Unsupervised Semantic Segmentation Facilitates Model Understanding

Xiaoyan Yu[1,2,3,*], Jannik Franzen[1,2,4,5], Lisa Mais[1,2,5], Peter Hirsch[1,2], Nick Lechtenbörger[5], Andreas Mardt[1,2], and Dagmar Kainmueller[1,2,5,*]

[1] Max-Delbrück-Center (MDC), Berlin, Germany
{firstname.lastname}@mdc-berlin.de [*] Corresponding authors
[2] Helmholtz Imaging
[3] Humboldt-Universität zu Berlin, Berlin, Germany
[4] Charité Universitätsmedizin, Berlin, Germany
[5] University of Potsdam, Potsdam, Germany

Abstract. Self-supervised learning (SSL) has produced a diverse landscape of vision transformers (ViTs) whose pretrained representations support a wide range of downstream tasks. Towards a better understanding of these models, a body of works has assessed the mechanics of their self-attention as well as which types of information they capture across their representations, revealing, e.g., stark differences between models trained with contrastive learning (CL) vs. masked image modeling (MIM). However, the total of these advances on model understanding has to date not yet fully permeated a larger community, where, e.g., insights that are specific to CL models are still at times generalized to MIM models. To make model understanding straightforward and intuitive for a broad community, we propose a simple and easily interpretable visualization protocol. Our protocol is based on visualizing unsupervised semantic segmentation results – yet by no means do we focus on top segmentation performance. Instead, our protocol allows us to easily convey model behavior that consistently emerges across images. Benchmarked on a diverse set of SSL models across layers and representations, our protocol allows us to gain novel insights into distinct positional biases and scaling behaviors, including, e.g., strong boundary artifacts in DINOv3-Large model tokens. These novel insights come on top of more easily conveying a range of previous findings. Our protocol further allows us to clearly visually convey and distinguish between *positional effects* and the closely related yet distinct *locality bias*, the latter being much more extensively studied in the literature so far. Our protocol is publicly available[6], serving to catalyze further model understanding for a broad community.

Keywords: SSL · positional effect · locality bias · scaling behavior

1 Introduction

Advances in self-supervised learning (SSL) have led to a diverse range of vision transformers (ViTs) that capture rich visual semantics without relying on

[6] https://github.com/Kainmueller-Lab/ssl-rep-seg

human-annotated labels. These pretrained models have been widely and successfully adopted across a broad spectrum of downstream vision tasks, ranging from image-level tasks such as image classification to dense pixel-level tasks such as object detection and segmentation [13, 18, 27, 28]. To generalize effectively across tasks, models must encode semantic information at both the global image level and the local patch level. Here, semantic information refers to high-level, meaning-related aspects of an image, including instance-, class- and parts-level information about objects and background as well as broader scene properties such as spatial arrangement and conceptual similarity across images.

Many works have investigated how ViTs are able to achieve this. A particular focus has been on dissecting short-range vs. long-range attention [20, 26, 31], where short-range (*local*) attention has been termed *locality bias*. Locality bias constitutes *relative* positional information – i.e., an encoding of what is close to a particular location *within an individual image*. It has been found to be strong in MIM as opposed to CL[7] (while both model types feature long-range semantically meaningful attention), serving to explain MIM's improved performance in some downstream tasks [20]. Related yet distinct from locality bias, *positional effect*, as formally defined and studied in [11], refers to positional information that is *absolute* in the sense that it appears consistently *across images*. Findings on the benefits of locality bias over positional effect have e.g. prompted a pivot from traditional (absolute) positional embeddings to rotary (relative) ones [24, 25]. While *locality bias* has been conveyed in an easily interpretable form by means of attention map- and attention matrix visualizations (as e.g., in [3]), *positional effect* is lacking an established distinct and intuitive visualization to date, hindering straightforward insights into model behavior.

Besides positional information and respective attention mechanisms, the scaling behavior of SSL models has also been scrutinized, finding that large models often underperform their base variants on dense prediction tasks. This has been attributed to decreased patch consistency observed in some large models [24]; it could also be that some large models capture more fine-grained semantic structure – a non-detrimental effect not reflected by standard benchmarks [21]. However, large models' underperformance versus their base variants has not yet been studied under a hypothesis of potentially detrimental positional effects.

Despite the manifold advances on model understanding discussed above, the community at times still attributes the widely known findings of [1] to general SSL models albeit they mainly apply to CL, e.g. implying that key embeddings capture strong semantic information in general [35] while this only holds for CL, or that positional information degrades in later ViT layers in general while, again, this mainly holds for CL [17].

Towards making model understanding easy and intuitive for a broad community, we propose a simple and straightforward visualization protocol. Based on unsupervised semantic segmentation by scalable k-means clustering, our protocol reveals behaviors that emerge consistently across images, thus allowing us

[7] Note, by *CL* we also refer to joint embedding objectives that do not feature an explicit push force, see e.g. [30] for a respective discussion.

to convey not just classical semantic information, but also positional effect. The protocol itself is not new – in fact it has been standard practice in the unsupervised segmentation community [13,14,22] – albeit with an exclusive focus on downstream segmentation performance. Instead, we borrow this protocol for the purpose of mechanistic model understanding. Previous visualization techniques for model understanding either do not establish correspondences across images (e.g., [5] as well as all kinds of attention map- and matrix visualizations), or the few that do (e.g., [11]) are not designed to facilitate comparisons across models, or are hard to interpret in this regard. Our protocol for the first time clearly conveys positional effect as opposed to locality bias.

We benchmark our protocol on a range of models, for their keys, queries, values and tokens across all layers. Our benchmark includes eight SSL models (MAE [15], MoCov3 [7], Mugs [34], iBOT [33], DINO [5], DINOv2 [19], DINOv2+reg [9], DINOv3 [24]) and two baselines (a supervised ViT [12] and CLIP [16]). Our results convey many interesting findings from related work in a coherent, easily interpretable and intuitive form, including (i) optimal semantic structure typically emerging in intermediate layers rather than the final layer [3, 24], (ii) block artifacts in CLIP and DINOv2 tokens [32], (iii) a progressive loss of fine-grained semantic detail in deeper layers of supervised and CLIP models [26], (iv) detrimental scaling behavior due to a loss of patch consistency [24], and (v) pretext-task-specific locality biases and attention mechanisms [1, 20, 26, 31]. For the latter, we newly find that strong positional effect dominates keys and queries in MIM (cf. Fig. 1), which constitutes an "upstream" cause for the respective known locality bias. Beyond conveying and refining previous findings, our results yield novel insights into detrimental scaling behavior, suggesting that strong positional effects specific to DINOv3-*Large* tokens cause underperformance as compared to the respective Base model (cf. Fig. 5).

