE (+Gram)					PLACE (+KL)				
Class	C-Ins ↑	C-Del ↑	C-Gini ↑	MSE ↓	D_{KL} ↓	C-Ins ↑	C-Del ↑	C-Gini ↑	MSE ↓	D_{KL} ↓	C-Ins ↑	C-Del ↑	C-Gini ↑	MSE ↓	D_{KL} ↓
ImageNet															
cassette	0.930 ± 0.018	0.465 ± 0.142	0.396 ± 0.005	1.021 ± 0.000	0.374 ± 0.002	0.953 ± 0.007	0.151 ± 0.039	0.645 ± 0.005	1.064 ± 0.003	0.434 ± 0.007	0.955 ± 0.006	0.206 ± 0.045	0.583 ± 0.022	1.035 ± 0.008	0.383 ± 0.020
chainsaw	0.942 ± 0.007	0.408 ± 0.071	0.434 ± 0.023	1.148 ± 0.001	0.461 ± 0.004	1.000 ± 0.000	0.115 ± 0.006	0.631 ± 0.009	1.168 ± 0.005	0.477 ± 0.007	1.000 ± 0.000	0.115 ± 0.009	0.643 ± 0.028	1.185 ± 0.025	0.491 ± 0.020
church	0.806 ± 0.020	0.599 ± 0.056	0.398 ± 0.007	0.936 ± 0.000	0.471 ± 0.014	0.996 ± 0.004	0.119 ± 0.020	0.652 ± 0.015	0.975 ± 0.012	0.529 ± 0.015	0.996 ± 0.002	0.168 ± 0.106	0.635 ± 0.032	0.965 ± 0.018	0.501 ± 0.028
english springer	0.813 ± 0.060	0.484 ± 0.037	0.420 ± 0.033	1.005 ± 0.001	0.387 ± 0.006	0.997 ± 0.006	0.065 ± 0.005	0.674 ± 0.010	1.042 ± 0.007	0.412 ± 0.003	0.996 ± 0.006	0.076 ± 0.007	0.635 ± 0.009	1.023 ± 0.003	0.396 ± 0.003
french horn	0.962 ± 0.004	0.639 ± 0.032	0.410 ± 0.007	1.088 ± 0.000	0.247 ± 0.003	1.000 ± 0.000	0.165 ± 0.032	0.686 ± 0.006	1.117 ± 0.004	0.286 ± 0.007	1.000 ± 0.000	0.220 ± 0.092	0.675 ± 0.023	1.116 ± 0.015	0.278 ± 0.021
garbage truck	0.862 ± 0.036	0.685 ± 0.033	0.386 ± 0.017	0.987 ± 0.001	0.314 ± 0.004	0.978 ± 0.021	0.250 ± 0.048	0.666 ± 0.017	1.018 ± 0.014	0.363 ± 0.021	0.979 ± 0.020	0.265 ± 0.049	0.663 ± 0.015	1.021 ± 0.009	0.359 ± 0.014
gas pump	0.963 ± 0.009	0.645 ± 0.014	0.395 ± 0.052	1.082 ± 0.001	0.416 ± 0.003	0.970 ± 0.002	0.234 ± 0.017	0.646 ± 0.007	1.117 ± 0.007	0.486 ± 0.013	0.971 ± 0.003	0.278 ± 0.055	0.628 ± 0.017	1.103 ± 0.017	0.451 ± 0.039
golf ball	0.939 ± 0.007	0.706 ± 0.015	0.533 ± 0.016	0.952 ± 0.001	0.066 ± 0.000	0.997 ± 0.001	0.109 ± 0.034	0.737 ± 0.005	0.983 ± 0.007	0.068 ± 0.001	0.995 ± 0.000	0.115 ± 0.041	0.737 ± 0.009	0.986 ± 0.014	0.069 ± 0.001
parachute	0.931 ± 0.010	0.686 ± 0.025	0.543 ± 0.017	0.820 ± 0.000	0.158 ± 0.003	1.000 ± 0.000	0.339 ± 0.066	0.732 ± 0.007	0.858 ± 0.012	0.176 ± 0.004	0.997 ± 0.003	0.373 ± 0.090	0.721 ± 0.025	0.857 ± 0.032	0.171 ± 0.013
tench	0.964 ± 0.023	0.544 ± 0.044	0.465 ± 0.034	0.911 ± 0.000	0.126 ± 0.003	1.000 ± 0.000	0.062 ± 0.018	0.712 ± 0.011	0.940 ± 0.012	0.153 ± 0.010	1.000 ± 0.000	0.076 ± 0.023	0.708 ± 0.017	0.940 ± 0.017	0.151 ± 0.016
COCO															
hot dog	0.201 ± 0.082	0.176 ± 0.113	0.322 ± 0.019	0.657 ± 0.000	0.070 ± 0.002	0.202 ± 0.087	0.173 ± 0.113	0.325 ± 0.015	0.657 ± 0.000	0.070 ± 0.002	0.193 ± 0.081	0.164 ± 0.115	0.325 ± 0.016	0.657 ± 0.000	0.070 ± 0.002
teddy bear	0.328 ± 0.055	0.844 ± 0.073	0.455 ± 0.028	0.575 ± 0.001	0.107 ± 0.001	0.447 ± 0.057	0.860 ± 0.054	0.605 ± 0.037	0.629 ± 0.029	0.106 ± 0.002	0.436 ± 0.017	0.864 ± 0.052	0.590 ± 0.044	0.618 ± 0.027	0.106 ± 0.001
train	0.429 ± 0.084	0.369 ± 0.175	0.315 ± 0.024	0.295 ± 0.000	0.005 ± 0.000	0.421 ± 0.101	0.371 ± 0.190	0.313 ± 0.024	0.295 ± 0.000	0.005 ± 0.000	0.429 ± 0.084	0.378 ± 0.176	0.319 ± 0.024	0.295 ± 0.000	0.004 ± 0.000
umbrella	0.352 ± 0.095	0.224 ± 0.195	0.358 ± 0.040	0.529 ± 0.000	0.086 ± 0.002	0.360 ± 0.096	0.236 ± 0.208	0.360 ± 0.040	0.529 ± 0.000	0.086 ± 0.002	0.353 ± 0.101	0.240 ± 0.209	0.362 ± 0.041	0.530 ± 0.000	0.088 ± 0.002
zebra	0.864 ± 0.063	0.548 ± 0.101	0.405 ± 0.052	0.871 ± 0.001	0.105 ± 0.003	0.851 ± 0.050	0.520 ± 0.147	0.540 ± 0.031	0.928 ± 0.035	0.104 ± 0.002	0.859 ± 0.054	0.477 ± 0.130	0.521 ± 0.034	0.908 ± 0.023	0.102 ± 0.003

Table 27: Per-class faithfulness and reconstruction quality for SAEs and Gated SAEs on ImageNet and COCO (DINOv2 ViT-B/14, $\ell = 9$). Mean $\pm$ std across 5 seeds.

