VISUALIZING DENSE NEURAL REPRESENTATIONS

Anonymous authors
Paper under double-blind review

ABSTRACT

Modern dense-feature extractors such as DINOv3 or Perception Encoder achieve strong performance on a variety of spatially fine-grained tasks. For a qualitative understanding, the features are usually mapped onto the three leading principal components which are treated as RGB values. Surprisingly, the noisiness of these visualizations varies substantially for models with similar downstream performance, suggesting a much larger gap in feature quality than indicated by quantitative evaluation. We propose to fix this issue by using an alternative visualization method based on the Laplace operator that inherits all advantages of PCA, while taking the spatial structure of the data into account. We first show that models with noisy PCA feature maps indeed contain linear components with substantially less noise, qualitatively explaining the relatively modest performance differences on dense tasks. Interestingly, the method reveals an opposite problem for the DINOv3 model: PCA visualizations overestimate the spatial smoothness of the model's representations by encoding a strong spatial bias. The proposed Laplacian PCA effectively removes this bias, revealing additional semantically meaningful structure. Quantitatively, we demonstrate that for all tested models linear components identified by Laplacian PCA align better with dimensions relevant for semantic segmentation and monocular depth estimation.

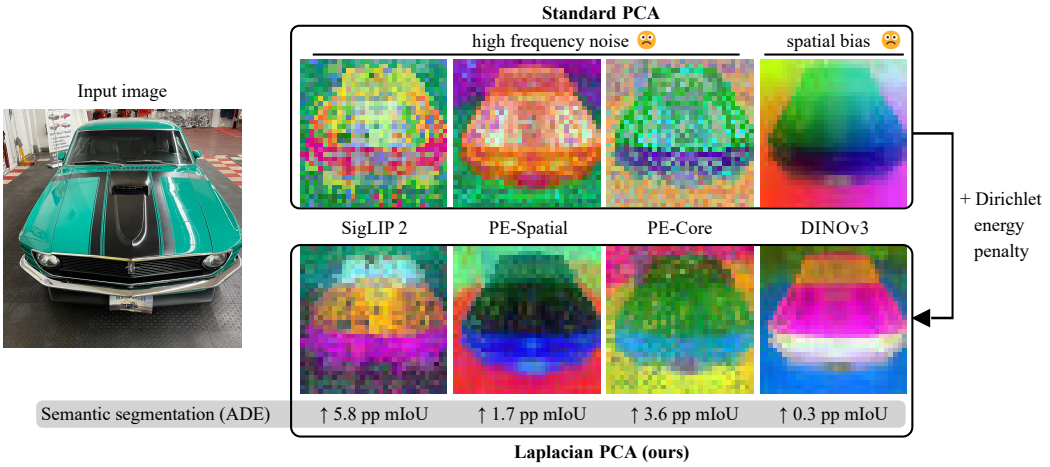


Figure 1: Common dense-feature model visualizations based on principal component analysis indicate that some models' representations are dominated by high-frequency noise (top row). Our proposed Laplacian PCA extracts linear components with higher spatial contiguity, enabling more faithful visualizations that are better aligned with task-relevant dimensions (bottom row).

1 INTRODUCTION

Multiple recent generalist vision models (Oquab et al., 2023; Tschannen et al., 2025; Bolya et al., 2025) not only extract global image representations, but also spatially dense feature maps. These can facilitate a variety of downstream tasks that rely on high spatial resolution, such as semantic segmentation (Zhou et al., 2017), depth estimation (Geiger et al., 2012) or correspondence matching (Jampani et al., 2023). Extracting such feature maps works by dividing an image into patches

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

(Dosovitskiy et al., 2021; Liu et al., 2022), and training the model to output a separate feature vector for each patch through a combination of self-supervision (Oquab et al., 2023; Siméoni et al., 2025), language-alignment (Tschannen et al., 2025), and supervision from other models (Bolya et al., 2025). These learned features can then be applied to a variety of tasks using lightweight task-specific probes (Lüddecke & Ecker, 2025) without retraining the backbone model. The simplest and surprisingly powerful method is to train an affine linear map that maps a token representation vector to a vector of task logits (Siméoni et al., 2025).

Caron et al. (2021) proposed to visualize model representations by computing the first three principal components of token activations for an image and mapping these dimensions to RGB values. We argue that the ubiquity of this approach in the literature on dense feature models in recent years reflects the fact that the quality of high-dimensional, spatially structured dense features is difficult to fully capture quantitatively. Information like which image patches group together, whether object boundaries are preserved or which semantic relationships in an image are encoded yields valuable insight into a model's performance, supplementing spatially collapsed benchmark results. One can argue that PCA visualizations have become part of a standard evaluation pipeline: They have been used to demonstrate the effectiveness of post-training interventions such as outlier removal (Wang et al., 2024; Darcet et al., 2024; Chen et al., 2025) or feature upsampling (Wimmer et al., 2026; Fu et al., 2024; Lappe & Giese, 2026), training-time interventions such as resolution scaling (Siméoni et al., 2025) or feature alignment (Bolya et al., 2025), as well as to explicitly compare the feature representations of different models in a benchmarking setting (Siméoni et al., 2025). Interestingly, PCA-based comparisons between different models (Siméoni et al., 2025; Bolya et al., 2025) have suggested large differences in the quality of features between models, especially in terms of noisiness, even though recent models are at least in the same ballpark in terms of linear task performance. We argue that this discrepancy, together with the pervasiveness of PCA-based visualization, warrants a detailed investigation of how faithfully PCA reflects the structure and quality of dense neural representations.

Summary of Contributions.

1. We show that the noisiness of their respective leading principal components increases from DINOv3 to Perception Encoder to SigLIP 2, quantifying previous qualitative findings. Utilizing these insights, we propose a simple, yet principled alternative visualization strategy that yields feature maps with higher spatial consistency without introducing additional computational cost or complexity.
2. We show that all tested models have higher-quality linear components than those identified by standard PCA, implying that PCA systematically underestimates dense-feature quality.
3. We demonstrate that our feature maps align substantially better with downstream semantic tasks, especially for otherwise very noisy models. In a perhaps surprising fashion, this holds true even for the highly convincing PCA visualizations of the DINOv3 model.

2 Background

Let $\boldsymbol{z} \in \mathbb{R}^{3\times H\times W}$ denote an image, and $f(\cdot)$ a dense-feature extractor. This means that the model first divides the image into non-overlapping patches $\boldsymbol{z}'_1, \ldots, \boldsymbol{z}'_{(h\cdot w)} \in \mathbb{R}^{3\times p\times p}$, where $p \in \mathbb{N}$ denotes the patch size. Applying a linear map to each (flattened) patch yields the initial patch token representation $\boldsymbol{Z} \in \mathbb{R}^{(h\cdot w)\times c}$, which is then processed by the Vision Transformer to obtain a latent encoding vector for each patch. It therefore constitutes a function $f : \boldsymbol{z} \mapsto \boldsymbol{X}$, where $\boldsymbol{X} \in \mathbb{R}^{(h\cdot w)\times c}$ is a representation in which each image patch token is assigned its own feature vector $\boldsymbol{x}_i \in \mathbb{R}^c$; we assume in the following that $\boldsymbol{X}$ is normalized to have zero column mean.