In summary, we contribute a simple and straightforward visualization protocol for qualitative assessment of ViT representations, complemented by contextual quantitative measures. A broad benchmark shows its power to facilitate model understanding and catalyze respective novel insights.

2 Proposed protocol for model understanding

Our goal is to facilitate and extend our understanding of how both semantic and positional information are encoded across different components of ViT [12] backbones, and how various SSL paradigms influence this structure. Our protocol to facilitate respective model understanding comprises the following core elements:

1. Unsupervised semantic segmentation by across-image cluster-based probing of embeddings (cf. Sec. 2.1), and
2. Joint visualization of segmentation results on exemplary sets of images alongside ground truth labels (cf. Sec. 2.2).

Putting visualizations of unsupervised segmentation results on a (random) set of exemplary images next to each other reveals where embeddings from same

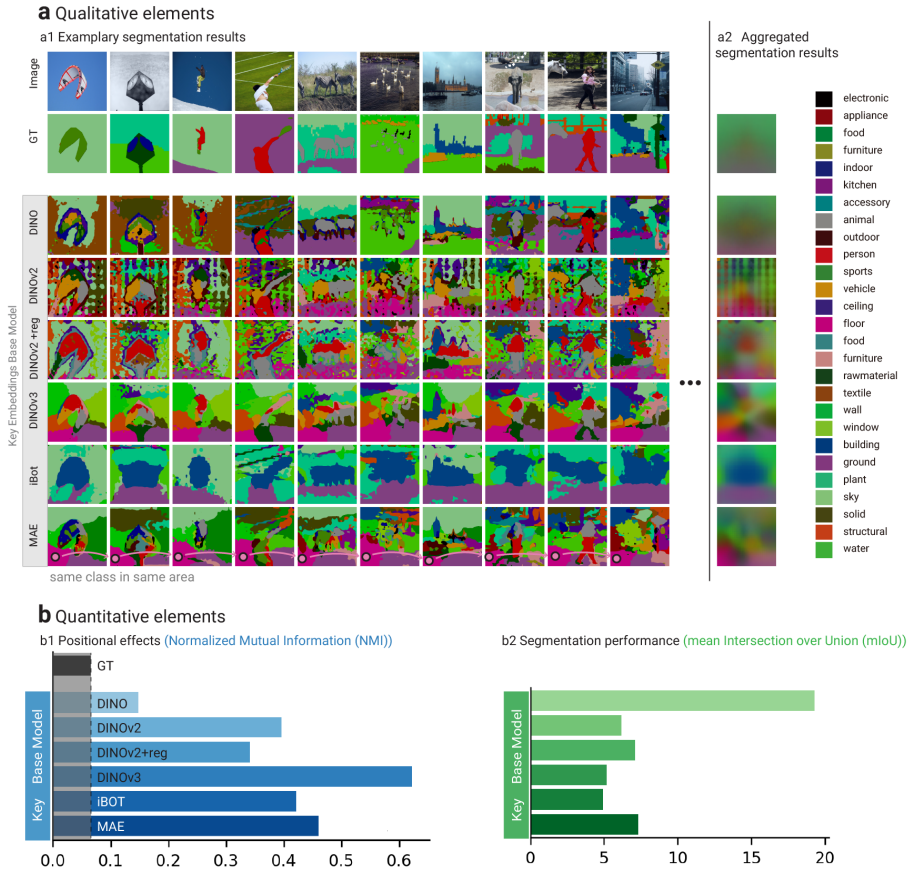

Fig. 1: Overview of our protocol. We here introduce the individual elements of the protocol, and at the same time showcase exemplary insights it facilitates. To this end, across rows, we vary the SSL training paradigm at fixed ViT-B architecture and fixed ViT layer selection strategy by best mIoU. **a)** Qualitative elements: (a1) Joint visualization of unsupervised semantic segmentation results on exemplary sets of images alongside respective ground truth label maps – **This reveals strong positional effects in key embeddings of MIM-based models**, including DINOv2, DINOv2+reg, DINOv3, iBOT and MAE. E.g., MAE key embeddings (bottom row) in the bottom-left corner of each individual image consistently cluster into a single class. In contrast, DINO, a CL model, exhibits substantially less positional effect in favor of more semantically meaningful structure. (a2) Aggregated visualization of unsupervised segmentation results as well as ground truth labels – This serves to confirm that the behavior observed in a1 extends across the whole dataset. **b)** Quantitative elements: (b1) Aggregate positional effect – This serves to quantitatively compare the positional effects as visually observed through the qualitative elements of the protocol, including the ground truth reference effect. (b2) Aggregate segmentation performance – This serves to quantitatively compare semantic structure as visually conveyed through the qualitative elements. For further insights, see Sec. 3 and Figs. 4 and 5.

clusters localize across images, intuitively visualizing dominant positional effects while simultaneously enabling standard qualitative assessment of segmentation performance, see Fig. 1(a1). To further capture a global, dataset-wide view on top of a (necessarily small) set of examples, we complement the above with:

3. aggregate visualization of results on the full dataset (cf. Sec. 2.2),
4. aggregate quantitative measure of positional effect (cf. Sec. 2.3), and
5. aggregate quantitative measure of segmentation quality (cf. Sec. 2.3),

see Figs. 1(a2) and 1(b). Importantly, all elements of the protocol (except aggregate segmentation quality) are also displayed for the ground truth labels, serving as a reference to gauge dataset-inherent spatial bias. The protocol applies to any kind of embedding from any ViT layer (cf. Fig. 2). It can also be applied multiple times, to different embeddings to facilitate comparative insights (cf. Sec. 2.4).

2.1 Unsupervised semantic segmentation

We leverage unsupervised semantic segmentation as a tool to probe embedding spaces. We argue that any fine-tuning regime toward a specific downstream task and dataset distorts the original embedding space, hence obscuring how SSL training governs the structure of learned embeddings. To stay clear of such effects, we deliberately opt for a bare-bones unsupervised clustering-based segmentation approach over complex pipelines tuned for SOTA segmentation performance [14, 22, 28, 29]. While the latter do attain stronger quantitative results, they introduce extensive empirical design choices and hyperparameter spaces, injecting confounding factors into the analysis. Our method of choice is *cluster-based probing* by means of across-image k-means clustering, as detailed below. Despite its simplicity, it forms the core of several SOTA approaches [13, 14].

