Dense Attentive Probing: Dense output with less than 100K learnable parameters.

Anonymous authors
Paper under double-blind review

Abstract

The paradigm of pretraining a backbone on a large set of (often unlabeled) images has gained popularity. The quality of the resulting features is commonly measured by freezing the backbone and training different task heads on top of it. However, current evaluations cover only classifications of whole images or require complex dense task heads which introduce a large number of parameters and add their own inductive biases. In this work, we propose dense attentive probing, a parameter-efficient readout to make dense prediction using arbitrary backbones independent of the size and resolution of their feature volume. To this end, we utilize a masked cross-attention layer with learnable mask sizes which enables dense prediction with a small parameter budget, thus providing relatively unbiased access to the features. We employ this method to evaluate common backbones in three dimensions: instance awareness, local semantics and spatial understanding. We find that DINOv2 outperforms all other backbones tested – including those supervised with masks and language – across all three task categories. Furthermore, our analysis suggests that self-supervised training tends to yield features that separate object instances better than vision-language models. Code is available at https://to.be.released.

1 Introduction

Driven by the success of self-supervised learning (Chen et al., 2020; He et al., 2022; Oquab et al., 2023) and vision-language training (Radford et al., 2021), in many computer vision tasks, training from scratch has largely been replaced by fine-tuning large pretrained backbones. Ideally, the latter provide powerful features such that in the fine-tuning step only a small number of parameters needs to be modified and no large datasets are required. Many factors influence the feature quality of backbones: for instance, the pretraining paradigm, model architecture and the training data. Therefore, for both computer vision scientists and practitioners, it is crucial to characterize strengths and weaknesses of large pretrained backbones through systematic benchmarks. For whole-image classification (i. e. predicting one label per image) such benchmarks exist, for example, ImageNet (Russakovsky et al., 2014), VTAB (Zhai et al., 2019) and FGVC (Jia et al., 2022). They often rely on the established approaches of linear and attentive probing. Linear probing applies global average pooling and then linearly maps the resulting feature vector to class predictions while attentive probing uses cross-attention with a query token that predicts the class. Both techniques are not applicable for dense prediction tasks (i. e. predicting a label for each pixel). For example, in linear probing the prediction has the same spatial resolution as the feature volume, which is usually too low to capture object structures. Furthermore, the resolution of the feature volume varies across different backbones, preventing a fair comparison. Therefore, often common task heads (e. g. UPerNet by Chen et al. (2024a)) are used at the price of adding a large number of parameters and introducing additional inductive biases. Thus, the resulting backbone feature quality measurements are mediated by compatibility and performance of these heads.

Here we address the problem of assessing and comparing the representational quality of dense feature volumes. To this end, we measure dense prediction performance of feature backbones as directly as possible by proposing a novel dense equivalent to attentive probing. Our model consists of a single masked cross-attention layer, introduces only a small number of parameters (less than 100K, often even less than 50K)

Model	learnable parameters
ConvAdapter (Chen et al., 2024a)	>24M
ViT-Adapter (Chen et al., 2022b)	2.5M to 23.7M
Conv LoRA (Zhong et al., 2024)	$\sim$ 4M
Dense Attentive Probing (ours)	0.05M to 0.5M

Table 1: Adapter methods for dense prediction with the number of learnable parameters.

and adds little computational overhead over the backbone. By using cross-attention, we decouple the size and resolution of the input image and encoder output from that of the dense output, i. e. generate outputs at any resolution. We introduce a learnable masking radius in the cross-attention layer, which allows the readout to work with variable feature locality.

Balestriero et al. (2023) note the lack of a standardized evaluation protocol for self-supervised learning methods on dense prediction tasks. We believe our dense attentive probing method can address this gap and experimentally assess the quality of various supervised, self-supervised, vision-language training methods. Specifically, we use the new readout to characterize features along these three dimensions: (1) instance disentanglement, i. e. how well are individual instances recognizable from the features; (2) local semantics, evaluating how meaningful the features are for a local classification; and (3) spatial understanding, which assesses how well is the 3d structure of the scene captured.