The goal of these models is to extract general information that can be decoded for a variety of downstream tasks that require high spatial resolution. Ideally, this information can be read out by a simple linear probe, allowing for inexpensive training and removing the need for finetuning or task-specialized architectures. The behavior of such linear probes on a specific image can be captured by the linear kernel Gram matrix

$$\boldsymbol{K} := \boldsymbol{X}\boldsymbol{X}^\intercal, \tag{1}$$

making it an object of particular interest. Indeed, previous work (Siméoni et al., 2025) explicitly regularizes $\boldsymbol{K}$ to ensure good performance on dense tasks. Given the important role of $\boldsymbol{K}$, using principal component analysis to visualize the learned features of such models is quite intuitive, as PCA aims to capture the linear variation of the data encoded in the Gram matrix. Main directions of variance between tokens are given exactly by the leading eigenvectors of $\boldsymbol{K}$. Mapping the first three eigenvectors to RGB values then yields a simple visual summary of the learned features (Caron et al., 2021). However, feature spaces are usually fairly high-dimensional, making it unclear how well three-dimensional plots can represent them. PCA captures those components that have high variance in the data, but these need not necessarily be the ones that are important for downstream performance. In particular, feature visualizations of some models are extremely noisy (see, for example, Siméoni et al. (2025) Fig. 13), meaning that the directions of highest variance are not properly aligned with semantic or spatial concepts. Despite this apparent noisiness, these models still have good linear performance on dense semantic tasks, indicating that *some* linear components *are* aligned better with scene semantics.

3 LAPLACIAN PCA

3.1 SPATIALLY CONTIGUOUS FEATURE MAPS

To make an educated guess as to how such components might be identified, note that scene semantics are usually piecewise constant in space, and depth and surface normals also mostly vary gradually. This implies the heuristic conclusion that components that are useful for these tasks should be those that have little local variation over image pixels. Note that this idea is not new: Bolya et al. (2025) also argued that good visualizations of patch token features should be steered towards being spatially contiguous. To this end, the authors suggested smoothing feature maps before computing principal components to obtain visualizations that have lower spatial frequency. While successfully removing noise, this approach has a substantial drawback: Downstream task probes typically operate tokenwise, so at each spatial location the information available to the probe is that represented in the corresponding token. Therefore, applying a low-pass filter to the feature maps creates a visualization of information that downstream task probes will not have access to. Further, spatially smoothing the feature maps before visualization will underestimate the model's spatial precision. We argue here that visualizations of dense feature maps should satisfy two important criteria: First, low-dimensional representations should be linear with respect to high-dimensional features to reflect the common use of linear downstream task probes. Second, the low-dimensional representation $\boldsymbol{y}_i$ of token i should depend only on the high-dimensional token representation $\boldsymbol{x}_i$. Together, these criteria restrict the visualization problem to finding an image-specific matrix $\boldsymbol{W}$ such that $\boldsymbol{Y} = \boldsymbol{X}\boldsymbol{W}$. The remainder of the text aims to find and then evaluate a specific choice of $\boldsymbol{W}$.

3.2 QUANTIFYING THE NOISINESS OF FEATURE MAPS

In the current literature, PCA visualizations of dense feature models have only ever been treated in a qualitative fashion, with example images showing that some models have noisier principal components than others. While visualizations are inherently a qualitative concept, a quantitative analysis is necessary to study what they encode and how these encodings relate to quantitative downstream tasks. Therefore, we first assess the noisiness of PCA visualizations of different models in terms of their respective expected Dirichlet energy. For a sufficiently smooth scalar function $h : \Omega \to \mathbb{R}$, the Dirichlet energy is defined as

$$E(h) := \frac{1}{2} \int_{\Omega} ||\nabla h(x)||_2^2 \mathrm{dx}, \tag{2}$$

and measures how much h varies locally. If $E(h)$ is large, h fluctuates rapidly, whereas a low E implies a smooth function. By considering each eigenvector $\boldsymbol{v} \in \mathbb{R}^{h \cdot w}$ of $\boldsymbol{K}$ a scalar field over the lattice of patch tokens Ω, we can quantify local variability using the discretized Dirichlet energy

$$E(\boldsymbol{v}) = \sum_{\{i,j\} \in \mathcal{E}} ||\boldsymbol{v}_i - \boldsymbol{v}_j||_2^2 = \boldsymbol{v}^\intercal \boldsymbol{L} \boldsymbol{v}, \tag{3}$$

where $\mathcal{E}$ denotes the edges between neighboring tokens on the lattice, and $\boldsymbol{L}$ denotes the associated graph Laplacian. Here, we consider tokens to be neighbors if and only if they are horizontal or vertical neighbors on the lattice, resulting in four neighbors for a token that does not sit on the boundary.

Equation (3) is then satisfied by the standard combinatorial Laplacian described in Section B.1. Figure 2 displays the average Dirichlet energy over images of the first three principal components for different models. Namely, we experiment with DINOv3 (Siméoni et al., 2025), SigLIP 2 (Tschannen et al., 2025), and the 'Spatial' and 'Core' variants of the Perception Encoder (Bolya et al., 2025). The qualitative findings from Figure 1 and prior work are clearly captured: DINOv3 has low energy, corresponding to spatially contiguous visualizations, SigLIP 2 has high energy, corresponding to noisy visualizations, and Perception Encoder models fall in between; among the latter two, the 'Spatial' variant has lower energy as expected from the training procedure (see Bolya et al. (2025)).

3.3 Taming the Dirichlet energy

Having quantified the existence of high-variance noisy feature directions, we now study whether low-energy components can be extracted from all models. Conveniently, the Dirichlet energy lends itself very well to regularizing principal components: Since for a discrete scalar field $\boldsymbol{u}$ on the lattice Ω we have $E(\boldsymbol{u}) = \boldsymbol{u}^\intercal \boldsymbol{L}\boldsymbol{u}$, and $\boldsymbol{L}$ is symmetric and positive semi-definite, we can decompose the space of such scalar fields into the discrete, orthonormal eigenfunctions $\boldsymbol{u}_1, \ldots, \boldsymbol{u}_n$ of $\boldsymbol{L}$, namely its eigenvectors, satisfying $E(\boldsymbol{u}_i) = \lambda_i$, where λ_i denotes the i-th eigenvalue. In particular, $\operatorname{argmax}_{||\boldsymbol{u}||_2=1} \boldsymbol{u}^\intercal \boldsymbol{L}\boldsymbol{u}$ is given by the leading eigenvector of $\boldsymbol{L}$. Conversely, the maximally smooth component is given by the eigenvector with lowest eigenvalue, or equivalently the leading eigenvector of $-\boldsymbol{L}$. A straightforward way of finding spatially contiguous components is then to make use of the fact that both the variance objective and the smoothness objective can be cast as an eigenvalue problem. In particular, we can find