Clustering-based probing: To directly probe the structure of pretrained embeddings, we use batch-wise clustering as in [13, 14]. Given a batch of images $\{\mathbf{x}_i\}_{i=1}^{B}$, we extract patch-level embeddings from a frozen SSL backbone, yielding feature embeddings $\mathbf{F}_i^{(l)} \in \mathbb{R}^{N \times d}$, where $N = H_p \times W_p$ is the number of patches, d is the embedding dimension, and l denotes the selected layer. We fit a codebook of K cluster centroids $\mathcal{C} = \{\mathbf{c}_k\}_{k=1}^{K} \subset \mathbb{R}^d$ to all patch embeddings across the batch via online k-means. Each embedding is assigned to its nearest centroid

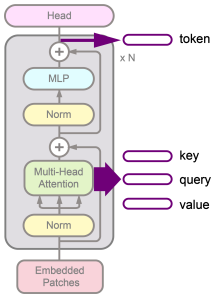

Fig. 2: Canonical embeddings produced by a ViT: keys, queries, and values of the Multi-Head Attention (MHA) blocks, and tokens, defined as the feed-forward network (FFN) outputs for patch tokens. Each of these embedding types is at hand at each layer of the ViT. To aggregate per-head embeddings yielded by the MHA block, for any given layer and type, we concatenate embeddings along the attention dimension. We then run cluster-based probing independently for each embedding type and at each layer of the ViT.

under cosine similarity, and the centroids are updated iteratively across batches, allowing us to capture cross-image semantic structure. See Appendix A.2 for details on hyperparameters and PCA-based initialization of centroids.

To produce pixel-wise semantic segmentation predictions, the patch-level embeddings $\mathbf{F}_i^{(l)} \in \mathbb{R}^{N \times d}$ are bilinearly upsampled to the original image resolution, yielding dense feature maps $\hat{\mathbf{F}}_i \in \mathbb{R}^{H \times W \times d}$. Each pixel is then assigned a semantic label via nearest-centroid lookup under cosine similarity:

$$\hat{y}_{hw} = \arg \max_{k \in \{1,\ldots,K\}} \frac{\hat{\mathbf{f}}_i^{(hw)} \cdot \mathbf{c}_k}{\left\| \hat{\mathbf{f}}_i^{(hw)} \right\|_2 \left\| \mathbf{c}_k \right\|_2}, \tag{1}$$

where $\hat{\mathbf{f}}_i^{(hw)} \in \mathbb{R}^d$ denotes the feature vector at spatial location (h, w).

Taken together, this approach provides a simple, interpretable, and semantically meaningful segmentation of images without introducing any learned decoder, complex post-processing, or iterative self-training.

2.2 Visualization of Results

Sets of Exemplary Images: We operate a standard unsupervised semantic segmentation setting, running Hungarian matching globally over the whole test set to map predicted clusters to ground-truth labels, and colorizing them respectively using the standard dataset color palette. See Fig. 1(a1) for examples.

Aggregate View: We construct an aggregated map from the class probability map $\mathbf{W} \in \mathbb{R}^{K \times H \times W}$, where $\mathbf{W}_{k,i,j}$ represents the average predicted probability of class k at pixel (i, j) over the entire test set. The per-pixel average probabilities are first normalized as $\tilde{\mathbf{W}}_{k,i,j} = \mathbf{W}_{k,i,j} / \sum_{k'} \mathbf{W}_{k',i,j}$, and the aggregated map $\mathbf{A} \in \mathbb{R}^{H \times W \times 3}$ is then computed as a weighted blend of the standard class colors $\mathbf{v}_k$ of this dataset as $\mathbf{A}_{i,j} = \sum_{k=1}^{K} \tilde{\mathbf{W}}_{k,i,j} \cdot \mathbf{v}_k$. See Fig. 1(a2) for examples.

2.3 Quantitative Measures

Positional Effect: To quantify positional effect in segmentation maps, we compute the class-wise mutual information (MI) $I_c(P; C)$ as follows:

$$I_c(P; c) = \sum_p \mathbb{P}(p, c) \log \frac{\mathbb{P}(p, c)}{\mathbb{P}(p)\mathbb{P}(c)} \tag{2}$$

where P denotes the discrete spatial position random variable with realization p, and c denotes a specific predicted class instance, and the empirical joint distribution $\mathbb{P}(p, c)$ is estimated from binarized pixel counts over all samples. The global MI and its normalized variant, normalized mutual information (NMI), are then defined as $I(P, C) = \sum_c I_c(P; c)$ and $I'(P, C) = \frac{I(P;C)}{H(C)}$, respectively, where $H(C) = -\sum_c \mathbb{P}(c) \log(\mathbb{P}(c))$ is the entropy of the predicted class distribution. The global NMI summarizes positional bias across all predicted classes, and enables comparability across embedding types and model families. See Fig. 1(b1)

for examples. Optionally, the class-wise MI, visualized for highest-scoring classes, may convey further insightful context. See Fig. 5(b3) for examples.

Segmentation Quality: Following standard practice for unsupervised semantic segmentation evaluation, we employ mean Intersection-over-Union (mIoU) computed under optimal label permutation via Hungarian matching between predicted clusters and ground-truth labels. See Fig. 1(b2) for examples.

2.4 Scenarios of Interest

Beyond insights into individual embedding spaces, applying our protocol multiple times and putting results next to each other, we can gain a range of *comparative* insights, including the following scenarios of interest:

- To observe differences across training paradigms: Pick one architecture, embedding type, and layer selection strategy – vary the training paradigm;
- To observe how model behavior evolves across ViT layers: Pick one model and embedding type – vary the layer;
- To observe differences across model sizes: Pick one training paradigm, embedding type, and layer selection strategy – vary the model size.

Regarding layer selection strategies, we can straightforwardly pick identical layers given a fixed architecture. Alternatively, we can pick layers with corresponding properties, like e.g. for any given model and embedding type, pick the layer that achieves best unsupervised semantic segmentation performance in terms of mIoU. The latter serves the specific purpose of gaining insights into the extent to which, for any given model, its embedding space with best emerging semantic structure is confounded by positional effect. Furthermore, it has the additional advantage that it is applicable for fixed as well as varying model architecture.

3 Results and Discussion

3.1 Experimental setup

Datasets: A dataset ideally suited for model understanding by means of our protocol would have no *inherent* spatial bias, as dataset-inherent spatial bias acts as a confounder by entangling true semantic structure with positional effect. Furthermore, a suitable dataset necessarily needs to allow for meaningful unsupervised semantic segmentation benchmarking of SSL ViTs. We use COCO-Stuff [4] as the main dataset because it has the least dataset-inherent spatial bias among common unsupervised segmentation benchmarks. In particular, it features diverse scene layouts and respective weaker spatial-semantic correlations than a number of other, more object-centric benchmarks. It further features diverse background class labels, allowing us to assess a model's ability to distinguish semantic background content (e.g., sky, grass, ceiling), not just foreground objects (e.g., people, animals). Following [13, 14, 22], we chose a 27-class subset of COCO-stuff. That said, to ensure that our study does not merely reveal

COCO-Stuff-inherent behaviors or hidden biases, we additionally evaluated on Cityscapes [8] and the PascalPart [6] animals subset, see Appendix A.7.