	Vanilla SAE					Gated SAE				
Class	C-Ins ↑	C-Del ↑	C-Gini ↑	MSE ↓	D_{KL} ↓	C-Ins ↑	C-Del ↑	C-Gini ↑	MSE ↓	D_{KL} ↓
ImageNet										
cassette	0.688 ± 0.016	0.677 ± 0.013	0.864 ± 0.002	0.078 ± 0.000	0.519 ± 0.005	0.435 ± 0.041	0.810 ± 0.026	0.757 ± 0.031	0.071 ± 0.000	0.433 ± 0.005
chainsaw	0.646 ± 0.014	0.604 ± 0.015	0.876 ± 0.002	0.084 ± 0.000	0.671 ± 0.081	0.438 ± 0.054	0.534 ± 0.054	0.730 ± 0.016	0.078 ± 0.000	0.343 ± 0.012
church	0.174 ± 0.023	0.319 ± 0.023	0.856 ± 0.007	0.081 ± 0.000	0.929 ± 0.021	0.179 ± 0.035	0.597 ± 0.054	0.734 ± 0.051	0.075 ± 0.000	0.712 ± 0.004
springer	0.710 ± 0.050	0.646 ± 0.041	0.852 ± 0.004	0.086 ± 0.000	0.572 ± 0.003	0.365 ± 0.027	0.815 ± 0.047	0.702 ± 0.072	0.080 ± 0.000	0.523 ± 0.006
french horn	0.434 ± 0.038	0.270 ± 0.020	0.874 ± 0.001	0.082 ± 0.000	0.517 ± 0.008	0.351 ± 0.086	0.264 ± 0.058	0.720 ± 0.053	0.077 ± 0.000	0.323 ± 0.009
garbage truck	0.888 ± 0.006	0.327 ± 0.039	0.862 ± 0.001	0.076 ± 0.000	0.486 ± 0.009	0.566 ± 0.058	0.684 ± 0.120	0.681 ± 0.030	0.070 ± 0.000	0.245 ± 0.003
gas pump	0.601 ± 0.013	0.670 ± 0.017	0.883 ± 0.003	0.077 ± 0.000	0.821 ± 0.006	0.303 ± 0.039	0.624 ± 0.061	0.783 ± 0.010	0.071 ± 0.000	0.451 ± 0.008
golf ball	0.876 ± 0.024	0.769 ± 0.022	0.870 ± 0.004	0.076 ± 0.000	0.152 ± 0.001	0.688 ± 0.054	0.678 ± 0.020	0.717 ± 0.028	0.071 ± 0.000	0.128 ± 0.003
parachute	0.822 ± 0.009	0.782 ± 0.015	0.851 ± 0.005	0.072 ± 0.000	0.305 ± 0.003	0.766 ± 0.043	0.625 ± 0.033	0.745 ± 0.043	0.066 ± 0.000	0.249 ± 0.005
tench	0.742 ± 0.014	0.394 ± 0.019	0.852 ± 0.001	0.076 ± 0.000	0.262 ± 0.004	0.601 ± 0.123	0.557 ± 0.093	0.665 ± 0.053	0.072 ± 0.000	0.260 ± 0.004
COCO										
hot dog	0.166 ± 0.033	0.135 ± 0.016	0.864 ± 0.009	0.081 ± 0.000	0.237 ± 0.002	0.221 ± 0.079	0.200 ± 0.071	0.741 ± 0.038	0.074 ± 0.000	0.203 ± 0.002
teddy bear	0.172 ± 0.013	0.064 ± 0.011	0.874 ± 0.009	0.086 ± 0.000	1.172 ± 0.013	0.104 ± 0.031	0.150 ± 0.098	0.798 ± 0.037	0.080 ± 0.000	1.017 ± 0.040
train	0.056 ± 0.009	0.276 ± 0.011	0.851 ± 0.002	0.072 ± 0.000	0.352 ± 0.008	0.063 ± 0.016	0.056 ± 0.046	0.706 ± 0.066	0.066 ± 0.000	0.233 ± 0.003
umbrella	0.089 ± 0.019	0.195 ± 0.023	0.849 ± 0.004	0.082 ± 0.000	0.617 ± 0.019	0.089 ± 0.046	0.093 ± 0.048	0.705 ± 0.079	0.075 ± 0.000	0.640 ± 0.040
zebra	0.229 ± 0.041	0.707 ± 0.032	0.853 ± 0.006	0.075 ± 0.000	0.564 ± 0.055	0.529 ± 0.147	0.834 ± 0.054	0.694 ± 0.037	0.069 ± 0.000	0.302 ± 0.003

Table 28: Per-class faithfulness and reconstruction quality for SAEs and Gated SAEs on ImageNet and COCO (ViT-B/16, $\ell = 9$). Mean $\pm$ std across 5 seeds.

	Vanilla SAE					Gated SAE				
Class	C-Ins ↑	C-Del ↑	C-Gini ↑	MSE ↓	D_{KL} ↓	C-Ins ↑	C-Del ↑	C-Gini ↑	MSE ↓	D_{KL} ↓
ImageNet										
cassette	0.121 ± 0.006	0.177 ± 0.013	0.438 ± 0.003	0.266 ± 0.000	0.977 ± 0.032	0.101 ± 0.019	0.094 ± 0.031	0.258 ± 0.017	0.254 ± 0.000	0.708 ± 0.012
chainsaw	0.561 ± 0.024	0.537 ± 0.011	0.480 ± 0.008	0.305 ± 0.000	1.663 ± 0.013	0.234 ± 0.071	0.242 ± 0.070	0.295 ± 0.016	0.294 ± 0.000	1.333 ± 0.019
church	0.274 ± 0.014	0.309 ± 0.007	0.384 ± 0.008	0.282 ± 0.000	1.455 ± 0.022	0.168 ± 0.029	0.217 ± 0.055	0.265 ± 0.024	0.270 ± 0.000	1.132 ± 0.035
springer	0.584 ± 0.017	0.645 ± 0.007	0.426 ± 0.010	0.294 ± 0.000	0.744 ± 0.007	0.512 ± 0.026	0.594 ± 0.064	0.325 ± 0.015	0.284 ± 0.000	0.569 ± 0.007
french horn	0.589 ± 0.011	0.565 ± 0.006	0.420 ± 0.001	0.296 ± 0.000	1.051 ± 0.023	0.471 ± 0.036	0.512 ± 0.023	0.280 ± 0.018	0.285 ± 0.000	0.864 ± 0.016
garbage truck	0.305 ± 0.007	0.412 ± 0.010	0.389 ± 0.000	0.270 ± 0.000	1.346 ± 0.014	0.253 ± 0.037	0.197 ± 0.055	0.233 ± 0.010	0.258 ± 0.000	1.013 ± 0.009
gas pump	0.330 ± 0.003	0.440 ± 0.005	0.373 ± 0.003	0.286 ± 0.000	2.180 ± 0.014	0.176 ± 0.013	0.342 ± 0.044	0.292 ± 0.016	0.274 ± 0.000	1.747 ± 0.029
golf ball	0.798 ± 0.002	0.735 ± 0.002	0.495 ± 0.002	0.302 ± 0.000	0.488 ± 0.009	0.829 ± 0.042	0.725 ± 0.074	0.372 ± 0.020	0.291 ± 0.000	0.434 ± 0.006
parachute	0.866 ± 0.004	0.583 ± 0.039	0.531 ± 0.002	0.246 ± 0.000	0.674 ± 0.007	0.786 ± 0.035	0.602 ± 0.031	0.396 ± 0.023	0.236 ± 0.001	0.556 ± 0.014
tench	0.843 ± 0.011	0.558 ± 0.019	0.396 ± 0.004	0.285 ± 0.000	0.421 ± 0.009	0.667 ± 0.054	0.573 ± 0.071	0.305 ± 0.014	0.274 ± 0.000	0.313 ± 0.004
COCO										
hot dog	0.996 ± 0.001	0.025 ± 0.002	0.430 ± 0.003	0.335 ± 0.000	0.089 ± 0.000	0.899 ± 0.015	0.130 ± 0.039	0.256 ± 0.012	0.323 ± 0.000	0.077 ± 0.000
teddy bear	0.999 ± 0.000	0.000 ± 0.000	0.455 ± 0.002	0.342 ± 0.000	0.071 ± 0.001	0.996 ± 0.002	0.045 ± 0.042	0.278 ± 0.016	0.329 ± 0.000	0.063 ± 0.001
train	0.999 ± 0.000	0.000 ± 0.000	0.352 ± 0.002	0.329 ± 0.000	0.072 ± 0.000	0.989 ± 0.008	0.025 ± 0.009	0.226 ± 0.015	0.317 ± 0.000	0.062 ± 0.001
umbrella	0.970 ± 0.007	0.139 ± 0.051	0.393 ± 0.002	0.371 ± 0.000	0.090 ± 0.000	0.856 ± 0.042	0.291 ± 0.040	0.286 ± 0.027	0.358 ± 0.000	0.086 ± 0.001
zebra	1.000 ± 0.000	0.000 ± 0.000	0.441 ± 0.002	0.258 ± 0.000	0.011 ± 0.000	0.998 ± 0.000	0.077 ± 0.024	0.336 ± 0.017	0.245 ± 0.001	0.007 ± 0.000

Table 29: Per-class faithfulness and reconstruction quality for SAEs and Gated SAEs on ImageNet and COCO (ViT-MAE B/16, $\ell = 9$). Mean $\pm$ std across 5 seeds.