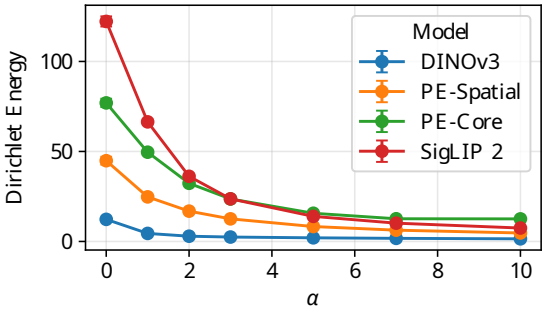


Figure 2: The Dirichlet energy of the first three principal components (standard PCA corresponds to $\alpha = 0$) captures the varying noise levels between models. The Dirichlet energy decreases with increasing regularization.

$$\boldsymbol{v}^* = \underset{||\boldsymbol{v}||_2=1}{\operatorname{argmax}} \; \boldsymbol{v}^\intercal \boldsymbol{K}\boldsymbol{v} - \alpha \boldsymbol{v}^\intercal \boldsymbol{L}\boldsymbol{v}, \tag{4}$$

which balances a variance maximization term $\boldsymbol{v}^\intercal \boldsymbol{K}\boldsymbol{v}$ with an energy minimization term $-\boldsymbol{v}^\intercal \boldsymbol{L}\boldsymbol{v}$. As $\boldsymbol{K} - \alpha\boldsymbol{L}$ is symmetric, orthogonal vectors that maximize Equation (4) can then be found easily by an eigendecomposition of $\boldsymbol{K} - \alpha\boldsymbol{L}$. This is exactly the formulation derived by Jiang et al. (2013), which, however, has a drawback for our application: While for regular PCA $\boldsymbol{v}^*$ always lies in the column space of X and can therefore be written as a linear map of the features, this is not generally true for the solution of Equation (4). Therefore, to ensure that the low-dimensional representations can in fact be represented as a linear map of the high-dimensional features, we follow the standard sample-space-to-feature-space formulation of PCA (Schölkopf & Smola, 2001) and solve

$$\boldsymbol{w}^* = \underset{||\boldsymbol{w}||_2=1}{\operatorname{argmax}} \; \boldsymbol{w}^\intercal \boldsymbol{X}^\intercal \boldsymbol{X}\boldsymbol{w} - \alpha \boldsymbol{w}^\intercal \boldsymbol{X}^\intercal \boldsymbol{L}\boldsymbol{X}\boldsymbol{w}, \tag{5}$$

by finding the first three eigenvectors of $\boldsymbol{X}^\intercal \boldsymbol{X} - \alpha \boldsymbol{X}^\intercal \boldsymbol{L}\boldsymbol{X}$, say $\boldsymbol{W} \in \mathbb{R}^{c\times 3}$, and computing the low-dimensional representation as $\boldsymbol{Y} = \boldsymbol{X}\boldsymbol{W}$. Finally, to make the regularization level α interpretable across models, we scale the matrices $\boldsymbol{X}^\intercal \boldsymbol{X}$ and $\boldsymbol{X}^\intercal \boldsymbol{L}\boldsymbol{X}$ so that the largest respective eigenvalues of both individual matrices are equal (Jiang et al., 2013).

4 Experiments

4.1 Laplacian PCA finds smoother components for noisy models

We first observe that increasing Dirichlet regularization yields a decrease in the Dirichlet energy of the three leading principal components for all models (Figure 2). This demonstrates that all models contain low-energy components; they differ only in how much variance these components explain.

Figure 3 displays the visualizations based on these components. As is common practice, we gather the patch token representation $\boldsymbol{X}$ of an image, project it onto the identified components to obtain $\boldsymbol{XW} \in \mathbb{R}^{(h \cdot w) \times 3}$, normalize to the $[0, 1]$ range and map each component to one of the color components in the RGB image representation. We observe that Laplacian PCA components of otherwise noisy models are substantially more spatially contiguous, revealing structure that better aligns with semantic concepts visible in the image. This finding does not extend to the DINOv3 model, where even weak regularization appears to yield oversmoothed visualizations.

4.2 Laplacian principal components are better aligned with downstream tasks

Given the previous findings, the question remains whether these regularized visualizations actually better reflect a model's behavior on downstream tasks. We hypothesized that PCA visualizations capture directions of high variance, but task-relevant low-variance dimensions could easily be discovered by a linear probe. Are these task-relevant dimensions picked up by Laplacian PCA?

To test this hypothesis, we trained linear probes on downstream tasks. For each task, this corresponds to learning the weights of a matrix $\boldsymbol{M}$, such that for an image representation $\boldsymbol{X}$, task logits are given by $\boldsymbol{XM}$. During evaluation, we tested how well a set of principal components $\boldsymbol{W} \in \mathbb{R}^{c \times 3}$ facilitates the task by computing the logits as $\boldsymbol{XWW}^\intercal\boldsymbol{M}$. In other words, we mapped features onto principal component scores and then back up to the original space. This procedure measures directly how well-suited the dimensions are for solving the task at hand. The weight matrix $\boldsymbol{M}$ was trained globally on the full feature representation without any knowledge of the dimensionality reduction. During evaluation, principal components were computed separately for each image just as for visualization purposes.

Table 1: Semantic segmentation performance on PASCAL VOC and ADE20K at $\alpha = 1$. Values are obtained by training a linear probe on the full feature space of each model, and evaluating on the basis spanned by the three leading principal components. 'PCA' refers to standard PCA (leftmost column in Figure 3). 'Laplacian PCA' refers to Laplacian PCA with $\alpha = 1$. Results align with qualitative findings from Figure 3: For the noisy models, Laplacian PCA maps at the $\alpha = 1$ level have better spatial structure than the baseline, leading to improved semantic performance. For DINOv3, which already has smooth feature maps using standard PCA, this regularization is too strong and yields blurry feature maps with decreased semantic accuracy.

	PASCAL VOC				ADE20K			
	mIoU ↑		PixAcc ↑		mIoU ↑		PixAcc ↑	
Model	PCA	Laplacian PCA	PCA	Laplacian PCA	PCA	Laplacian PCA	PCA	Laplacian PCA
SigLIP 2	52.6	**59.4**	86.1	**88.7**	23.3	**29.1**	57.1	**64.6**
PE-Spatial	63.0	**63.3**	91.5	**91.8**	25.5	**27.2**	68.4	**70.6**
PE-Core	54.9	**58.2**	88.1	**89.4**	21.8	**25.4**	60.9	**66.2**
DINOv3	**69.1**	59.3	**92.7**	89.1	**33.6**	25.2	**73.3**	68.9