2 Related work

Representation learning Self-supervised representation learning has been a popular research topic with multiple approaches that can roughly be categorized into joint-embedding (Chen et al., 2020; 2021; Caron et al., 2021) and reconstruction-based (He et al., 2022). DINOv2 is based on the iBot (Zhou et al., 2021) method which uses a joint-embedding architecture in combination with self-distillation and reconstruction. VicRegL (Bardes et al., 2021) and many other recent methods explicitly addresses local features by modeling losses at the token level. There has been a discussion about which techniques leads to better and more efficient features for perception tasks (Balestriero & LeCun, 2024). The approaches discussed above mainly address classification. Another stream of research, called object-centric learning, focuses on learning disentangled object representations. While early methods only worked on synthetic data (Burgess et al., 2019; Locatello et al., 2020) more recent approaches succeed on natural images (Zadaianchuk et al., 2024; Aydemir et al., 2023). Recently, a new method for evaluating such object-centric representations was proposed Didolkar et al. (2024). The main difference of our present work to object-centric methods is that we assume encode object instances are encoded implicitly in feature volumes whereas object-centric methods explicitly represent objects in their architecture (e. g. in slots). The seminal CLIP model (Radford et al., 2021) introduced another stream of research called vision-language models (VLMs), where the model is trained on aligning text-image pairs. Later, this training paradigm was simplified to use a sigmoid-based loss function (Zhai et al., 2023) instead of a softmax-based loss, making the method less dependent on the batch size. Recently, the role of data is investigated more closely in the context of vision-language models (Gadre et al., 2024; Xu et al., 2024; Fang et al., 2024).

Feature evaluation Evaluations on features predate the deep learning era in computer vision. Recently, there have been numerous attempts at characterizing and comparing common feature backbones but with different objectives. The works by Bonnen et al. (2024) and El Banani et al. (2024) focus on 3d shape understanding. Chen et al. (2024b) design a zero-shot benchmark for image encoders in contrastive vision-language pretraining setting and propose the ViTamin architecture. Goldblum et al. (2024) evaluate classification, instance segmentation, object detection and retrieval. Our work differs in focusing on dense prediction tasks without large heads enabling a more direct measurement of the feature quality. Further efforts to characterize vision backbones include the timm leaderboard (Wightman, 2019) for image classification, CLIP benchmark (LAION-AI, 2022) for vision-language models and CV-Bench for multimodal large language models (MLLMs; Tong et al., 2024).

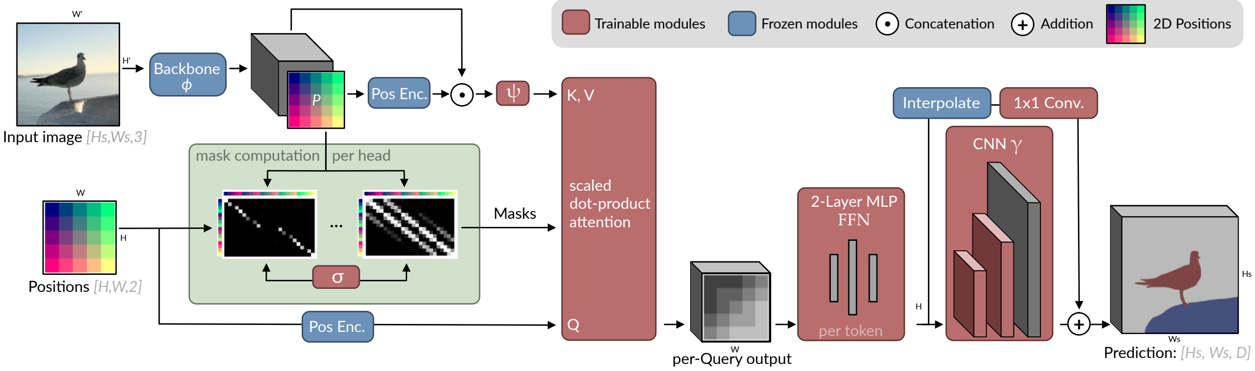

Figure 1: Dense attentive probing design: Queries in form of a coordinate grid attend to features extracted from an arbitrary backbone. The transformed queries upscaled by a small CNN to yield a task-specific output. Our dense readout decouples the predicted output size from the size of the feature volume. The masking strength is defined by a Gaussian function over the distance between corresponding points in the feature volume and the query grid, with the standard deviation being a learnable scalar parameter for each attention head.