	Vanilla SAE					Gated SAE				
Class	C-Ins ↑	C-Del ↑	C-Gini ↑	MSE ↓	D_{KL} ↓	C-Ins ↑	C-Del ↑	C-Gini ↑	MSE ↓	D_{KL} ↓
ImageNet										
cassette	0.902 ± 0.018	0.441 ± 0.028	0.190 ± 0.014	1.021 ± 0.000	0.391 ± 0.006	0.557 ± 0.058	0.423 ± 0.091	0.265 ± 0.024	1.021 ± 0.009	0.373 ± 0.010
chainsaw	0.993 ± 0.007	0.274 ± 0.114	0.228 ± 0.007	1.127 ± 0.000	0.438 ± 0.004	0.515 ± 0.048	0.387 ± 0.033	0.220 ± 0.047	1.124 ± 0.009	0.413 ± 0.014
church	0.714 ± 0.069	0.212 ± 0.054	0.196 ± 0.002	0.921 ± 0.000	0.488 ± 0.001	0.437 ± 0.047	0.468 ± 0.038	0.268 ± 0.065	0.927 ± 0.019	0.465 ± 0.021
springer	0.967 ± 0.012	0.137 ± 0.086	0.221 ± 0.001	0.984 ± 0.000	0.428 ± 0.004	0.681 ± 0.081	0.442 ± 0.078	0.276 ± 0.048	0.993 ± 0.015	0.386 ± 0.009
french horn	1.000 ± 0.000	0.004 ± 0.001	0.234 ± 0.003	1.067 ± 0.000	0.243 ± 0.001	0.920 ± 0.055	0.525 ± 0.066	0.284 ± 0.029	1.075 ± 0.011	0.229 ± 0.005
garbage truck	0.958 ± 0.029	0.436 ± 0.059	0.210 ± 0.010	0.970 ± 0.000	0.307 ± 0.002	0.668 ± 0.083	0.442 ± 0.083	0.234 ± 0.024	0.965 ± 0.009	0.282 ± 0.006
gas pump	0.864 ± 0.002	0.528 ± 0.008	0.209 ± 0.005	1.059 ± 0.000	0.413 ± 0.005	0.795 ± 0.121	0.441 ± 0.062	0.254 ± 0.038	1.065 ± 0.009	0.371 ± 0.004
golf ball	0.972 ± 0.004	0.377 ± 0.052	0.319 ± 0.009	0.932 ± 0.000	0.062 ± 0.001	0.967 ± 0.010	0.601 ± 0.063	0.332 ± 0.021	0.924 ± 0.009	0.058 ± 0.001
parachute	0.991 ± 0.006	0.254 ± 0.013	0.290 ± 0.006	0.799 ± 0.000	0.152 ± 0.002	0.849 ± 0.066	0.414 ± 0.078	0.336 ± 0.052	0.799 ± 0.011	0.147 ± 0.002
tench	1.000 ± 0.000	0.003 ± 0.003	0.278 ± 0.006	0.891 ± 0.000	0.126 ± 0.001	0.945 ± 0.051	0.613 ± 0.086	0.283 ± 0.013	0.901 ± 0.004	0.115 ± 0.002
COCO										
hot dog	0.491 ± 0.158	0.443 ± 0.084	0.200 ± 0.009	0.667 ± 0.000	0.076 ± 0.000	0.390 ± 0.108	0.150 ± 0.070	0.246 ± 0.045	0.658 ± 0.009	0.070 ± 0.003
teddy bear	0.250 ± 0.026	0.480 ± 0.057	0.274 ± 0.002	0.582 ± 0.000	0.104 ± 0.001	0.498 ± 0.165	0.165 ± 0.210	0.349 ± 0.052	0.580 ± 0.009	0.104 ± 0.006
train	1.000 ± 0.000	0.012 ± 0.025	0.164 ± 0.023	0.305 ± 0.000	0.006 ± 0.000	0.887 ± 0.108	0.532 ± 0.337	0.378 ± 0.029	0.367 ± 0.012	0.006 ± 0.001
umbrella	0.813 ± 0.056	0.146 ± 0.067	0.202 ± 0.012	0.541 ± 0.000	0.092 ± 0.002	0.336 ± 0.080	0.499 ± 0.146	0.320 ± 0.020	0.572 ± 0.013	0.121 ± 0.007
zebra	0.902 ± 0.043	0.273 ± 0.044	0.157 ± 0.007	0.875 ± 0.000	0.098 ± 0.002	0.495 ± 0.169	0.729 ± 0.090	0.246 ± 0.041	0.864 ± 0.014	0.108 ± 0.006

Discussion. The results demonstrate distinct behavioral trends across the three transformer backbones, particularly when comparing the PLACE framework against Sparse Autoencoder (SAE) baselines.

For the self-supervised **DINOv2** model (Tables 24 and 27), +Gram has a higher mean C-Ins than the reconstruction-only baseline on 7 of 10 ImageNet classes and 3 of 5 COCO classes, and a higher mean C-Del on 8 of 10 and 4 of 5; these per-class differences are not tested for significance and several lie within one standard deviation. Furthermore, the D_{KL} of the +Gram variant is close to that of the supervised reference (+KL variant), although on this backbone reconstruction alone attains a comparable D_{KL}. Compared to SAE baselines, PLACE attains higher sparsity at a modest cost in reconstruction. Gated and Vanilla SAEs achieve lower reconstruction errors (MSE $\approx$0.07 vs. $\approx$0.09) and, on most classes, lower D_{KL}, but lower concept sparsity (C-Gini $0.85-0.88$ for Vanilla and $0.67-0.80$ for Gated SAEs). PLACE exceeds 0.89 C-Gini on all ImageNet classes; on COCO, its C-Gini ($0.85-0.91$) is close to that of Vanilla SAEs, consistent with the non-significant difference in Table 1.

Conversely, for the supervised **ViT-B/16** architecture (Tables 25 and 28), the supervised +KL variant yields the highest C-Ins on 6 of 10 ImageNet classes while keeping D_{KL} below 0.03. The label-free +Gram variant keeps D_{KL} below 10^{-3} and attains higher C-Gini than +KL on all ImageNet classes. We explicitly note that for this specific supervised backbone, the reconstruction-only PLACE baseline achieves a KL divergence less than 10^{-3} (reported as 0.000 due to rounding), indicating that the subspace captured by pure MSE already aligns with the classifier's decision boundaries. On ImageNet, SAEs show higher per-class D_{KL} ($0.3-2.2$) and lower C-Gini ($0.23-0.53$) than PLACE; the C-Gini difference is significant, whereas the D_{KL} difference is not, owing to its large variance across classes (Section E.15). Notably, we observe a systematic architectural difference: supervised ViT-B/16 tends to exhibit remarkably high C-Ins values coupled with comparatively lower C-Del scores, suggesting it encodes class-specific features with a higher degree of redundancy than self-supervised models.