Semantic Segmentation. To test how well components align with scene semantics, we consider the task of predicting a semantic class for each image pixel. We perform experiments as described above on ADE20k (Zhou et al., 2017) and PASCAL VOC (Chen et al., 2014), reporting the mean Intersection over Union (mIoU) and Pixel Accuracy (PixAcc) in Table 1. For now, we fix $\alpha = 1$ for all models for simplicity. We observe that Laplacian components have a substantial performance advantage over regular principal components for the noisy models, demonstrating that the perceived improvement in feature map quality from Figure 3 translates to better semantic alignment. Interestingly, as one might also suspect from Figure 3, this is not true for the DINOv3 model: Unregularized principal components are already spatially smooth, and adding a penalty at the $\alpha = 1$ level yields qualitatively blurry feature maps, which also translate to a drop in semantic segmentation performance. Further, Figure 4 displays mIoU as a function of the regularization strength α. For the noisy models, performance increases monotonically, before decreasing monotonically once regularization becomes too harsh. Therefore, there is a positive sweet spot at which visualization dimensions align best with scene semantics. This effect occurs in both datasets and we further validate its robustness

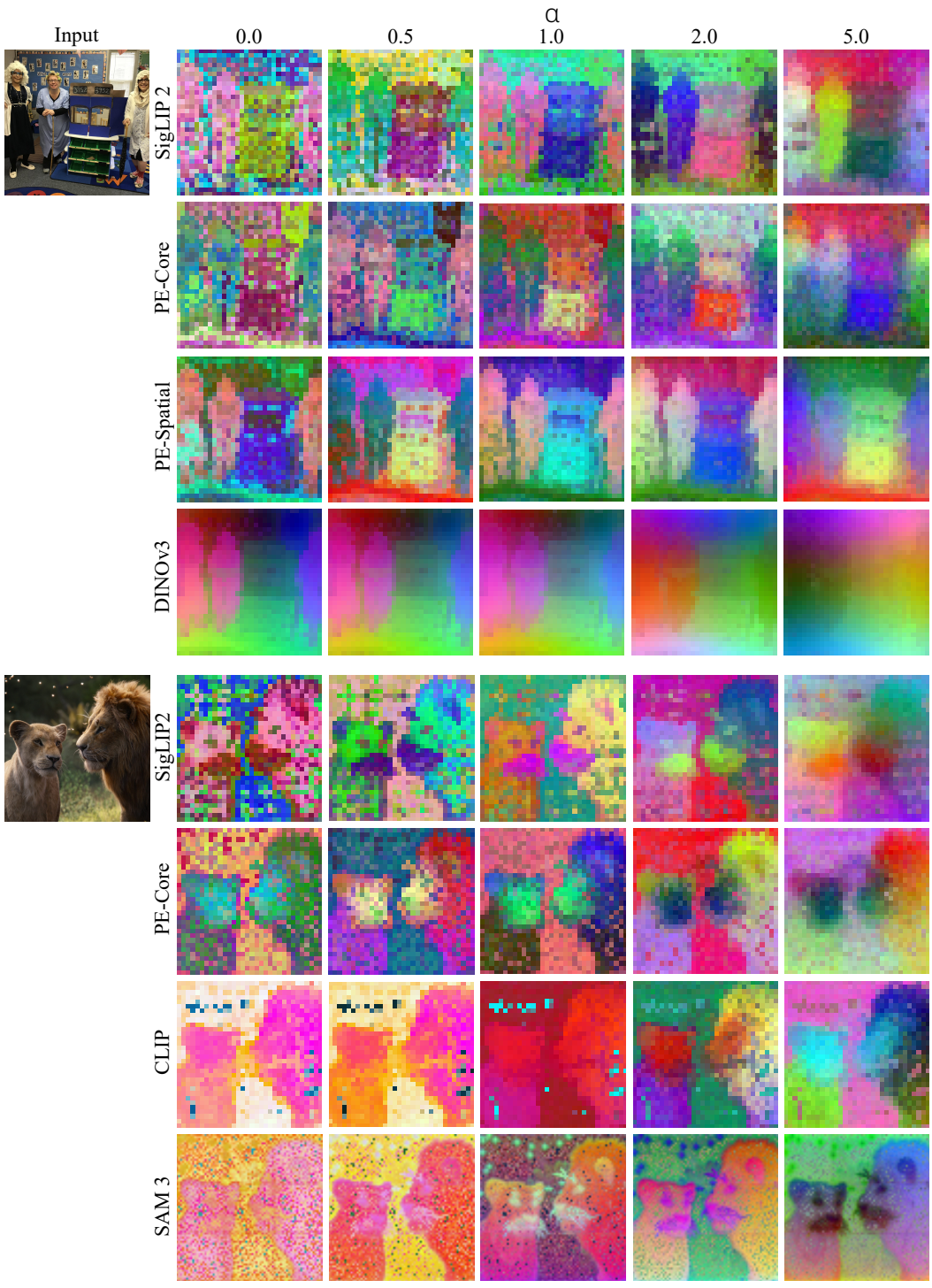


Figure 3: Visualizations generated with Laplacian principal component analysis. The regularization hyperparameter increases from left to right, and $\alpha = 0$ reduces to regular PCA. SigLIP 2 and the Perception Encoder models exhibit noisy feature maps without regularization. The same is true for CLIP (Radford et al., 2021; Ilharco et al., 2021) and the Segment-Anything 3 model (SAM 3) (Carion et al., 2026). For all these models, incorporating spatial regularization reveals low-dimensional structure with substantially lower noise levels, and qualitatively better semantic alignment. Importantly, representations are not spatially pooled in any form. Each low-dimensional patch token representation is computed as a linear map of the high-dimensional representation of only that token.

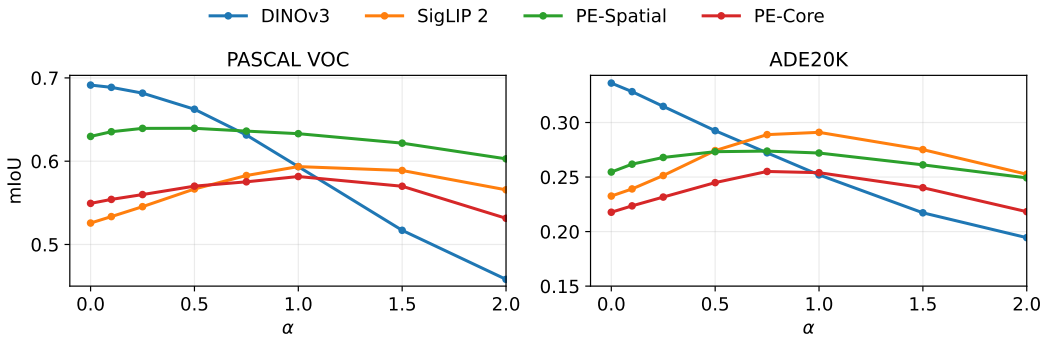


Figure 4: Semantic segmentation performance as a function of regularization strength. For the models with noisy feature visualizations, performance increases up to a maximum, and then drops again once components are regularized too harshly. For DINOv3, performance drops monotonically with increasing regularization.

Table 2: Depth estimation performance (RMSE) on NYU Depth V2. Lower is better. The experimental setup mirrors the one in Table 1. We report results for $\alpha = 1$, as well as for the best value for each individual model.