SSL models: All models in our study share a Vision Transformer (ViT) backbone [12], enabling architecture-aligned comparisons. We evaluate eight representative SSL frameworks – MAE [15], MoCov3 [7], Mugs [34], iBOT [33], DINO [5], DINOv2 [19], including its register-token variant [9], and DINOv3 [24] – and two supervised baselines: a standard ImageNet-supervised ViT [12] and CLIP [16]. ViT-Base serves as the primary architecture, with ViT-Large included to assess scaling effects. Key characteristics of each model are listed in Appendix A.1.

Experiments: We run the following setups of our protocol:

- We vary the model training paradigm, at fixed ViT-B architecture and fixed layer selection policy based on best segmentation performance.
- We vary the layer (from first to last), for some fixed ViT-B models.
- We vary model size at fixed training paradigm and fixed layer selection policy based on best segmentation performance.

3.2 Results

Unless otherwise specified, results presented here are for ViT-B models on COCO-Stuff. Corresponding ViT-L results as well as additional models and datasets omitted here for clarity are provided in Appendices A.5, A.6 and A.7. Regarding layer selection by best segmentation performance, Figure 3 lists the best-performing layer for each model and embedding type. Note that we pick last-layer tokens *after* the final LayerNorm; see Appendix A.12 for a respective ablation.

Behavior across Varying Training Paradigms: Figure 1 shows results for the key embeddings from the best-performing (in terms of segmentation mIoU) layer of each of several MIM-based models, alongside DINO as a CL baseline. For MIM, as opposed to CL, embeddings cluster consistently across images according to their global location, indicating strongly dominant positional effect. This observed behavior is related yet distinct from previous findings regarding locality bias in MIM vs. CL models [1, 3, 24]. We further discuss the relation between positional effect and locality bias in Sec. 3.3(1).

At the same time, MIM keys from peak-performing layers exhibit considerably inferior segmentation performance (mIoU) as compared to CL, see also Fig. 3: Here, MIM keys and queries form a subset (highlighted by blue outline) of notably lower performance compared to other models and embedding types. This gives rise to the hypothesis that pronounced positional effect causes degraded segmentation performance (see Sec. 3.3(2) for further discussion).

Deserving of mention is the fact that observable positional effect is already inherent in the ground-truth (GT) semantic labels themselves, arising from characteristic spatial regularities in natural image statistics. E.g., *humans* are disproportionately concentrated near the image center, *sky* occupies the upper region, and *ground* predominantly appears in the lower half. Nevertheless, the positional

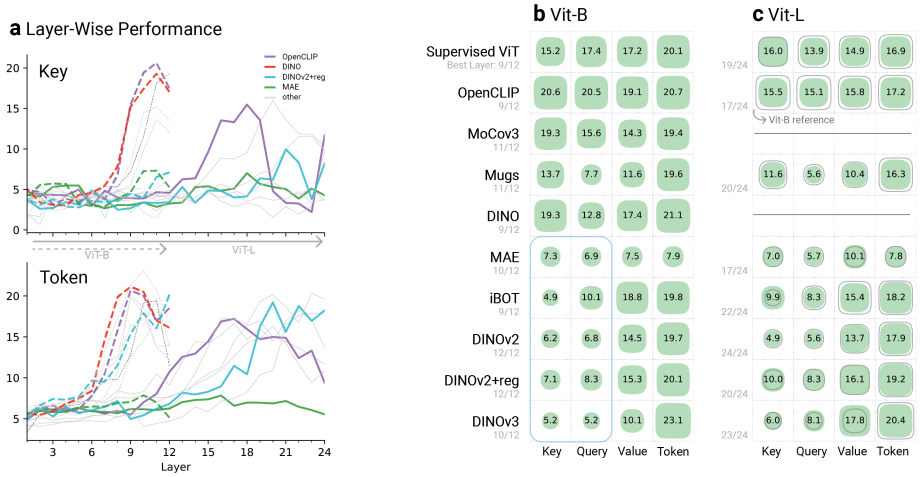

Fig. 3: Performance of **key**, **query**, **value**, and **token** embeddings in the **unsupervised semantic segmentation** task across SSL models and model sizes. **a)** displays the layer-wise segmentation performance based on key and token embeddings for four selected models. **b)**/**c)** present the overall performance of all models, using the respective best-performing layer for ViT-Base and ViT-Large. Models highlighted in blue exhibit notably low segmentation performance with query and key embeddings.

effects in MIM keys substantially exacerbate this tendency, as revealed by respective aggregate visualizations and quantitative measures (cf. Fig. 1(a2) and (b1)): Across all models we observe elevated global NMI relative to GT.

As a side note, block-structured artifacts are clearly visible in DINOv2 results, which have been reported and attributed to positional embeddings in [32]. See Appendix A.5 for further respective results, also including CLIP.

Behavior across Layers: Figure 4 provides results across layers for keys (top) and tokens (bottom) of DINO, MAE and DINOv3. Regarding keys, DINO exhibits positional effects in early layers (layers 2 and 3), which diminish progressively in deeper layers. In contrast, MAE shows strong positional effects only in later layers (layer 5), while DINOv3 is dominated by positional effect consistently across all layers. Regarding tokens, positional effect is substantially reduced compared to keys across all models; DINOv3 remains a notable exception, as its last layers exhibit a sharp onset of strong positional effect.

DINOv3's comparatively strong positional effects across embeddings and layers is surprising as its use of RoPE makes positional embeddings relative as opposed to absolute – thus we would expect them to manifest as locality bias rather than positional effect, or at least expect diminished positional effects compared to models not using RoPE – yet we observe the opposite.

Scaling Behavior: Figure 5 provides results across scales, namely ViT-B vs. ViT-L, for DINOv3, iBOT, DINOv2 and DINOv2+reg token embeddings. We highlight a particularly notable finding concerning the positional effect in