Adapters and parameter-efficient fine-tuning Adapters are (often small) sub-networks that are trained to take generic features and *adapt* them to solve a specific task. For computer vision problems, many adapters were proposed that address whole-image classification (Chen et al., 2022a; Steitz & Roth, 2024). Also, the attentive probing (or attentional pooling) used in the CoCa (Yu et al., 2022) and V-JEPA evaluation (Bardes et al., 2023) can be considered a minimal adapter for whole image classification. These methods are not straightforward applicable for dense prediction. Based on the upsampling method FeatUp (Fu et al., 2024), linear evaluation can be applied in higher resolutions. To our knowledge this has not been done before but we compare to a baseline that uses this approach. The method proposed by Bhattacharjee et al. (2023) adapts to dense images but addresses multi-task learning, whereas our goal is to fine-tune on single tasks. Yang et al. (2024) mainly adapt the cluster-prediction paradigm for regression tasks. In both methods, the backbones are not entirely frozen.

In another research stream, learnable parameters are added inside the frozen backbone network, for instance in Adapter (Houlsby et al., 2019), low-rank adaptation (Hu et al., 2021, LoRA) and scaling-and-shifting (Lian et al., 2022). ViT-Adapter (Chen et al., 2022b) applies this paradigm for dense prediction tasks but builds on established task heads for segmentation (UperNet) and detection (Mask R-CNN and HTC++). Furthermore, the number of parameters introduced by the adapter depends on the backbone, ranging from 2.5M to 23.7M parameters. ConvAdapter (Chen et al., 2024a) propose an adapter specifically for convolutional networks. In case of dense prediction, their method uses standard task heads which require a large number of parameters. Zhong et al. (2024) apply low-rank adaptation to image segmentation tasks by introducing around 4M parameters, which is an order of magnitude more than our approach. For an in-depth review of adapters we refer to the survey of Yu et al. (2024).

3 Dense Attentive Probing

In this section, we introduce the **D**ense **A**ttentive **P**robing (DeAP) method (Fig. 1). It is designed to be a parameter and compute efficient readout that uses cross-attention to make dense predictions based on features from a frozen backbone. An arbitrary (frozen) image backbone ϕ receives an image $\mathbf{x}$ of size $(Hs, Ws, 3)$ and generates features of size (H, W, D') with s indicating the backbone's stride (the factor by which the backbone reduces the spatial resolution of its input). These features are concatenated (denoted by $[]$) with a non-learnable, sinusoidal positional encoding (PE) of the grid coordinates $\mathbf{p}$. The resulting activations are projected to the internal dimension D and flattened along the spatial dimensions (both by ψ), yielding $\mathbf{F}(\mathbf{x})$ of size (HW, D):

$$\mathbf{F}(\mathbf{x}) = \psi([PE(\mathbf{p}), \phi(\mathbf{x})]) \tag{1}$$

🔵 Supervised		
Supervised	ViT-B (86M)	ImageNet
SAM2	Sam B+	SA-1B

🟢 Self-supervised			
MoCoV3	J	ViT-B (86M)	ImageNet
MAE	R	ViT-B (86M)	ImageNet
Hiera	R	Hiera B+ (69M)	ImageNet
DINO	J	ViT-B (86M)	ImageNet
DINOv2	J & R	ViT-B (86M)	LVD-142M
DINOv2	J & R	ViT-L (304M)	LVD-142M

🔴 Vision-language		
CLIP	ViT-B (86M)	CLIP
MetaCLIP	ViT-B (86M)	CC-400M
SigLIP	ViT-B (86M)	Webli
SigLIP (SO)	ViT (414M)	Webli
SigLIP 512	ViT-B (86M)	Webli
Aim2 (300M)		many
ViTamin-L2 (333M)		DataComp-1

Table 2: Models, their backbone architectures (with parameters) and their training datasets. J and R denote joint-embedding and reconstruction self-supervised learning methods.