Finally, the self-supervised **ViT-MAE B/16** architecture (Tables 26 and 29) presents a different behavioral profile. On ImageNet, the +Gram and +KL variants behave similarly to each other but differ markedly from reconstruction: they raise C-Gini (from $0.39-0.54$ to $0.58-0.74$) and C-Ins (to

$0.95-1.00$), while C-Del drops (from $0.41-0.71$ to $0.06-0.37$). On COCO, the three variants are close on three of five classes. Across ImageNet, ViT-MAE thus yields high C-Ins alongside moderate-to-low C-Del, a redundancy pattern somewhat echoing the supervised ViT-B/16, but accompanied by higher reconstruction errors (MSE ≈ 1.0) and D_{KL} ($0.07-0.53$). Vanilla and Gated SAEs evaluated on ViT-MAE exhibit an almost complete collapse in concept sparsity across both datasets, with C-Gini frequently falling to ~ 0.20. While PLACE also exhibits high C-Del variance on some COCO classes (e.g., `train` and `umbrella`), it attains higher C-Gini than the SAE baselines on all ImageNet classes and on four of five COCO classes. This is consistent with masked autoencoding producing dense, distributed representations that resist sparse linear disentanglement, which affects SAEs more than PLACE.

E.12 Per-class agreement scores

This section provides the comprehensive, class-by-class breakdown of the alignment metrics summarized in Section 4.3 (Table 2). We evaluate how closely the unsupervised concept banks extracted via pure reconstruction and PLACE (+Gram) recover the supervised representation subspace (+KL) on both DINOv2 ViT-B/14 and ViT-MAE.

Alignment is measured via RSVC [37] using Maximum Cosine Similarity (MCS) for individual vector alignment and CMCS (Pearson correlation of Lasso predictions) for broader subspace span. Values are reported as mean $\pm$ std over 5 seeds.

As shown in Table 30, the effect of geometric regularization differs between the backbones. For DINOv2, PLACE (+Gram) obtains a higher mean MCS than pure reconstruction in 9 out of 10 classes, although most per-class differences lie within one standard deviation.

Conversely, for ViT-MAE, pure reconstruction already achieves exceptionally high baseline alignment scores (often reaching an MCS > 0.90), leaving much less headroom for geometric refinement. Consequently, the $+$Gram regularization improves individual concept alignment in only a minority of classes (e.g., `church`, `garbage_truck`, `cassette`). This suggests that, in the class-conditioned setting, the masked autoencoder's representation space is less responsive to the explicit geometric constraint than the contrastive DINOv2 space.

Table 30: Per-class agreement of unsupervised concept banks with the supervised (+KL) reference on ImageNet for both DINOv2 and ViT-MAE. Higher values indicate stronger alignment.

	DINOv2 (ViT-B/14)				ViT-MAE (ViT-B/16)			
	Reconstruction vs. +KL		+Gram vs. +KL		Reconstruction vs. +KL		+Gram vs. +KL	
Class	MCS ↑	CMCS ↑	MCS ↑	CMCS ↑	MCS ↑	CMCS ↑	MCS ↑	CMCS ↑
`cassette`	0.859 ± 0.018	0.754 ± 0.016	0.875 ± 0.010	0.747 ± 0.014	0.900 ± 0.014	0.867 ± 0.011	0.905 ± 0.019	0.862 ± 0.011
`chainsaw`	0.841 ± 0.019	0.707 ± 0.007	0.828 ± 0.011	0.687 ± 0.007	0.905 ± 0.025	0.827 ± 0.014	0.903 ± 0.027	0.816 ± 0.017
`church`	0.841 ± 0.014	0.712 ± 0.019	0.866 ± 0.007	0.703 ± 0.018	0.907 ± 0.047	0.838 ± 0.017	0.942 ± 0.014	0.838 ± 0.018
`english_springer`	0.822 ± 0.013	0.760 ± 0.009	0.863 ± 0.011	0.750 ± 0.009	0.911 ± 0.033	0.862 ± 0.011	0.902 ± 0.027	0.858 ± 0.011
`french_horn`	0.830 ± 0.017	0.719 ± 0.012	0.849 ± 0.011	0.705 ± 0.014	0.925 ± 0.016	0.871 ± 0.007	0.922 ± 0.021	0.863 ± 0.008
`garbage_truck`	0.819 ± 0.007	0.697 ± 0.007	0.843 ± 0.007	0.687 ± 0.011	0.879 ± 0.033	0.847 ± 0.005	0.912 ± 0.012	0.843 ± 0.005
`gas_pump`	0.775 ± 0.036	0.663 ± 0.011	0.798 ± 0.038	0.658 ± 0.009	0.953 ± 0.014	0.869 ± 0.012	0.940 ± 0.027	0.850 ± 0.013
`golf_ball`	0.909 ± 0.017	0.794 ± 0.008	0.912 ± 0.016	0.782 ± 0.011	0.954 ± 0.013	0.878 ± 0.008	0.948 ± 0.019	0.874 ± 0.008
`parachute`	0.902 ± 0.006	0.777 ± 0.007	0.908 ± 0.005	0.774 ± 0.009	0.927 ± 0.021	0.889 ± 0.004	0.918 ± 0.026	0.881 ± 0.004
`tench`	0.805 ± 0.022	0.734 ± 0.016	0.829 ± 0.036	0.727 ± 0.014	0.946 ± 0.014	0.878 ± 0.005	0.940 ± 0.026	0.870 ± 0.006

E.13 Robustness to Image Perturbations

In this section, we analyze the structural resilience of PLACE against input signal degradations. We verify whether the extracted semantic directions and their predictive faithfulness remain stable even in non-ideal, real-world scenarios.

Experimental Setup. Following the ImageNet-C protocol [31], we apply five orthogonal perturbations (Gaussian Noise, Gaussian Blur, JPEG Compression, Contrast, and Pixelate) at five increasing severity levels ($L \in \{1, \dots, 5\}$). We do not filter the images to execute the algorithm only on correctly classified ones; avoiding this filtering prevents dataset reduction and ensures we strictly isolate the effect of the noise perturbations on the representation geometry.

Discussion. The results in Table 31 highlight a strength of PLACE: the stability of the representation and the uncoupling of semantic faithfulness from structural noise. Across almost all perturbations, dictionary stability (W_{stab}) remains strong (> 0.88), and activation sparsity (C-Gini) is practically

Table 31: Comprehensive Robustness Analysis with stability and faithfulness metrics across five independent perturbations. Mean baseline values for clean images: MSE = 0.0944, $W_{stab} = 0.9402$, C-Gini $= 0.9269$, $D_{KL} = 1.3829$, $\mathcal{L}_{\text{gram}} = 0.0005$. We report results on DINOv2 ViT-B/14 (MSE+Gram, ImageNet *church*, r=25, s=2), mean $\pm$ std over five seeds.