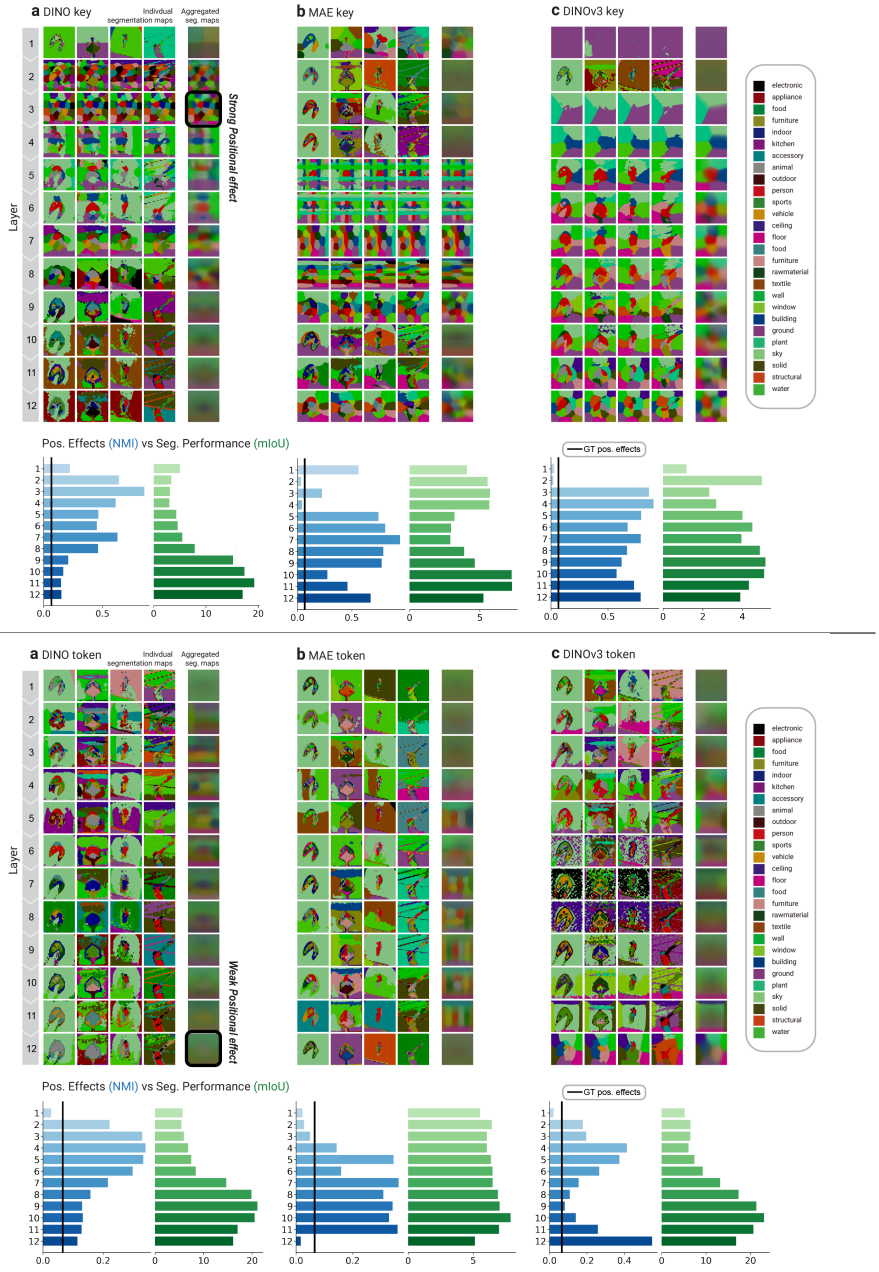

Fig. 4: Model behavior across layers: Unsupervised semantic segmentation results for three exemplary models: **a)** DINO, **b)** MAE, and **c)** DINOv3. **Top: Key embeddings:** The three models exhibit distinct patterns of positional effect: it is confined to earlier layers in DINO, strongly emerges in intermediate layers in MAE, and persists strongly across all layers in DINOv3. **Bottom: Token embeddings:** Positional effect is largely reduced compared to key embeddings across models. Notably, however, DINOv3 remains an exception, with its last layer still exhibiting strong positional effect.

DINOv3-Large: Here, individual and aggregated segmentation maps reveal that the upper and lower regions of images are heavily affected by positional effect, forming spurious clusters systematically mis-assigned as *ceiling* and *floor* irrespective of ground-truth class. This positional effect is also partially visible along vertical boundaries, where affected regions are incorrectly segmented as a single *solid* class. In comparison, the DINOv3-Base variant, while also affected, exhibits less pronounced positional effect, only in the lower boundary region. Accordingly, DINOv3-Large shows disproportionately high class-wise MI values (cf. Fig. 5(b3)) for semantically spurious classes such as *floor*, *ceiling*, *solid*, as well as high global NMI as compared to its Base counterpart. We hypothesize that this pronounced positional effect in the Large variant is a key contributor to DINOv3's degraded scaling behavior, as discussed in more depth in Sec. 3.3.

Beyond DINOv3, we observe further model-specific patterns that become more pronounced with scale, see Fig. 5: DINOv2-Large displays a similarly strong positional effect as DINOv3, manifesting as consistent cluster assignments tied to fixed spatial regions across images. In contrast, DINOv2+reg largely mitigates this effect. However, it produces noticeably noisier segmentations, though boundary localization itself does not appear to degrade. This increased noise aligns with patch inconsistencies reported in [24] for larger models.

Regarding iBOT, both Base and Large variants exhibit pronounced positional effects; see e.g. the recurring appearance of a "vehicle-like" segment in the lower central region of images irrespective of semantic content. However, cluster boundaries still tend to respect the underlying semantic structure of the images. Besides positional effect, we also notice that iBOT exhibits more fine-grained and semantically consistent clustering as model size increases: In the Large variant, body parts such as heads, legs, and arms are more consistently grouped across humans and animal species compared to the Base model (see also Fig. 6). Increasing the number of clusters further reveals more fine-grained, position-aware partitions of these body parts as well as other objects (see Appendix A.6).

In summary, as observed quantitatively in Fig. 3, increasing model size does not necessarily translate into improved performance in downstream semantic segmentation. In many cases, Large models underperform their Base counterparts, consistent with observations in [3, 24]. Note that this is in contrast to zero-shot semantic keypoint correspondence, where we observe favorable scaling behavior (cf. Appendix A.8), similar to reports for depth estimation [10]. While [2, 24] attribute inverse scaling to a pretext/downstream task misalignment and loss of patch consistency, respectively, we observe increased positional effect as an additional, previously unreported contributor, in particular for DINOv3.

3.3 Extended Insights and Discussion

(1) Positional effect causes known locality bias in MIM: Previous works have repeatedly shown the existence or diminishing of *locality bias* across layers [1, 3, 24], also revealing respective distinct behaviors of MIM vs. CL. We here replicate and expand upon these findings (see Fig. 7), identifying positional effects in keys and queries as an upstream cause for known locality biases.