To generate a dense output, we use a cross-attention-based approach: Spatial queries $\mathbf{Q}$ of size $(H_Q W_Q, 8)$ attend to the feature volume $\mathbf{F}(\mathbf{x})$, with each query $\mathbf{Q}_j \in \mathbb{R}^8$ being responsible for generating the output of a local region. The queries $\mathbf{Q}$ are fixed, 8-dimensional positional sinusoidal encodings of the respective output positions (normally just a grid of positions) and thus have no learnable parameters.

To enable the model to account for feature locality, we modify the cross-attention to consider spatial proximity by adding $\mathbf{M}(\sigma)$. The computation per head h (out of H heads) is described by

$$\mathbf{T}^{(h)} = \text{softmax}\left(\frac{(\mathbf{Q}W_q^{(h)})(\mathbf{F}(\mathbf{x})W_k^{(h)})^T}{\sqrt{d_k}} + \mathbf{M}(\sigma)\right)(\mathbf{F}(\mathbf{x})W_v^{(h)}), \tag{2}$$

with $W_v^{(h)} \in \mathbb{R}^{D\times\frac{2D}{H}}$, $W_k^{(h)} \in \mathbb{R}^{D\times\frac{D}{H}}$ and $W_q \in \mathbb{R}^{8\times\frac{D}{H}}$. Since the attention considers all backbone tokens for each query, $\mathbf{M}(\sigma)$ has size $(H_Q W_Q, HW)$. Each element M_{ij} depends on the squared euclidean distance d_{ij}^2 between pixel i in the query and i in the feature volume through the function $M_{ij} = \frac{1}{\sigma\sqrt{2\pi}}\exp\left(-\frac{d_{ij}^2}{2\sigma^2}\right)$. Here, σ is a learned parameter per attention head. This means, the size of the region around each query position from which features are considered is adjustable by the model. The output is obtained by a concatenation of all H head outputs $\mathbf{T} = [\mathbf{T}^{(0)}, ...\mathbf{T}^{(H-1)}]$. Similar to the transformer layer, after this layer, each token $\mathbf{T}_i$ is processed independently by a 2-layer multi-layer perceptron (or feedforward layer) FFN, i.e. $\mathbf{T}'_i = \text{FFN}(\mathbf{T}_i)$. In the hidden dimension, the vectors are expanded by a factor of two.

Instead of using a separate query for every output pixel, regions of size 8 × 8 are processed jointly for efficiency reasons, i. e. each query $\mathbf{q}$ generates 64 pixels of the output. This is realized through a small CNN γ operating on the output of all tokens using transposed convolutions to increase spatial resolution. This CNN has three blocks, each composed of convolution and transposed convolution and ReLU non-linearity to increase resolution. These blocks are followed by a final convolution layer, a skip-connection enables efficient learning. The number of channels in this CNN is given by $D_{\text{CNN}} = \max(D/4, D_{\text{out}})$, with the number of output channels D_{out} being task-dependent. The tokens $\mathbf{T}'$ are re-arranged to the spatial shape (H,W,D) and jointly processed by the CNN γ, yielding the final output $\gamma(\mathbf{T}')$. Dense attentive probing consists of only a single masked cross attention layer.

Since the queries only receive a position as input and can attend to all features, the masked cross-attention architecture resembles implicit (or coordinate-based) networks (Mescheder et al., 2019; Park et al., 2019), especially PiFU (Saito et al., 2019) and PixelNerf (Yu et al., 2021) which combine implicit networks with feature volumes. The modification of the attention through a bias term is similar to GraphDINO (Weis et al., 2021).

4 Experiments

Choice of models and training datasets We select a broad range of feature backbones that encompass different training paradigms and were trained on different datasets (Tab. 2). This enables us to conduct controlled comparisons along several axes, for instance, datasets and pretraining task. In general, we differentiate between three broad classes of methods: supervised (Dosovitskiy et al., 2021; Ravi et al., 2024), self-supervised (Caron et al., 2021; He et al., 2022; Ryali et al., 2023), and vision-language (Radford et al., 2021; Xu et al., 2024; Fini et al., 2024; Chen et al., 2024b; Liu et al., 2022). The latter involves training