Perturbation	Metric	L1	L2	L3	L4	L5
Gaussian Noise	MSE	0.0935 ± 0.0022	0.0930 ± 0.0010	0.0903 ± 0.0012	0.0862 ± 0.0014	0.0780 ± 0.0007
	W_{stab}	0.9097	0.9480	0.9288	0.8836	0.8889
	C-Gini	0.9270 ± 0.0092	0.9323 ± 0.0032	0.9288 ± 0.0038	0.9249 ± 0.0053	0.9273 ± 0.0035
	D_{KL}	1.2902 ± 0.1526	1.2518 ± 0.0485	1.0652 ± 0.0776	1.0010 ± 0.0694	0.9120 ± 0.0388
	$\mathcal{L}_{\text{gram}}$	0.0004 ± 0.0001	0.0005 ± 0.0000	0.0004 ± 0.0000	0.0004 ± 0.0001	0.0008 ± 0.0002
Gaussian Blur	MSE	0.0952 ± 0.0003	0.0931 ± 0.0017	0.0930 ± 0.0017	0.0935 ± 0.0019	0.0990 ± 0.0017
	W_{stab}	0.9543	0.9483	0.9686	0.9312	0.9491
	C-Gini	0.9306 ± 0.0013	0.9244 ± 0.0060	0.9280 ± 0.0057	0.9315 ± 0.0085	0.9268 ± 0.0066
	D_{KL}	1.4464 ± 0.0323	1.3124 ± 0.0885	1.2385 ± 0.1189	0.9192 ± 0.0939	0.7228 ± 0.0448
	$\mathcal{L}_{\text{gram}}$	0.0005 ± 0.0000	0.0005 ± 0.0001	0.0004 ± 0.0000	0.0004 ± 0.0000	0.0004 ± 0.0001
JPEG Comp.	MSE	0.0959 ± 0.0002	0.0944 ± 0.0016	0.0948 ± 0.0022	0.0947 ± 0.0014	0.0895 ± 0.0010
	W_{stab}	0.9437	0.9607	0.9465	0.9432	0.9445
	C-Gini	0.9315 ± 0.0010	0.9259 ± 0.0062	0.9264 ± 0.0098	0.9288 ± 0.0055	0.9139 ± 0.0069
	D_{KL}	1.4505 ± 0.0300	1.3003 ± 0.0943	1.2801 ± 0.1347	1.2554 ± 0.0604	1.0057 ± 0.0621
	$\mathcal{L}_{\text{gram}}$	0.0004 ± 0.0001	0.0004 ± 0.0000	0.0005 ± 0.0001	0.0005 ± 0.0001	0.0006 ± 0.0001
Contrast	MSE	0.0934 ± 0.0016	0.0937 ± 0.0013	0.0926 ± 0.0025	0.0935 ± 0.0021	0.0922 ± 0.0004
	W_{stab}	0.9543	0.9672	0.9529	0.9459	0.9689
	C-Gini	0.9234 ± 0.0063	0.9257 ± 0.0051	0.9196 ± 0.0135	0.9282 ± 0.0078	0.9303 ± 0.0017
	D_{KL}	1.3018 ± 0.1076	1.3533 ± 0.0848	1.2708 ± 0.1883	1.3685 ± 0.1358	1.3094 ± 0.0375
	$\mathcal{L}_{\text{gram}}$	0.0005 ± 0.0001	0.0005 ± 0.0000	0.0006 ± 0.0001	0.0006 ± 0.0001	0.0006 ± 0.0001
Pixelate	MSE	0.0939 ± 0.0017	0.0948 ± 0.0017	0.0924 ± 0.0015	0.0801 ± 0.0016	0.0787 ± 0.0012
	W_{stab}	0.9457	0.9890	0.9926	0.9242	0.9233
	C-Gini	0.9275 ± 0.0066	0.9285 ± 0.0062	0.9306 ± 0.0045	0.9243 ± 0.0076	0.9203 ± 0.0072
	D_{KL}	1.1853 ± 0.0891	1.2319 ± 0.1203	1.3160 ± 0.0763	0.6125 ± 0.0504	0.4480 ± 0.0470
	$\mathcal{L}_{\text{gram}}$	0.0004 ± 0.0001	0.0004 ± 0.0000	0.0004 ± 0.0000	0.0007 ± 0.0002	0.0007 ± 0.0001

unfazed. Interestingly, rather than deteriorating, the predictive faithfulness (D_{KL}) generally tightens (decreases) or remains stable relative to the clean baseline as perturbation severity increases, particularly in noise, blur, and pixelation scenarios. Meanwhile, reconstruction metrics (MSE and $\mathcal{L}_{\text{gram}}$) exhibit only constrained variance. This demonstrates that our localized geometric constraints successfully anchor the extraction process to the model's intrinsic abstractions, preventing the concepts from collapsing into visual noise, and confirming that the extracted concepts remain functionally robust in downstream predictive influence.

E.14 Task-agnostic setting: full results

Tables 32 and 33 report the full metrics and dictionary stability for the task-agnostic setting of Section 4.4: a single concept bank per dataset (r=100), learned from 200 unlabeled images per run sampled from the union of the evaluated classes, over 20 runs. Labels and classifier outputs are not used by the reconstruction or Gram objectives; the +KL variant is computed only as a reference.

E.15 Statistical significance of the benchmark comparison

We compare the top-performing PLACE variant per metric with the strongest sparse autoencoder baseline of the same backbone in Table 1, using two-sided Welch's t-tests over 5 seeds, and apply the Holm–Bonferroni correction across the complete family of comparisons ($\alpha = 0.05$). Table 34 reports the adjusted p-values and the statistically supported outcome.

F Evaluation Metrics Description

To rigorously evaluate the quality, faithfulness, and interpretability of the extracted concepts, we employ the following standard metrics. Let $f(\cdot)$ denote the classification model, $A \in \mathbb{R}^{N \times D}$ the original patch-token activations, and $U \in \mathbb{R}^{N \times r}$, $W \in \mathbb{R}^{D \times r}$ the extracted concept activation and dictionary matrices, respectively, such that $A \approx UW^\top$.

Concept Insertion (C-Ins): Measures the ability of the extracted concepts to independently drive the model's prediction. Starting from a completely masked image, we progressively reveal the input

Table 32: Task-agnostic setting: reconstruction, sparsity, predictive consistency, and Gram alignment (r=100). Mean $\pm$ std over 20 runs.

Backbone	Dataset	Method	MSE ↓	C-Gini ↑	D_{KL} ↓	$\mathcal{L}_{\text{gram}}$ ↓
DINOv2	ImageNet	PLACE (reconstruction)	0.128 ± 0.001	0.987 ± 0.002	4.357 ± 0.133	0.0027 ± 0.0004
		PLACE (+Gram)	0.135 ± 0.002	0.985 ± 0.003	4.903 ± 0.082	0.0014 ± 0.0003
		PLACE (+KL)	0.123 ± 0.003	0.961 ± 0.004	0.071 ± 0.013	0.0064 ± 0.0010
		SAE	0.076 ± 0.002	0.914 ± 0.003	0.798 ± 0.021	0.0008 ± 0.0002
		Gated SAE	0.067 ± 0.002	0.868 ± 0.009	0.553 ± 0.006	0.0004 ± 0.0002
	COCO	PLACE (reconstruction)	0.107 ± 0.002	0.982 ± 0.002	0.125 ± 0.010	0.0009 ± 0.0002
		PLACE (+Gram)	0.112 ± 0.002	0.978 ± 0.004	0.326 ± 0.042	0.0007 ± 0.0002
		PLACE (+KL)	0.111 ± 0.003	0.971 ± 0.007	0.003 ± 0.002	0.0022 ± 0.0005
		SAE	0.075 ± 0.002	0.918 ± 0.005	0.062 ± 0.002	0.0008 ± 0.0001
		Gated SAE	0.065 ± 0.002	0.874 ± 0.007	0.048 ± 0.001	0.0003 ± 0.0002
ViT-MAE	ImageNet	PLACE (reconstruction)	0.965 ± 0.002	0.511 ± 0.009	0.467 ± 0.005	0.0052 ± 0.0002
		PLACE (+Gram)	0.986 ± 0.006	0.806 ± 0.011	0.429 ± 0.014	0.0052 ± 0.0004
		PLACE (+KL)	0.998 ± 0.005	0.794 ± 0.008	0.032 ± 0.002	0.0083 ± 0.0008
		SAE	0.547 ± 0.002	0.426 ± 0.006	0.545 ± 0.008	0.0026 ± 0.0001
		Gated SAE	0.523 ± 0.010	0.558 ± 0.014	0.530 ± 0.018	0.0006 ± 0.0002
	COCO	PLACE (reconstruction)	0.545 ± 0.002	0.543 ± 0.007	0.070 ± 0.001	0.0024 ± 0.0001
		PLACE (+Gram)	0.569 ± 0.009	0.773 ± 0.006	0.074 ± 0.002	0.0031 ± 0.0006
		PLACE (+KL)	0.585 ± 0.017	0.770 ± 0.007	0.004 ± 0.001	0.0047 ± 0.0012
		SAE	0.501 ± 0.002	0.444 ± 0.011	0.067 ± 0.001	0.0029 ± 0.0002
		Gated SAE	0.487 ± 0.012	0.554 ± 0.017	0.070 ± 0.002	0.0007 ± 0.0002

Table 33: Task-agnostic setting: dictionary stability (W-Stab) across runs.