Model ↓	PCA	Laplacian PCA ($\alpha = 1$)	Laplacian PCA (Best α)
SigLIP 2	0.971	0.937	**0.933**
PE-Spatial	0.808	0.769	**0.758**
PE-Core	0.992	0.993	**0.980**
DINOv3	0.870	0.869	**0.867**

on random subsets in the appendix (Figure 7). There, we also show that the same effect occurs for CLIP and SAM 3 models. As observable in Figure 3, the quantitative sweet spot corresponds well to the visually most convincing feature maps. For DINOv3, which has smooth visualizations even without regularization, alignment with semantics drops monotonically and rapidly with increasing α. This phenomenon has some interesting implications, warranting a separate discussion that we shall present later on.

Depth Estimation. We performed the same experiments for monocular depth estimation on NYUv2 (Silberman et al., 2012) and report the root mean squared error (RMSE) in Table 2. Alignment gains as measured by task performance are more modest compared to semantic segmentation, likely because depth information is not encoded particularly strongly in either regular or Laplacian PCA. That being said, Laplacian PCA outperforms regular PCA on all models, and fixing $\alpha = 1$ at least does not yield a drop in performance for any model.

Baseline DINOv3 visualizations are too smooth. We have observed that penalizing the Dirichlet energy leads to better visualizations of models that otherwise exhibit noise in their feature maps. However, the already highly contiguous PCA visualizations of DINOv3 features deteriorate quickly both qualitatively and quantitatively when introducing regularization. Since Laplacian PCA naturally reduces to regular PCA by setting $\alpha = 0$, we first argue that this finding does not violate the proposed framework for visualization. Further, noting that the semantic alignment of all other models follows an inverse U-shape in Figure 4, we hypothesize that this should also be true for DINOv3. As the formulation in Equation (4) does not assume that α be positive, we can also compute components with negative regularization, i.e. an optimization problem that actually encourages a *higher* Dirichlet energy. Figure 5 a. shows that the alignment of DINOv3 components with image semantics indeed follows an inverse U-shape, with the optimal value being negative. Therefore, we argue that standard PCA actually identifies components that are *too* smooth in this case. Laplacian PCA with negative regularization reliably removes a clearly visible, strong spatial bias from low-dimensional representations, yielding much stronger separation of semantic concepts (Figure 6).

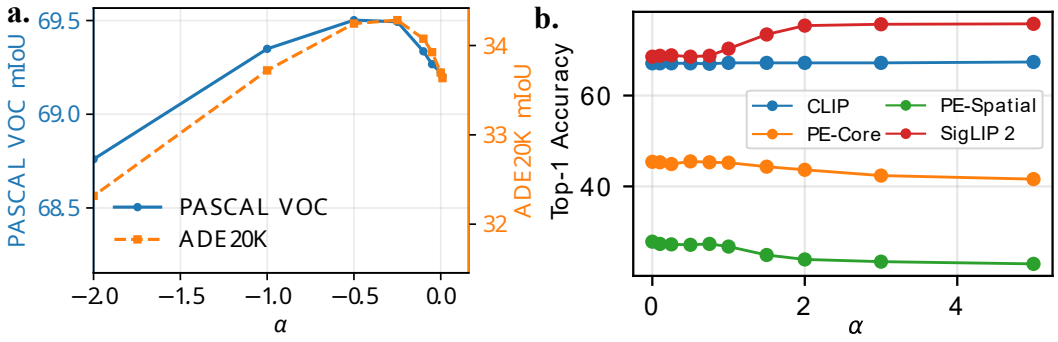


Figure 5: a.) PCA visualizations of DINOv3 features are actually *too* smooth and benefit from regularization towards *higher* Dirichlet energy. b) Components of noisy models have stable global classification performance in the α range that is optimal for dense tasks, showing that Laplacian PCA does not discard globally relevant dimensions.

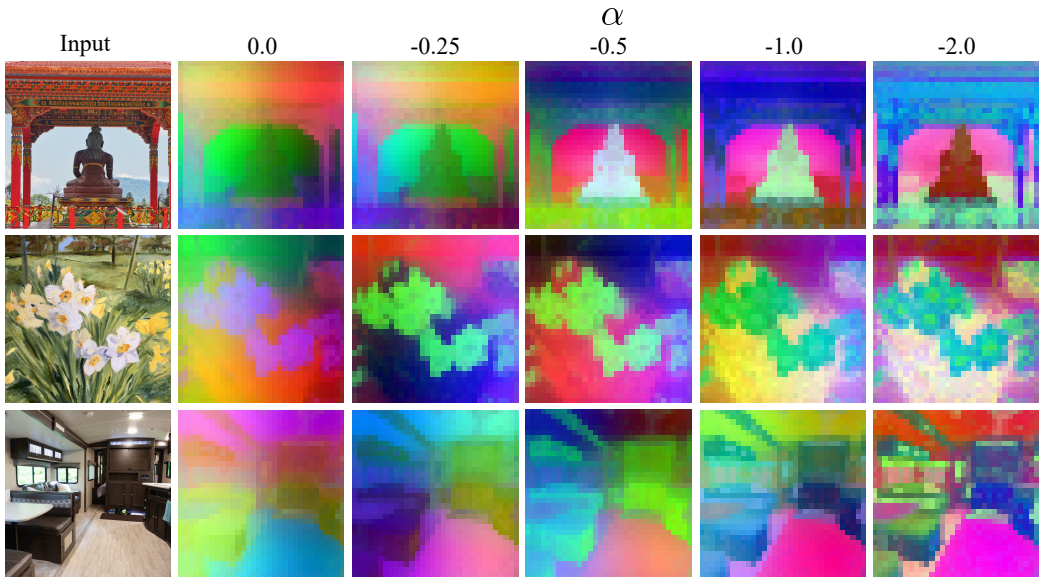


Figure 6: Visualizations of DINOv3 dense features when applying negative regularization, such that components with low Dirichlet energy are penalized. Unregularized PCA components exhibit strong spatial bias and vary slowly over space, even across spatial boundaries of semantic concepts. Note that this is not an effect of blurring as PCA operates tokenwise; tokens clearly have strong spatial encoding in their directions of maximal variance. Laplacian PCA reliably removes this spatial bias, yielding much stronger separation of semantic concepts.

Regularized components retain global semantic information. Motivated by prior work on 'outlier tokens' (Darcet et al., 2024) in ViTs, we wondered whether locally noisy dimensions are highly relevant for global tasks. Under this hypothesis, good alignment of lower-energy components with local semantics should come with a trade-off of poor alignment with globally relevant dimensions. To test this, we ran an experiment analogous to the ones described in Section 4.2 for global image classification on the CUB dataset (Wah et al., 2011) (see Section B.2.3). Briefly, we do not find such a trade-off; choosing α values that are optimal for local semantics leaves global semantic performance unchanged (Figure 5 b.). Increasing regularization beyond this point hurts alignment with globally relevant dimensions in the PE models but improves it substantially for SigLIP 2, surprisingly indicating that the latter stores global information in very low-energy components. Note that the high-energy noisy dimensions we consider here are conceptually different from the high-norm outlier tokens treated in previous works, and we do not claim that this contradicts previous findings (Darcet et al., 2024; Lappe & Giese, 2025).