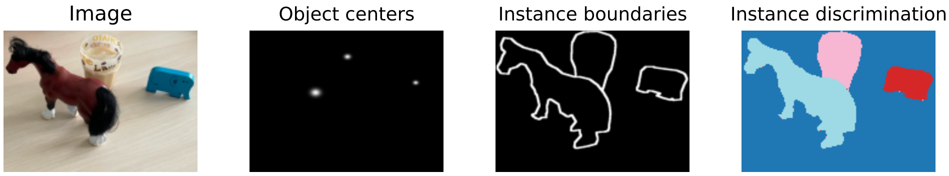

Figure 2: We use three tasks to probe instance awareness: object deteciton based on CenterNet, instance boundaries and instance discrimination.

on image-caption pairs often obtained from the internet, while self-supervised training operates on images only. Many models are trained on the ImageNet dataset (Russakovsky et al., 2014), but there are several exceptions: All SigLIP models are trained on the Webli dataset, a Google-internal dataset of 10 billion images with 12 billion multi-lingual text-image pairs. MetaCLIP uses a selection of the open LAION dataset (Schuhmann et al., 2021), CLIP is trained on the unpublished CLIP dataset by OpenAI. DINOv2 (Oquab et al., 2023) is trained on the LVD-142M, a Meta-internal dataset of 142M images which were deduplicated and curated to be similar to ImageNet-22k images. In our experiments, we use the DinoV2 version with registers Darcet et al. (2024). The data mix of Aim2 (Fini et al., 2024) contains DFN-2B, COYO, the proprietary HQITP dataset and synthetic data.

We decide to mainly focus on vision transformers as many approaches share the same architecture and checkpoints are available for a large number of training paradigms. To ensure comparability with other work and control for model architecture, we primarily use ViT-B/16 and similarly sized models in our experiments. We also include larger models in some cases to obtain an estimate of how much performance can be improved simply by scaling-up model size. Pretrained-weights are obtained from the timm package (Wightman, 2019) or the code repositories of the methods.

Experiment design We provide images in the native resolution of the respective backbones to prevent out-of-distribution input. For a fair comparison, we use images of size 224×224 as the basis for all tasks, if not indicated otherwise. When backbones require a higher native input resolution, we scale the 224×224 image to the respective resolution. For generating predictions, we make use of the capability of our model to decouple input and output resolution (see Sec. 3): The output size is fixed to 224×224 for all models (even for backbones with larger input image sizes), ensuring a fair comparison across models and limiting the advantage of large input sizes for the backbone. In the masked cross attention we use a dimension $D = 16$ and eight attention heads (i. e. two dimensions per head). Although the readout could attend to backbone activations at different layers we opt for using only the last layer, motivated by Li et al. (2022) who found that a simple feature pyramid using the last layer without top-down connections works best.

We pursue a straightforward approach to comparison: We freeze the backbone features, train the small readout in a supervised way and evaluate on a held-out test set. The rationale is that the low expressivity and capacity of the readout forces it to directly rely on the feature volume for making a dense prediction. This setup differs from conventional task heads (e. g. in object detection) which are able to perform more complex computations on the features. For example, using Faster R-CNN on top of a ResNet50 backbone adds around 18 million parameters to the model.[1]

4.1 Evaluation tasks

To characterize a broad spectrum of traits of the backbones we implement three task categories: probing instance awareness, local semantics and spatial understanding.

Instance awareness In this task we evaluate how well the features are able to disentangle individual instances, for example multiple apples in a bowl. In the field of object-centric machine learning (Burgess

[1] These calculations were obtained using the Faster R-CNN implementation in PyTorch vision (Paszke et al., 2019), in more detail FPN: 3.3M, RPN: 0.6M, ROI heads: 14.3M parameters.

et al., 2019) models are designed to disentangle instances. Here we assess the extent to which instance discrimination is already encoded in the features of different backbones as a result of their pretraining. We consider the following three notions of how instances can be encoded (Fig. 2):