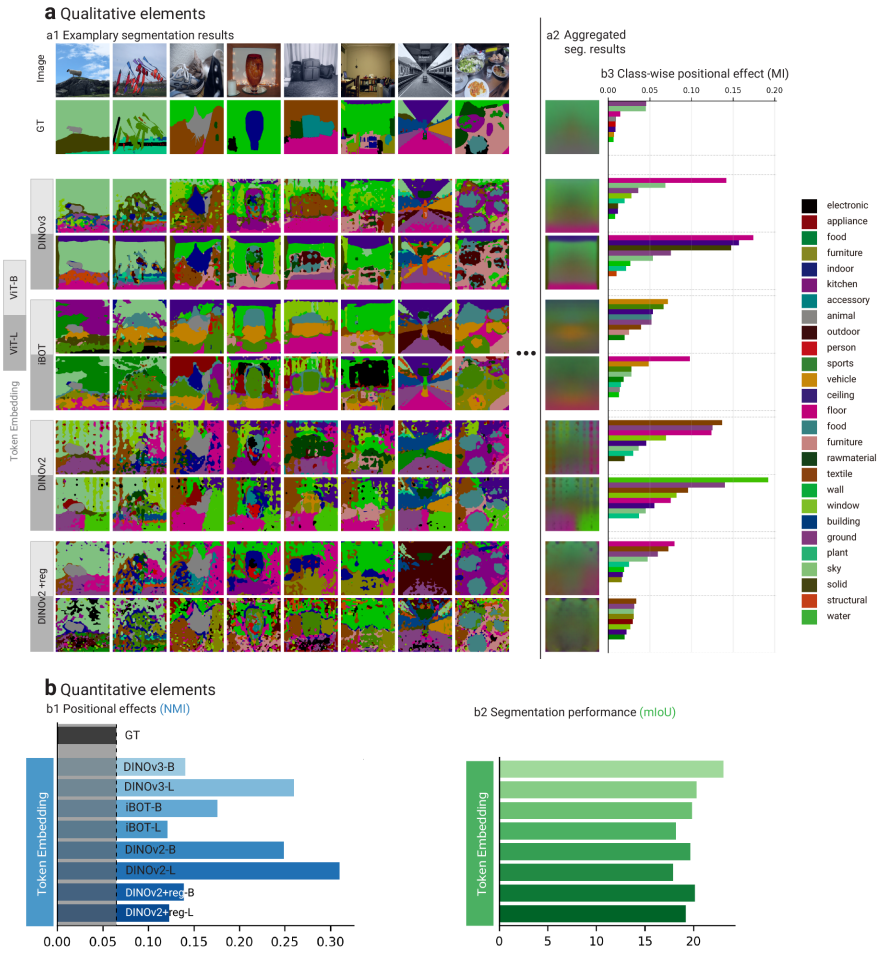

Fig. 5: Behavior across scales: Clustering results for **token embeddings** from the best-performing layer of the Base and Large variants of DINOv3, iBOT, DINOv2 and DINOv2+reg. We observe a notably increased positional effect in DINOv3-L compared to its -B counterpart alongside inferior segmentation performance in terms of mIoU. DINOv2 exhibits a similar pattern. For iBOT, we observe notable positional effects across sizes, yet to a lesser extent and along with a shift in clusters in the -L variant, as further studied in Fig. 6. For DINOv2 with registers, increasing model size leads to a larger number of small, fragmented clusters, consistent with observations in [24].

In more detail, it has been shown that the vertical stripes in attention matrices (see Fig. 7(a)) reflect two phenomena: If the stripe corresponds to a background position, this indicates the use of background locations to store global information [9]. If the stripe corresponds to a foreground position, this has been termed *query collapse* and indicates that even most background queries attend

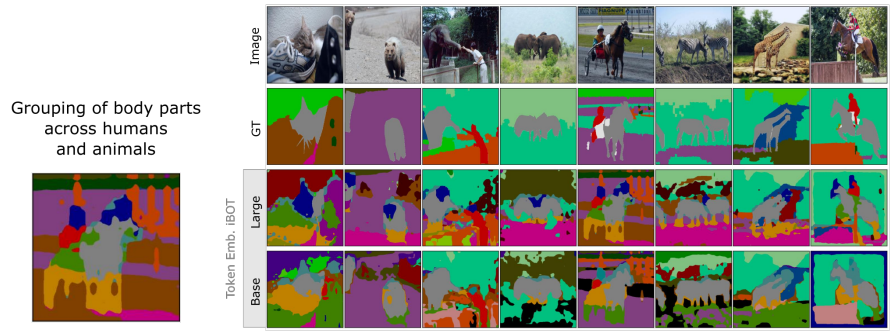

Fig. 6: Positive scaling effect in iBOT: Unsupervised segmentation results for token embeddings from the best-performing layer of the Base and Large variants of iBOT focusing on animal images. We observe more fine-grained clustering of object parts for the larger model. For example, in the Large model, head regions of humans and different animals are assigned to more similar embeddings compared to the Base variant.

to the foreground object [20]. The diagonals, on the other hand, reflect locality bias – i.e., independent of position, queries attend to nearby positions [3].

To compare these effects quantitatively across models/layers, we borrow the information-theoretic notion of MI [23], applied to query–key attention maps; i.e., we compute the MI $I(Q, K)$ per image/head and aggregate to yield $\hat{I}(Q, K)$, both defined in Appendix A.3 Eqs. (3) to (4). This proxy measure of locality bias yields higher values for diagonal/local query–key structures (see Fig. 7(c)).

We can thus quantitatively confirm that locality bias decreases for CL models in deeper layers. For MAE it emerges at mid-depth and stays elevated until the end (see Fig. 7(d)). DINOv2+reg shows the strongest and most persistent locality bias; DINOv3 has a similarly persistent bias but at a lower level.

As expected, we find strong correlation between key/query positional effect and locality bias in terms of $\hat{I}(Q, K)$ ($\rho = 0.70$ with $p < .001$, see Fig. 8 (**right**)). We deem consistent positional effects in keys and queries at any given layer causal for downstream locality bias: Attention values that serve to quantify locality bias are directly computed as (normalized) dot products of their upstream keys and queries; thus downstream locality bias directly follows.

(2) Positional effect correlates negatively with segmentation performance: We quantified the correlation between positional effect and segmentation mIoU, finding significant negative correlation ($\rho = -0.59$ with $p < .001$, see Fig. 8 (**left**)). Regarding respective causality: Any delta positional effect over the ground truth labels' effect constitutes segmentation error; however, not all segmentation error is due to (adverse) positional effect. We hypothesize that the observed inverse scaling behavior of DINOv3 can be attributed to the measured increased positional effect because our visualization suggests notable impact; yet there are confounders we cannot disentangle, e.g., potentially shifting (mis-) alignment of model-encoded clusters and GT labels (as discussed in Sec. 3.2).