5 DISCUSSION

Choosing α. For exploratory analysis, plotting a range of both positive and negative values is likely to yield the most informative results. However, when benchmarking models, a principled selection is desirable. In this case, we propose to choose the values that respectively optimize task performance for each model as described in Section 4.2. As task probes have to be trained anyway for benchmarking, optimal visualization values can be found from a single pass over the test set and therefore add little computational demand. As shown in Figure 4 and Figure 7, the optimal value is robust against different choices of semantic classes and datasets.

Related Work. The Laplacian PCA relates to a body of work that uses graph structure for dimensionality reduction. Locality Preserving Projections (He & Niyogi, 2003) construct a discrete Laplacian matrix that aims to approximate the Laplace-Beltrami operator of the data manifold. They solve an eigendecomposition of $\boldsymbol{X}^\intercal \boldsymbol{L} \boldsymbol{X}$ (in our notation) to find scalar maps that vary smoothly over the manifold and therefore assign similar values to samples that are close on the manifold. A similar strategy is proposed in Belkin & Niyogi (2003). In contrast, we do not construct $\boldsymbol{L}$ to capture similarities in the data structure, but only to regularize on the lattice of patch tokens that we project to. While these methods optimize spectral properties of the low-dimensional graph to reduce dimensionality, in our Laplacian formulation the primary objective for structure preservation is still data variance, just like in regular PCA. More similar to the formulation presented here is that of Jiang et al. (2013), which computes an eigendecomposition of $\boldsymbol{K} - \alpha \boldsymbol{L}$. As for the two aforementioned methods, the Laplacian is still computed from sample similarity rather than fixed to a specific target space. Further, different from ours, this approach does not enforce linearity of the principal components, as discussed in Section 3.3.

Placed in the context of the literature on dense-feature extractors, our results demonstrate clearly why models with noisy feature visualizations, primarily SigLIP 2 and Perception Encoder, still achieve good performance on downstream tasks: The directions of maximal variance are noisy, but high-variance dimensions with less noise exist and can be visualized, and align much better with those chosen by downstream task probes. Therefore, standard PCA underestimates the feature quality of these models. For DINOv3, we demonstrated an interesting reversal of this effect: PCA underestimates feature quality by overemphasizing a strong spatial bias. This spatial bias has also been noted by Cuttano et al. (2026), who successfully de-bias the feature space by estimating the pathological dimensions on images with little semantic structure. The fact that undesirable model biases encoded in high-variance components can be effectively removed using simple strategies further strengthens the argument for a visualization method that ignores such biases for accurate qualitative model evaluation.

Limitations. Like any dimensionality reduction method, the proposed approach is unable to fully preserve the structure of high-dimensional embeddings, which must be taken into account when using it as a qualitative evaluation method. A more specific limitation pertains to the linearity: Lightweight non-linear task probes are also popular (Lüddecke et al., 2026), and a linear projection may fail to expose structure that is accessible to such probes. Extending a visualization method to faithfully reflect the decoding capacity of nonlinear probes is nontrivial. Nonparametric dimensionality-reduction methods such as t-SNE (Van der Maaten & Hinton, 2008) or UMAP (McInnes et al., 2020) provide one possible alternative, but assessing the complexity of the probe that would be necessary to realize nonparametric embeddings is difficult.

Conclusion. While PCA is a natural choice for visualizing features extracted by dense models, we have shown that it extracts suboptimal linear components for all currently popular model families both qualitatively and in terms of semantic task performance. We have proposed a simple Laplace penalty for PCA, resulting in a method that extracts more faithful components and is easy to implement and compute. Beyond demonstrating that features of current dense models are better than they appear, we hope to aid future efforts regarding dense-feature extractors and related applications.

AI USE STATEMENT

In this work, we used generative AI tools for refining hypotheses, providing feedback on research methodology or experiments and to implement methods. We did not use generative AI tools to

develop the conceptual framework, formulate mathematical claims, assist with translation, support qualitative and thematic data analysis and interpret results, or to write the paper. Research tasks relating to automatic proofs are not applicable to this work. We used generative AI for literature research and coding, both of which were carefully reviewed. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

Reproducibility statement

The detailed construction of the graph Laplacian, as well as links to all models used in the analysis are provided in the appendix.

References

Mikhail Belkin and Partha Niyogi. Laplacian Eigenmaps for Dimensionality Reduction and Data Representation. *Neural Computation*, 15(6):1373–1396, June 2003. ISSN 0899-7667. doi: 10.1162/089976603321780317.

Daniel Bolya, Po-Yao Huang, Peize Sun, Jang Hyun Cho, Andrea Madotto, Chen Wei, Tengyu Ma, Jiale Zhi, Jathushan Rajasegaran, Hanoona Rasheed, Junke Wang, Marco Monteiro, Hu Xu, Shiyu Dong, Nikhila Ravi, Daniel Li, Piotr Dollár, and Christoph Feichtenhofer. Perception Encoder: The best visual embeddings are not at the output of the network. In *NeurIPS*. arXiv, April 2025. doi: 10.48550/arXiv.2504.13181.

Nicolas Carion, Laura Gustafson, Yuan-Ting Hu, Shoubhik Debnath, Ronghang Hu, Didac Suris, Chaitanya Ryali, Kalyan Vasudev Alwala, Haitham Khedr, Andrew Huang, Jie Lei, Tengyu Ma, Baishan Guo, Arpit Kalla, Markus Marks, Joseph Greer, Meng Wang, Peize Sun, Roman Rädle, Triantafyllos Afouras, Effrosyni Mavroudi, Katherine Xu, Tsung-Han Wu, Yu Zhou, Liliane Momeni, Rishi Hazra, Shuangrui Ding, Sagar Vaze, Francois Porcher, Feng Li, Siyuan Li, Aishwarya Kamath, Ho Kei Cheng, Piotr Dollár, Nikhila Ravi, Kate Saenko, Pengchuan Zhang, and Christoph Feichtenhofer. SAM 3: Segment Anything with Concepts, March 2026.

Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jegou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging Properties in Self-Supervised Vision Transformers. In *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 9630–9640, October 2021. doi: 10.1109/ICCV48922.2021.00951.

Xianjie Chen, Roozbeh Mottaghi, Xiaobai Liu, Sanja Fidler, Raquel Urtasun, and Alan Yuille. Detect What You Can: Detecting and Representing Objects Using Holistic Models and Body Parts. *2014 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1979–1986, June 2014. doi: 10.1109/CVPR.2014.254.

Yinjie Chen, Zipeng Yan, Chong Zhou, Bo Dai, and Andrew F. Luo. Vision Transformers with Self-Distilled Registers, May 2025.

Claudia Cuttano, Gabriele Trivigno, Christoph Reich, Daniel Cremers, Carlo Masone, and Stefan Roth. INSID3: Training-Free In-Context Segmentation with DINOv3, March 2026.

Timothée Darcet, Maxime Oquab, Julien Mairal, and Piotr Bojanowski. Vision Transformers Need Registers, April 2024.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale, June 2021.