Backbone	Dataset	Reconstruction	+Gram	+KL
DINOv2	ImageNet	0.813	0.871	0.900
	COCO	0.727	0.833	0.857
ViT-MAE	ImageNet	0.854	0.881	0.886
	COCO	0.747	0.865	0.866

Table 34: Holm-adjusted outcomes of the PLACE versus SAE comparisons in Table 1: two-sided Welch's t-tests (5 seeds) between the top-performing PLACE variant and the strongest SAE baseline per metric and configuration, Holm–Bonferroni corrected across the complete family of comparisons ($\alpha = 0.05$). For rows grouping several non-significant configurations, the smallest adjusted p-value in the group is reported. n.s.: not significant after correction.

Metric	Dataset and backbone	Supported result	p (Holm)
C-Gini	ImageNet: DINOv2	PLACE	0.006
C-Gini	ImageNet: ViT-B/16	PLACE	0.036
C-Gini	ImageNet: ViT-MAE	PLACE	$< 10^{-4}$
C-Gini	COCO: ViT-B/16	PLACE	0.002
C-Gini	COCO: DINOv2, ViT-MAE	n.s.	≥ 0.127
C-Ins	All six configurations	n.s.	1.000
C-Del	COCO: ViT-B/16	PLACE	≤ 0.01
C-Del	ImageNet: all three; COCO: DINOv2, ViT-MAE	n.s.	≥ 0.192
MSE	ImageNet: DINOv2	Gated SAE (n.s. vs. SAE)	0.014
D_{KL}	ImageNet: DINOv2	Gated SAE (n.s. vs. SAE)	0.007
MSE, D_{KL}	Remaining five configurations	n.s.	≥ 0.192

patches corresponding to the highest values in U. For K insertion steps, let $x_{\text{ins}}^{(k)}$ be the image with the top k-th fraction of patches revealed, and y the target class. C-Ins is computed as the Area Under the Curve (AUC) of the target-class probability:

$$\text{C-Ins} = \frac{1}{K}\sum_{k=1}^{K} p(y \mid x_{\text{ins}}^{(k)}) \tag{39}$$

A higher score indicates that the concepts capture the core features necessary for classification.

Concept Deletion (C-Del): Measures the model's causal reliance on the extracted concepts. Conversely to C-Ins, we progressively mask the top-activating patches. Let $x_{\text{del}}^{(k)}$ be the image with the top k-th fraction of patches masked. Following FACE [7], we report $1 - \text{AUC}$, so a higher score indicates a sharper drop in confidence:

$$\text{C-Del} = 1 - \left(\frac{1}{K}\sum_{k=1}^{K} p(y \mid x_{\text{del}}^{(k)})\right) \tag{40}$$

Concept Gini Sparsity (C-Gini): Quantifies the monosemanticity and distinctiveness of the extracted concepts. For a given concept activation vector $\mathbf{u}_j \in \mathbb{R}^N$ (the j-th column of the coefficient matrix U, representing the magnitude of concept c_j across the N spatial patch tokens), the spatial Gini coefficient is computed as:

$$G(\mathbf{u}_j) = \frac{\sum_{i=1}^{N}\sum_{m=1}^{N}|u_{i,j} - u_{m,j}|}{2N\sum_{i=1}^{N} u_{i,j}} \tag{41}$$

We report the mean C-Gini across all r concepts. The Gini index operates as a direct quantitative proxy for monosemanticity within the spatial geometry of Vision Transformers. Because interpretable visual features, such as isolated object parts, are inherently localized, a monosemantic direction in the latent space naturally yields a spatially sparse activation map. Therefore, a high C-Gini score (approaching 1.0) indicates that the extracted concept does not collapse into a polysemantic mixture of global statistics (e.g., activating densely across the entire scene or distributed textures), but instead fires selectively on a highly specific and isolated subset of visual features.

Reconstruction Error (MSE): Evaluates the geometric fidelity of the low-rank factorization, computed as the Mean Squared Error between the original activations and the reconstructed space:

$$\text{MSE} = \frac{1}{N \cdot D}\|A - UW^\top\|_F^2 \tag{42}$$

Predictive Divergence (D_{KL}): Measures the downstream predictive consistency. It is the Kullback-Leibler divergence between the class probability distribution produced by the original frozen model and the distribution produced when intermediate activations are replaced by their reconstruction $\hat{A} = UW^\top$:

$$D_{\text{KL}} = \sum_{c=1}^{C} p_c(f(A)) \log\left(\frac{p_c(f(A))}{p_c(f(UW^\top))}\right) \tag{43}$$

Maximum Cosine Similarity (MCS): Evaluates representation subspace agreement between our unsupervised dictionary W and a supervised reference dictionary $\hat{W}$ as used in RSVC [37]. We find the optimal assignment via the Hungarian algorithm (permutation π):

$$\text{MCS}(W, \hat{W}) = \frac{1}{r}\max_{\pi \in \Pi}\sum_{i=1}^{r}\frac{\mathbf{w}_i^\top \hat{\mathbf{w}}_{\pi(i)}}{\|\mathbf{w}_i\|_2\|\hat{\mathbf{w}}_{\pi(i)}\|_2} \tag{44}$$

Subspace Correlation (CMCS): Assesses the broader representational subspace agreement. It is the Pearson correlation coefficient ρ of the predictions made by Lasso regression models trained on the respective concept activation spaces, used in the RSVC framework [37].

Dictionary Stability (W-Stab): Evaluates the reproducibility of the factorization across different random initialization seeds. For two dictionaries $W^{(1)}$ and $W^{(2)}$ extracted from independent runs, it is computed exactly as the MCS equation above, verifying that the extraction robustly converges to consistent concept directions.

G Broader Impact

PLACE is a diagnostic tool for understanding the internal representations of Vision Foundation Models. On the positive side, faithful concept-based explanations can support safety auditing and trust calibration when deploying VFMs in high-stakes settings such as medical imaging or autonomous systems, by revealing which visual patterns the model relies on. Making such tools applicable without classifier supervision broadens their reach to the growing class of foundation models that are deployed as general-purpose feature extractors with no fixed downstream task.