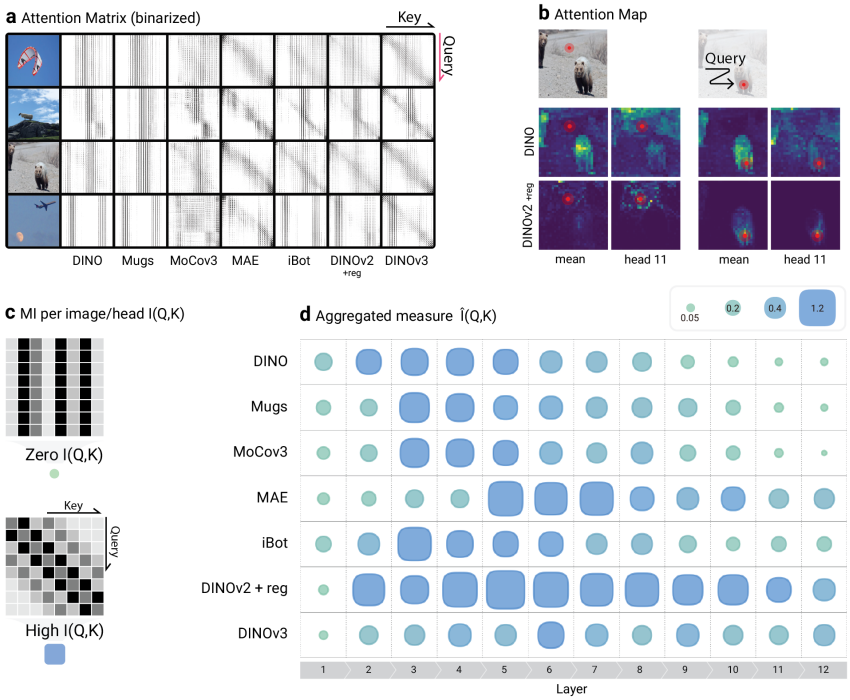

Fig. 7: Locality Biases observable in Attention Matrices. **a)** Attention matrices for multiple models, averaged over attention heads. For improved visualization, maps were binarized w.r.t. their 95th percentile. **b)** Exemplary attention maps for two different queries, illustrating the stronger locality of attention in DINOv2+reg compared to DINO. **c)** Vertically "barcoded" attention patterns indicate zero mutual information $I(Q, K)$ between respective keys and queries, whereas diagonal structures correspond to high $I(Q, K)$. **d)** Aggregated measure $\hat{I}(Q, K)$, averaged over images and heads. Notably, CL models (DINO, Mugs, MoCov3) exhibit a pronounced decrease in $\hat{I}(Q, K)$ towards deeper layers, while MIM models maintain a higher mean.

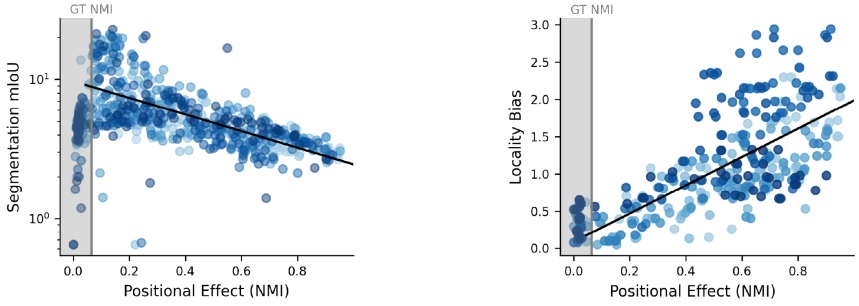

Fig. 8: Left: Correlation between positional effect and segmentation performance (in log-scale) for all models and layers, across Keys, Queries, Values, and Tokens. **Right:** Correlation between positional effect and locality bias in Key and Query Embeddings.

(3) Tokens yield highest segmentation performance: Our quantitative evaluation of segmentation performance in Fig. 3 shows that token embeddings consistently achieve the highest performance across models (the only exception being MAE-Large). The same trend is observed for the semantic correspondence task (see Appendix A.9). This finding contrasts with the results of [1], who report superior performance of key embeddings in keypoint matching. We attribute this discrepancy primarily to the limited model scope of [1], which evaluates exclusively on DINO, a model where the performance gap between key and token embeddings is relatively small, and on only 360 sampled image pairs. Similarly, prior works on unsupervised segmentation pipelines (e.g., TokenCut and CutLER [28, 29]) report that key embeddings from DINO can outperform token embeddings. However, these methods address per-image object discovery rather than dataset-level semantic segmentation: Their graph-based clustering is performed *independently for each image*. Hence their conclusions apply to object- or region boundaries, yet not necessarily to *across-image* semantic classes.

At the same time, our layer-wise analysis of unsupervised semantic segmentation in Fig. 3 and Appendix Figs. 10 and 14 (for the keypoint correspondence task) confirms previous findings that, across model families, optimal downstream performance most often emerges in intermediate rather than the final layers [3, 24].

4 Conclusion

This work proposes a simple and straightforward visualization protocol for intuitive understanding of ViT representations. We employ it in a systematic benchmark of modern self-supervised vision models, across layers and embedding types. Evaluating eight representative SSL models together with supervised and CLIP baselines, we provide a unified and architecture-controlled analysis of how different pretraining objectives shape ViT representations.

Our benchmark yields several previously unreported insights on top of intuitively and consistently visualizing a broad range of previously reported findings. Regarding novel insights, we find that **dominant positional effect in keys and queries causes locality bias** previously observed for Masked Image Modeling. Furthermore, **inverse scaling of DINOv3 appears to stem from strong positional effect**, calling for a thorough reassessment of its training paradigm (including distillation) as well as its use of RoPE.

We release all code and results, serving as a resource for understanding *where* and *how* semantic information and positional effect emerge in large-scale vision pretraining, thus facilitating further model understanding and advancement.

Acknowledgments: This work was supported by the Helmholtz Einstein International Berlin Research School in Data Science (HEIBRiDS), German Research Foundation (DFG) Research Training Group CompCancer (RTG2424), DFG Collaborative Research Center FONDA (SFB 1404, project no. 414984028), and the Synergy Unit of the Helmholtz Foundation Model Initiative.