Stephanie Fu, Mark Hamilton, Laura Brandt, Axel Feldman, Zhoutong Zhang, and William T. Freeman. FeatUp: A Model-Agnostic Framework for Features at Any Resolution, April 2024.

Andreas Geiger, Philip Lenz, and Raquel Urtasun. Are we ready for autonomous driving? The KITTI vision benchmark suite. In *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3354–3361, June 2012. doi: 10.1109/CVPR.2012.6248074.

Xiaofei He and Partha Niyogi. Locality Preserving Projections. In *Advances in Neural Information Processing Systems*, volume 16. MIT Press, 2003.

Gabriel Ilharco, Mitchell Wortsman, Nicholas Carlini, Rohan Taori, Achal Dave, Vaishaal Shankar, Hongseok Namkoong, John Miller, Hannaneh Hajishirzi, Ali Farhadi, and Ludwig Schmidt. OpenCLIP. Zenodo, July 2021.

Varun Jampani, Kevis-Kokitsi Maninis, Andreas Engelhardt, Arjun Karpur, Karen Truong, Kyle Sargent, Stefan Popov, Andre Araujo, Ricardo Martin-Brualla, Kaushal Patel, Daniel Vlasic, Vittorio Ferrari, Ameesh Makadia, Ce Liu, Yuanzhen Li, and Howard Zhou. NAVI: Category-agnostic image collections with high-quality 3D shape and pose annotations. In *NeurIPS*, 2023.

Bo Jiang, Chris Ding, Bio Luo, and Jin Tang. Graph-Laplacian PCA: Closed-Form Solution and Robustness. In *2013 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3492–3498, June 2013. doi: 10.1109/CVPR.2013.448.

Alexander Lappe and Martin A. Giese. Register and [CLS] tokens induce a decoupling of local and global features in large ViTs. In *NeurIPS*, September 2025.

Alexander Lappe and Martin A. Giese. Another BRIXEL in the Wall: Towards Cheaper Dense Features. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7605–7614, 2026.

Zhuang Liu, Hanzi Mao, Chao-Yuan Wu, Christoph Feichtenhofer, Trevor Darrell, and Saining Xie. A ConvNet for the 2020s, March 2022.

Timo Lüddecke and Alexander S. Ecker. Characterizing Vision Backbones for Dense Prediction with Dense Attentive Probing. *Transactions on Machine Learning Research*, April 2025. ISSN 2835-8856.

Timo Lüddecke, Jan Frederik Meier, Jan van Delden, and Alexander Ecker. LiDeRe: A Lightweight Readout for Fast and Data-Efficient Dense Prediction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 2959–2971, 2026.

Leland McInnes, John Healy, and James Melville. UMAP: Uniform Manifold Approximation and Projection for Dimension Reduction, September 2020.

Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy V. Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, Mido Assran, Nicolas Ballas, Wojciech Galuba, Russell Howes, Po-Yao Huang, Shang-Wen Li, Ishan Misra, Michael Rabbat, Vasu Sharma, Gabriel Synnaeve, Hu Xu, Herve Jegou, Julien Mairal, Patrick Labatut, Armand Joulin, and Piotr Bojanowski. DINOv2: Learning Robust Visual Features without Supervision. *Transactions on Machine Learning Research*, July 2023. ISSN 2835-8856.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning Transferable Visual Models From Natural Language Supervision. In *Proceedings of the 38th International Conference on Machine Learning*, pp. 8748–8763. PMLR, July 2021.

Bernhard Schölkopf and Alexander J. Smola. *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond*. The MIT Press, December 2001. ISBN 978-0-262-25693-3. doi: 10.7551/mitpress/4175.001.0001.

Nathan Silberman, Derek Hoiem, Pushmeet Kohli, and Rob Fergus. Indoor Segmentation and Support Inference from RGBD Images. In David Hutchison, Takeo Kanade, Josef Kittler, Jon M. Kleinberg, Friedemann Mattern, John C. Mitchell, Moni Naor, Oscar Nierstrasz, C. Pandu Rangan, Bernhard Steffen, Madhu Sudan, Demetri Terzopoulos, Doug Tygar, Moshe Y. Vardi, Gerhard Weikum, Andrew Fitzgibbon, Svetlana Lazebnik, Pietro Perona, Yoichi Sato, and Cordelia Schmid (eds.), *Computer Vision – ECCV 2012*, volume 7576, pp. 746–760, Berlin, Heidelberg, 2012. Springer Berlin Heidelberg. ISBN 978-3-642-33714-7 978-3-642-33715-4. doi: 10.1007/978-3-642-33715-4_54.

Oriane Siméoni, Huy V. Vo, Maximilian Seitzer, Federico Baldassarre, Maxime Oquab, Cijo Jose, Vasil Khalidov, Marc Szafraniec, Seungeun Yi, Michaël Ramamonjisoa, Francisco Massa, Daniel Haziza, Luca Wehrstedt, Jianyuan Wang, Timothée Darcet, Théo Moutakanni, Leonel Sentana, Claire Roberts, Andrea Vedaldi, Jamie Tolan, John Brandt, Camille Couprie, Julien Mairal, Hervé Jégou, Patrick Labatut, and Piotr Bojanowski. DINOv3, August 2025.

Michael Tschannen, Alexey Gritsenko, Xiao Wang, Muhammad Ferjad Naeem, Ibrahim Alabdulmohsin, Nikhil Parthasarathy, Talfan Evans, Lucas Beyer, Ye Xia, Basil Mustafa, Olivier Hénaff, Jeremiah Harmsen, Andreas Steiner, and Xiaohua Zhai. SigLIP 2: Multilingual Vision-Language Encoders with Improved Semantic Understanding, Localization, and Dense Features, February 2025.

Laurens Van der Maaten and Geoffrey Hinton. Visualizing data using t-SNE. *Journal of machine learning research*, 9(11), 2008.

Catherine Wah, Steve Branson, Peter Welinder, Pietro Perona, and Serge Belongie. The Caltech-UCSD Birds-200-2011 Dataset. November 2011.

Haoqi Wang, Tong Zhang, and Mathieu Salzmann. SINDER: Repairing the Singular Defects of DINOv2, July 2024.

Thomas Wimmer, Prune Truong, Marie-Julie Rakotosaona, Michael Oechsle, Federico Tombari, Bernt Schiele, and Jan Eric Lenssen. AnyUp: Universal Feature Upsampling, February 2026.

Bolei Zhou, Hang Zhao, Xavier Puig, Sanja Fidler, Adela Barriuso, and Antonio Torralba. Scene Parsing through ADE20K Dataset. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5122–5130, July 2017. doi: 10.1109/CVPR.2017.544.