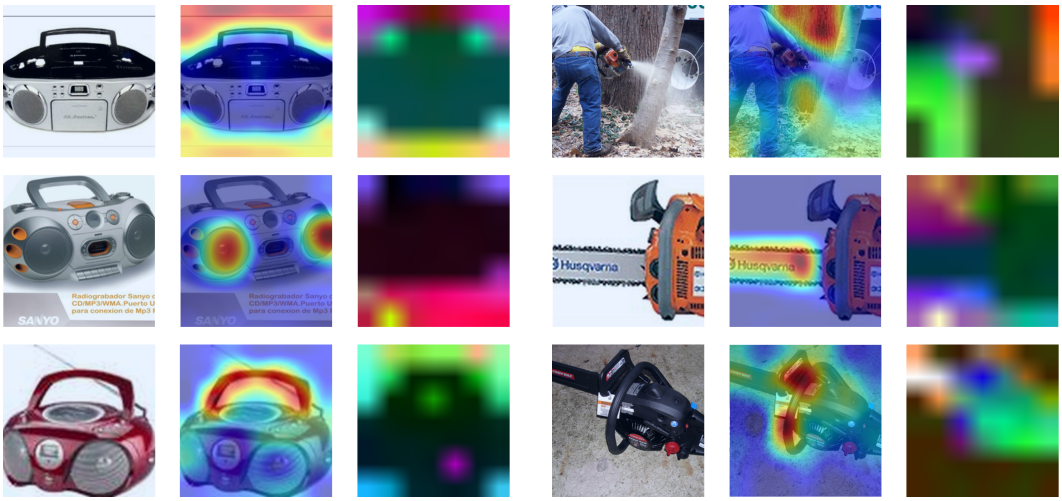
On the negative side, detailed knowledge of a model's concept structure could be exploited to craft targeted adversarial perturbations that manipulate specific concept activations while remaining imperceptible, or to engineer inputs that suppress safety-relevant concepts. More broadly, concept-based explanations may convey a false sense of transparency: users may over-trust a model because its explanations appear interpretable, even when faithfulness metrics indicate imperfect alignment. We emphasise that PLACE provides a decomposition of intermediate activations, not a causal account of the model's decision process, and should be used as one component of a broader evaluation pipeline rather than as a standalone guarantee of model reliability.

H Additional Qualitative Results

In this section, we present extended visualizations for all 10 ImageNet classes used in our evaluation to demonstrate the consistency of our approach across diverse semantic categories. For each class, we show three concepts extracted from the supervised `ViT-B/16` model using the PLACE (+Gram) loss at layer $\ell = 9$.

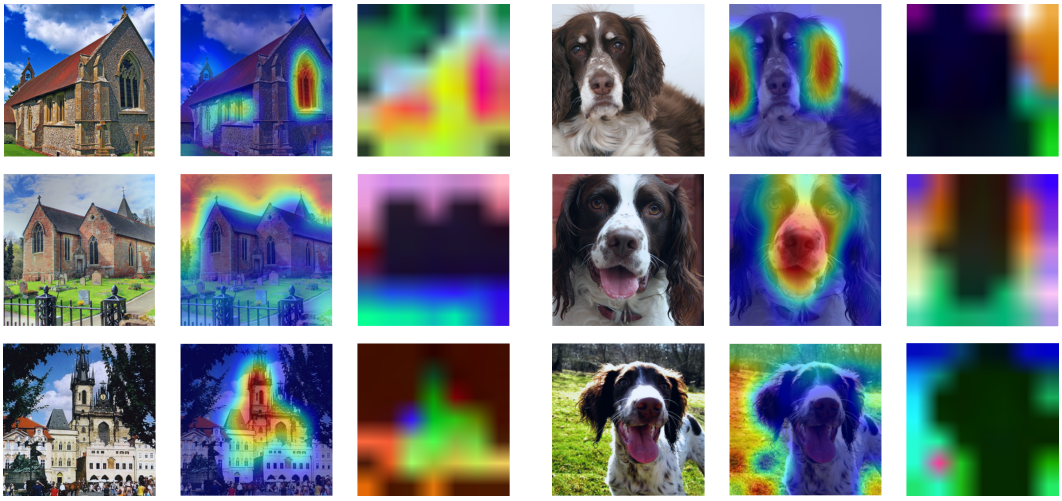
Each class panel displays the concepts with the following views: the original image (highlighting the highest-activating region), the spatial heatmap produced by the coefficient matrix $\mathbf{U}$, and the PCA projection of the original patch-token activations as described in Appendix D.3.

Object and Context Disentanglement. Crucially, our qualitative analysis reveals that PLACE does not merely act as a foreground segmenter. Because the extraction objective uses no labels, the Semi-NMF factorization captures the semantic composition of the scene, separating the primary object from its contextual environment. As highlighted in the figures below, the top concepts frequently isolate interacting subjects (e.g., humans holding objects), specific backgrounds, skies, and terrain, demonstrating the richness of the discovered geometric subspace.



(a) Class: `cassette` (b) Class: `chainsaw`

Figure 5: Top-3 highest-activating concepts for evaluated ImageNet classes (Part 1/5). Notably, for the `cassette` class, the first concept captures the background. For the `chainsaw`, the factorization isolates the interacting environment (a log/tree in the first concept) separately from the tool itself.



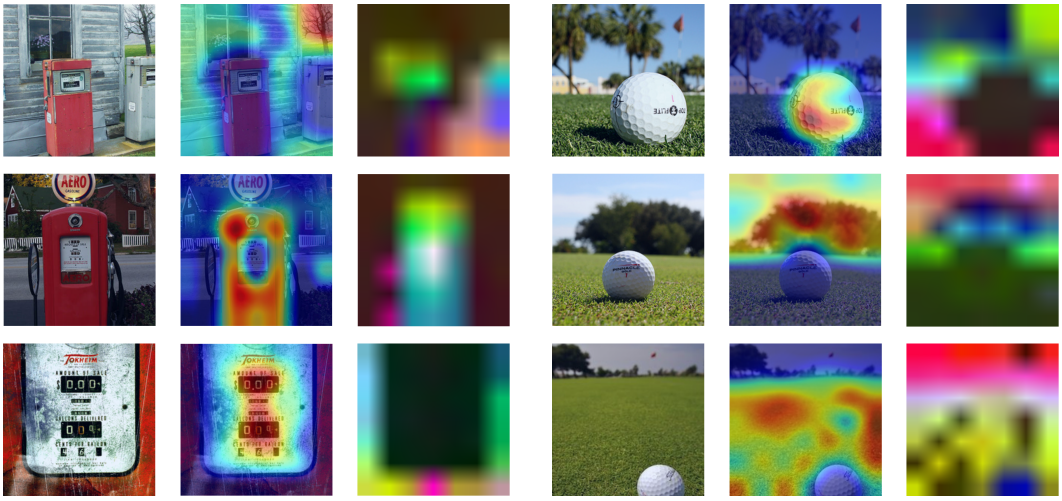
(a) Class: `church` (b) Class: `english_springer`

Figure 6: Top-3 highest-activating concepts (Part 2/5). Here, PLACE clearly disentangles the main subject from its environment: the second concept of `church` isolates the sky, while the third concept of `english_springer` isolates the lawn.



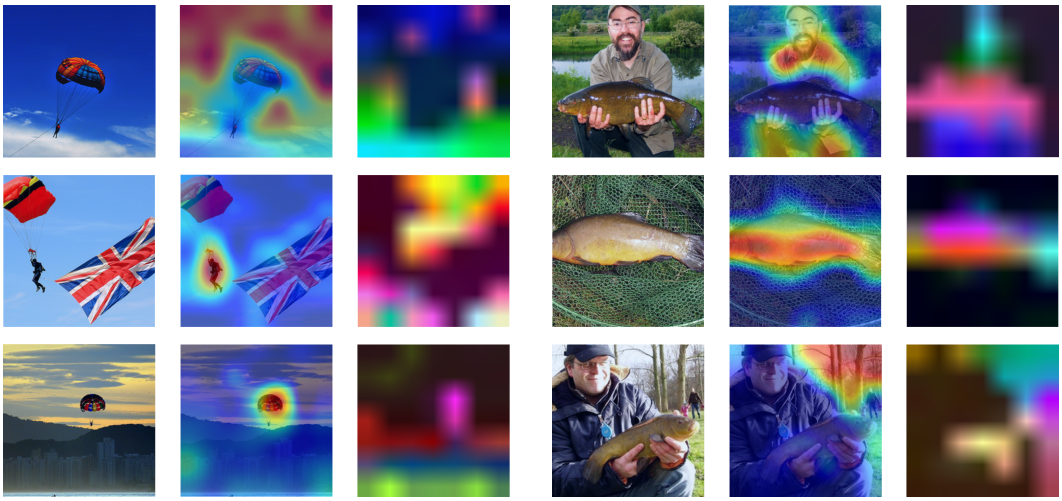
(a) Class: `french_horn` (b) Class: `garbage_truck`

Figure 7: Top-3 highest-activating concepts (Part 3/5). The decomposition successfully captures complex interactions and backgrounds: the first concept of `french_horn` focuses on the person playing the instrument, while a top concept in `garbage_truck` isolates the background trees rather than the vehicle itself.