References

1. Amir, S., Gandelsman, Y., Bagon, S., Dekel, T.: Deep vit features as dense visual descriptors. arXiv preprint arXiv:2112.05814 **2**(3), 4 (2021)
2. Balestriero, R., LeCun, Y.: Lejepa: Provable and scalable self-supervised learning without the heuristics (2025), https://arxiv.org/abs/2511.08544, accessed: 2026-06-25
3. Bolya, D., Huang, P.Y., Sun, P., Cho, J.H., Madotto, A., Wei, C., Ma, T., Zhi, J., Rajasegaran, J., Rasheed, H., et al.: Perception encoder: The best visual embeddings are not at the output of the network. arXiv preprint arXiv:2504.13181 (2025)
4. Caesar, H., Uijlings, J., Ferrari, V.: Coco-stuff: Thing and stuff classes in context. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 1209–1218 (2018)
5. Caron, M., Touvron, H., Misra, I., Jégou, H., Mairal, J., Bojanowski, P., Joulin, A.: Emerging properties in self-supervised vision transformers. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 9650–9660 (2021)
6. Chen, X., Mottaghi, R., Liu, X., Fidler, S., Urtasun, R., Yuille, A.: Detect what you can: Detecting and representing objects using holistic models and body parts. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 1971–1978 (2014)
7. Chen, X., Xie, S., He, K.: An empirical study of training self-supervised vision transformers. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 9640–9649 (2021)
8. Cordts, M., Omran, M., Ramos, S., Rehfeld, T., Enzweiler, M., Benenson, R., Franke, U., Roth, S., Schiele, B.: The cityscapes dataset for semantic urban scene understanding. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 3213–3223 (2016)
9. Darcet, T., Oquab, M., Mairal, J., Bojanowski, P.: Vision transformers need registers. arXiv preprint arXiv:2309.16588 (2023)
10. Dehghani, M., Djolonga, J., Mustafa, B., Padlewski, P., Heek, J., Gilmer, J., Steiner, A.P., Caron, M., Geirhos, R., Alabdulmohsin, I., et al.: Scaling vision transformers to 22 billion parameters. In: International conference on machine learning. pp. 7480–7512. PMLR (2023)
11. Doshi, F.R., Fel, T., Konkle, T., Alvarez, G.: Bi-orthogonal factor decomposition for vision transformers (2026), https://arxiv.org/abs/2601.05328, accessed: 2026-06-25
12. Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., et al.: An image is worth 16x16 words: Transformers for image recognition at scale. arXiv preprint arXiv:2010.11929 (2020)
13. Hahn, O., Araslanov, N., Schaub-Meyer, S., Roth, S.: Boosting unsupervised semantic segmentation with principal mask proposals. arXiv preprint arXiv:2404.16818 (2024)
14. Hamilton, M., Zhang, Z., Hariharan, B., Snavely, N., Freeman, W.T.: Unsupervised semantic segmentation by distilling feature correspondences. arXiv preprint arXiv:2203.08414 (2022)
15. He, K., Chen, X., Xie, S., Li, Y., Dollár, P., Girshick, R.: Masked autoencoders are scalable vision learners. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 16000–16009 (2022)

16. Ilharco, G., Wortsman, M., Wightman, R., Gordon, C., Carlini, N., Taori, R., Dave, A., Shankar, V., Namkoong, H., Miller, J., et al.: Openclip (2021)
17. Li, Y., Salehi, S., Ungar, L., Kording, K.: Does object binding naturally emerge in large pretrained vision transformers? Advances in Neural Information Processing Systems **38**, 3394–3423 (2026)
18. Liu, Z., Lin, Y., Cao, Y., Hu, H., Wei, Y., Zhang, Z., Lin, S., Guo, B.: Swin transformer: Hierarchical vision transformer using shifted windows. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 10012–10022 (2021)
19. Oquab, M., Darcet, T., Moutakanni, T., Vo, H., Szafraniec, M., Khalidov, V., Fernandez, P., Haziza, D., Massa, F., El-Nouby, A., et al.: Dinov2: Learning robust visual features without supervision. arXiv preprint arXiv:2304.07193 (2023)
20. Park, N., Kim, W., Heo, B., Kim, T., Yun, S.: What do self-supervised vision transformers learn? (2023), https://arxiv.org/abs/2305.00729, accessed: 2026-06-25
21. Rossetti, S., Pirri, F.: Hierarchy-agnostic unsupervised segmentation: parsing semantic image structure. Advances in Neural Information Processing Systems **37**, 98898–98935 (2024)
22. Seong, H.S., Moon, W., Lee, S., Heo, J.P.: Leveraging hidden positives for unsupervised semantic segmentation. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 19540–19549 (2023)
23. Shannon, C.E.: A mathematical theory of communication. The Bell system technical journal **27**(3), 379–423 (1948)
24. Siméoni, O., Vo, H.V., Seitzer, M., Baldassarre, F., Oquab, M., Jose, C., Khalidov, V., Szafraniec, M., Yi, S., Ramamonjisoa, M., et al.: Dinov3. arXiv preprint arXiv:2508.10104 (2025)
25. Su, J., Ahmed, M., Lu, Y., Pan, S., Bo, W., Liu, Y.: Roformer: Enhanced transformer with rotary position embedding. Neurocomputing **568**, 127063 (2024)
26. Walmer, M., Suri, S., Gupta, K., Shrivastava, A.: Teaching matters: Investigating the role of supervision in vision transformers. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 7486–7496 (2023)
27. Wang, W., Xie, E., Li, X., Fan, D.P., Song, K., Liang, D., Lu, T., Luo, P., Shao, L.: Pyramid vision transformer: A versatile backbone for dense prediction without convolutions. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 568–578 (2021)
28. Wang, X., Girdhar, R., Yu, S.X., Misra, I.: Cut and learn for unsupervised object detection and instance segmentation. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 3124–3134 (2023)
29. Wang, Y., Shen, X., Yuan, Y., Du, Y., Li, M., Hu, S.X., Crowley, J.L., Vaufreydaz, D.: Tokencut: Segmenting objects in images and videos with self-supervised transformer and normalized cut. IEEE transactions on pattern analysis and machine intelligence **45**(12), 15790–15801 (2023)
30. Wu, Z., Zhang, J., Pai, D., Wang, X., Singh, C., Yang, J., Gao, J., Ma, Y.: Simplifying dino via coding rate regularization. arXiv preprint arXiv:2502.10385 (2025)
31. Xie, Z., Geng, Z., Hu, J., Zhang, Z., Hu, H., Cao, Y.: Revealing the dark secrets of masked image modeling. In: 2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 14475–14485 (2023). https://doi.org/10.1109/CVPR52729.2023.01391
32. Yang, J., Luo, K.Z., Li, J., Deng, C., Guibas, L., Krishnan, D., Weinberger, K.Q., Tian, Y., Wang, Y.: Denoising vision transformers. In: European Conference on Computer Vision. pp. 453–469. Springer (2024)

33. Zhou, J., Wei, C., Wang, H., Shen, W., Xie, C., Yuille, A., Kong, T.: ibot: Image bert pre-training with online tokenizer. arXiv preprint arXiv:2111.07832 (2021)
34. Zhou, P., Zhou, Y., Si, C., Yu, W., Ng, T.K., Yan, S.: Mugs: A multi-granular self-supervised learning framework. arXiv preprint arXiv:2203.14415 (2022)
35. Zhou, T., Xia, W., Zhang, F., Chang, B., Wang, W., Yuan, Y., Konukoglu, E., Cremers, D.: Image segmentation in foundation model era: A survey. arXiv preprint arXiv:2408.12957 (2024)