A Additional Results

A.1 Robustness of α

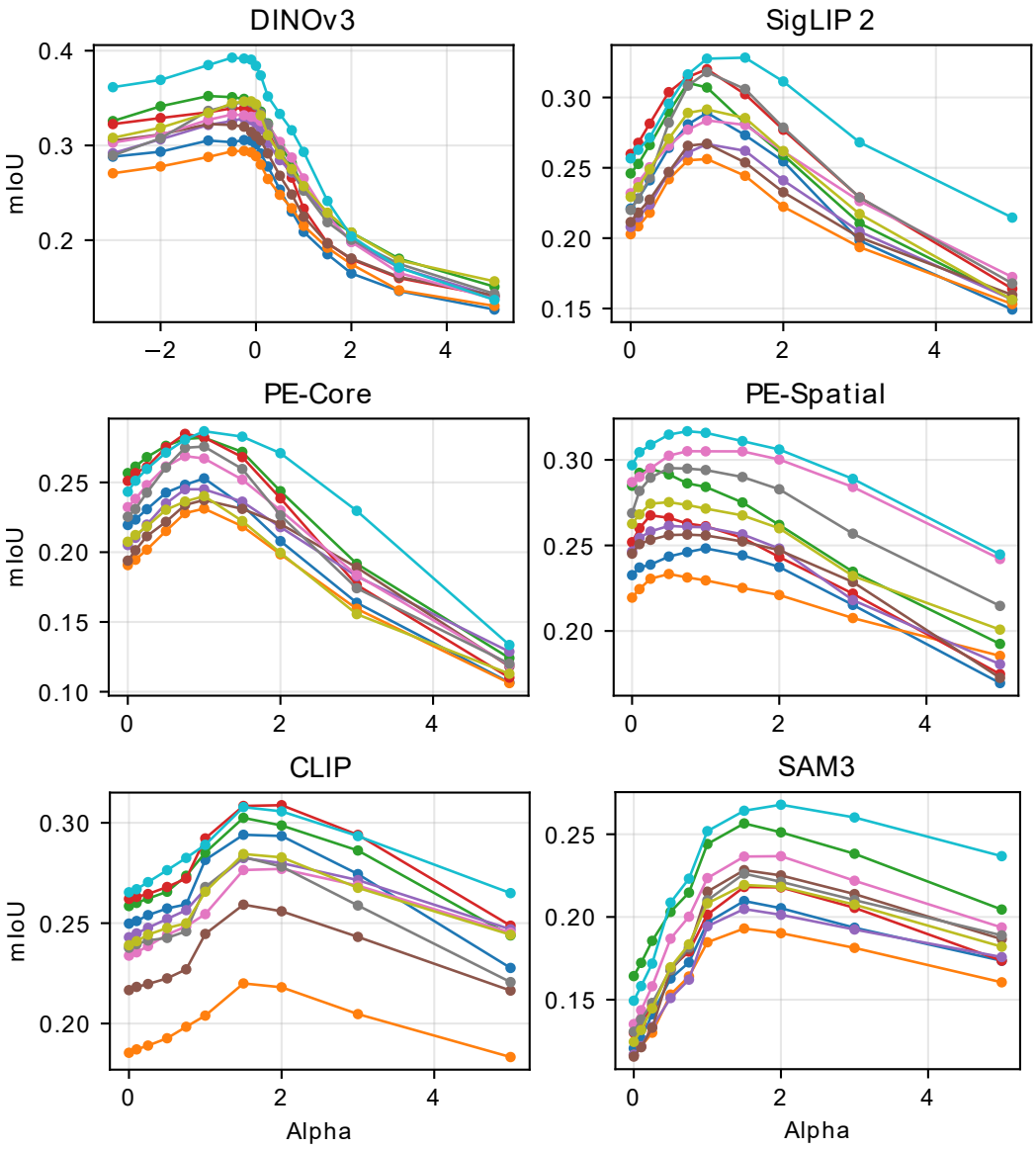


Figure 7: Repeats of the semantic segmentation probe on ADE20k described in Section 4.2. Each line corresponds to a random subset consisting of 30 (out of 150) classes. Results are preserved across repeats, showing that the choice of α does not depend on a specific set of semantic classes. CLIP and the Segment-Anything Model 3 (SAM3) ViT backbone display the same behavior as the noisy models discussed in the main text.

B Method Details

B.1 Construction of the Laplacian

We construct the graph Laplacian $\boldsymbol{L}$ on the two-dimensional lattice of patch tokens. Let $n = h \cdot w$ denote the number of tokens and index each token by its lattice position (i, j). In our experiments, the token lattice is square, such that $h = w = \sqrt{n}$. Two tokens are connected by an edge if and

only if they are direct horizontal or vertical neighbors. Thus, the resulting graph corresponds to a four-neighborhood grid without periodic boundary conditions.

Let $\boldsymbol{A} \in \mathbb{R}^{n \times n}$ denote the adjacency matrix of this graph,

$$A_{ij} = \begin{cases} 1, & \text{if tokens } i \text{ and } j \text{ are horizontal or vertical neighbors,} \\ 0, & \text{otherwise,} \end{cases} \tag{6}$$

and let $\boldsymbol{D}$ be the diagonal degree matrix with

$$D_{ii} = \sum_j A_{ij}. \tag{7}$$

The graph Laplacian is the standard combinatorial Laplacian

$$\boldsymbol{L} = \boldsymbol{D} - \boldsymbol{A}. \tag{8}$$

Consequently, interior tokens have diagonal entries $L_{ii} = 4$, edge tokens have $L_{ii} = 3$, and corner tokens have $L_{ii} = 2$, while $L_{ij} = -1$ for neighboring tokens and $L_{ij} = 0$ otherwise.

For any scalar field $\boldsymbol{v} \in \mathbb{R}^n$ defined on the token lattice, this construction satisfies

$$\boldsymbol{v}^\intercal \boldsymbol{L} \boldsymbol{v} = \sum_{i,j \in \mathcal{E}} (v_i - v_j)^2, \tag{9}$$

where each undirected edge i, j is counted once.

B.2 Experimental details

B.2.1 Models

The DINOv3 model is the original distilled ViT-B-16 version. SigLIP 2 is the ViT-B-16-512 model (meaning that it was trained at 512 pixels resolution). We used PE-Spatial-B16-512 and PE-Core-S16-384, which is the architecture and training recipe most similar to the other models. The CLIP model is the ViT-B OpenCLIP implementation. Finally, we used the official SAM3 release.

B.2.2 Dense Experiments

We implemented linear probes as 1×1 convolutions with bias and attached them to the final layer of each model. Class logits were bilinearly upsampled to pixel resolution to match the size of the label tensors. We utilized standard training splits and trained with the AdamW optimizer at 5e-4 learning rate until convergence. Probes were trained at resolutions of 512, 512, 288 on ADE20k, PASCAL and NYU, respectively. For fair evaluation of α, we evaluated the same linear probe (including equal weights) for all levels of α.

B.2.3 Global classification experiment

To test whether noisy dimensions are crucial for storing global image information, we ran an experiment similar to that described in Section 4.2 on the CUB dataset (Wah et al., 2011). The task is global image classification rather than a dense prediction. We used a simple attention-based probe consisting of a pair of trainable vectors: A query and a readout. During training, the probe had access to the full high-dimensional feature map. During evaluation, we kept the original dimensionality for computing attention values, but mapped to PC space and back before attention pooling. The pooled representation was then fed into the linear layer. For mathematical detail, see the analogous procedure described in Section 4.2. Probes were trained at 224px resolution with 1e-3 learning rate.