(a) Class: `gas_pump` (b) Class: `golf_ball`

Figure 8: Top-3 highest-activating concepts (Part 4/5). Contextual separation is prominent: `gas_pump` features background foliage in its components, while `golf_ball` separates the core object (the ball) from the surrounding environmental elements like trees and the lawn.



(a) Class: `parachute` (b) Class: `tench`

Figure 9: Top-3 highest-activating concepts (Part 5/5). The decomposition is coherent here: for `parachute`, the components isolate the sky (first), the skydiver (second), and the canopy (third). Similarly, for `tench`, the decomposition distinctly separates the person holding the catch (first), the fish itself (second), and the background trees (third).

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract and introduction claim (i) a patch-token Semi-NMF formulation for ViTs that yields concept heatmaps without architectural modifications, validated in Section 3 and qualitatively in Figure 1; (ii) a Gram-matrix alignment loss that, jointly with reconstruction, upper-bounds the downstream KL divergence of any linear classifier, stated in Section 3.1 and proven in Appendix C; (iii) sparser concepts than SAEs in four of six configurations and faster extraction, supported by statistical tests in Section 4.1 and Appendix E.15; and (iv) agreement between unsupervised and supervised concept banks, quantified via RSVC in Section 4.3 and, for a single bank learned from unlabeled multi-class data, in Section 4.4. Configurations in which PLACE does not outperform the baselines are reported explicitly, and limitations on scope (VFM family, looseness of the theoretical bound, low-data COCO regime) are discussed in Section 5.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: Section 5 discusses (i) backbone coverage, (ii) the looseness of the theoretical bound in Appendix C, (iii) degraded performance on low-sample-count COCO classes, (iv) quantization error from token grouping, (v) the linear factorizability assumption and the fixed dictionary rank, (vi) the reliance of RSVC on linear predictive models, and (vii) the absence of human evaluation and of automatic textual labels for the discovered concepts.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.

- If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
- While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [Yes]

   Justification: Proposition 1 (Section 3.1) states that jointly minimizing reconstruction error and Gram-matrix deviation bounds the downstream KL divergence for any linear classifier. The full proof, including all assumptions, is provided in Appendix C, with a discussion of tightness and practical implications in Appendix C.6.

   Guidelines:

   - The answer [N/A] means that the paper does not include theoretical results.
   - All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
   - All assumptions should be clearly stated or referenced in the statement of any theorems.
   - The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
   - Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
   - Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: The paper provides complete implementation details, including datasets (ImageNet [20, 63] and COCO [45]), model architectures (ResNet [29], ViT-B/16 [23], ViT-MAE [30], and DINOv2 [56]), optimization parameters, decomposition strategies, and hyperparameter choices. These are described in the main text and Appendix D.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.

- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: Code is publicly available at `https://github.com/Restuccia-Group/PLACE`, with instructions to reproduce the main experimental results. All datasets (ImageNet [20, 63] and COCO [45]) and pretrained model checkpoints used in our experiments are publicly available. The paper and appendices additionally provide the algorithmic detail (Semi-NMF formulation, loss functions, hyperparameters in Appendix D, and optimization procedure) needed for independent reimplementation.

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: The paper specifies all experimental settings necessary to understand and reproduce the results: dataset splits (10 ImageNet [20, 63] classes and 5 COCO [45] classes, with class-conditioned concept banks extracted from correctly classified images,

and a task-agnostic setting with 200 randomly sampled unlabeled images per run and $r = 100$, Section 4.4), model architectures (ResNet [29], ViT-MAE [30], ViT-B/16 [23], DINOv2 [56]), data preprocessing (sliding-window cropping, patch-token extraction), matrix decomposition setup (rank selection, NNDSVD initialization), and the training procedure (AdamW optimizer, learning rate, early stopping). Loss hyperparameter sensitivity is reported in Section E.2; the full configuration is listed in Section D.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
- The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

Answer: [Yes]

Justification: Quantitative results are reported as mean $\pm$ standard deviation (1-sigma) over independent runs with different random seeds: five runs for the class-conditioned experiments (seeds in Appendix D) and twenty runs for the task-agnostic setting (Section 4.4). The variability captured is that of the random initialization and optimization of the factorization and, in the task-agnostic setting, of the random sampling of the images in each run. Differences between PLACE and the SAE baselines are tested with two-sided Welch's t-tests, which do not assume equal variances, and differences between reconstruction and +Gram in the task-agnostic setting with paired two-sided Wilcoxon signed-rank tests; both are Holm–Bonferroni corrected, and adjusted p-values are reported in Appendix E.15 and Table 3. Dictionary stability (W-Stab) is reported as a single cross-seed statistic without error bars.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

Answer: [Yes]

Justification: All experiments were performed on a single NVIDIA RTX 5060 Ti GPU (16 GB VRAM) with CUDA 12.8. The core computation involves activation extraction followed by low-rank Semi-NMF optimization; per-class concept extraction completes in under 10 s for the reconstruction and Gram variants (detailed breakdown in Appendix E.9). The compute requirements are modest and accessible on consumer-grade hardware.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
- The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
- The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

Answer: [Yes]

Justification: We have reviewed the NeurIPS Code of Ethics and confirm that our work adheres to its guidelines. The research involves no human subjects or personally identifiable information. Broader societal impacts and potential risks are discussed in Section G.

Guidelines:

- The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
- If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
- The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

Answer: [Yes]

Justification: Section G discusses both positive impacts (transparency, trust calibration, and safety auditing in sensitive deployment scenarios) and negative risks (targeted adversarial perturbations exploiting revealed concept structure, and the possibility that interpretable explanations may foster unwarranted trust).

Guidelines:

- The answer [N/A] means that there is no societal impact of the work performed.
- If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.

- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [N/A]

    Justification: The paper does not involve the release of pretrained generative models, language models, or scraped datasets that pose a high risk of misuse. The released code pertains to concept-based interpretability for vision foundation models and vision classifiers.

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: All datasets (ImageNet [20, 63], non-commercial research license; COCO [45], CC-BY 4.0) and pretrained models (ResNet via `timm` [29], Apache 2.0; ViT-B/16 via `timm` [23], Apache 2.0; ViT-MAE [30], CC-BY-NC 4.0; and DINOv2 [56], Apache 2.0) are cited in the main text. Baseline methods (ICE [71], CRAFT [25], FACE [7]) are cited from their original publications. All usage complies with the respective licenses and terms of use.

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
    - If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [N/A]

    Justification: The paper does not release new datasets or pretrained models.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

    Justification: The paper does not involve crowdsourcing or research with human subjects.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [N/A]

    Justification: The paper does not involve crowdsourcing or research with human subjects.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

Answer: [N/A]

Justification: The core method does not involve LLMs as any important, original, or non-standard component.

Guidelines:

- The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.