- Boundaries: The objective is to outline individual objects in the image. We frame this problem as a binary segmentation and consequently use an output dimension $D_{\text{out}} = 1$ as well as the binary cross-entropy loss function on the output.
- CenterNet heatmaps: CenterNet (Zhou et al., 2019) is a detector that uses multiple heatmaps for detection. We use the center map and regress the bounding box sizes in a second heatmap. Both heatmaps are predicted using individual dense attentive probing readouts with $D_{\text{out}}^{\text{center}} = 1$ and $D_{\text{out}}^{\text{size}} = 2$. As proposed in the original paper, we obtain detections by finding 9-neighborhood maxima in the center heatmap and extracting the bounding box sizes at these locations. We further process the detections with non-maximum suppression. For comparison we report average precision of large objects.
- Instance discrimination: Another way to encode instances is to generate a latent space where features within instances are the same (or similar) while being different to all other instances. If this works perfectly, clustering the latent vectors of all pixels would yield instances. This task is sometimes also called coloring (Novotny et al., 2018). We train on only 8,000 sample images and treat every instance as an individual class (resulting in a around 60,000 classes). We first use dense attentive probing to map the features to a latent space (in our case $D_{\text{out}} = 32$). Then, a linear layer maps each local 32-dimensional feature to a probability over all instances in the dataset. Thus, the problem is essentially framed as semantic segmentation with 60,000 classes. This way, the latent features before the classification head learn to discriminate instances. For testing, we cluster these features obtained from unseen images. For clustering we use k-means and provide the ground-truth number of instances as well as a foreground mask. Then we compare the predicted foreground instances with the ground truth instance segmentation based on the adjusted rand index.

For these experiments, we use the COCO dataset (Lin et al., 2014), with the 5,000 images from the validation set being used for testing. For the instance discrimination task, we compute the ARI (adjusted rand index) test scores only on images with at least three large objects (resulting in a subset of 754 images). Note, these tasks do not involve classifying the instances into object categories, unlike typically done in instance segmentation (this is assessed below in “local semantics”).

Local semantics A natural choice for evaluating local semantics is a semantic segmentation task. Here we rely on two benchmarks: Pascal VOC 2012 (Everingham et al., 2015) and COCO Stuff (Caesar et al., 2018). The Pascal VOC 2012 encompasses a fairly small set of only 1,464 training images. For COCO Stuff, we train on 100,000 images. We account for the larger number of classes in COCO Stuff by setting the internal dimension of the CNN, D_{CNN}, to 32.

Spatial understanding To assess how well the features capture the 3D structure of the scene, we implement the well-known monocular depth estimation task: The models need to infer the depth (i. e. position along the z-axis) for every pixel of the visible scene based on the features provided by the backbone. We frame this as a depth map estimation problem, i. e. $D_{\text{out}} = 1$, relying on the NYUv2 dataset (Nathan Silberman & Fergus, 2012) for training and testing the depth estimation readout. We first scale the input images to a resolution of 216 × 288 and then to the native resolution of the backbones.

4.2 Baselines

We compare our method with other dense readout baselines. In all cases, the goal is to obtain a high resolution prediction based on a low-resolution feature volume $\mathbf{F}(\mathbf{x})$ of size (HW, D).

Bicubic Interpolation A natural choice to increase resolution is to interpolate in the feature volume F. This baseline replaces the cross-attention of our model by projection to a low dimension followed by a

					Inst. Disc.		Boundaries			Object Detection	
Cat.	Backbone	I	F	P	ARI ↑	P_{learn}	CE ↓	IoU ↑	P_{learn}	AP_{lg} ↑	P_{learn}
⚪	Random (untrained)	224	14	85.8	23.5	0.053	0.2544	0.8	0.043	0.0015	0.087
🔵	ImageNet	224	14	85.8	36.3	0.053	0.1787	16.0	0.043	0.1126	0.087
🔵	SAM V2 B+	1024	64	80.8	**50.0**	0.028	0.1394	30.2	0.018	0.1759	0.035
🟢	MoCo V3	224	14	85.8	41.8	0.053	0.1613	21.4	0.043	0.2078	0.087
🟢	Dino	224	28	85.9	41.5	0.053	0.1433	28.3	0.043	0.2344	0.087
🟢	Dino V2	518	37	86.6	46.8	0.053	**0.1297**	32.4	0.043	0.2666	0.087
🟢	Dino V2 (ViT-L)	518	37	304.4	46.4	0.066	0.1450	**33.8**	0.056	**0.2811**	0.112
🟢	MAE	224	14	85.8	49.8	0.053	0.1500	25.3	0.043	0.2662	0.087
🟢	Hiera B+	224	7	69.1	44.8	0.060	0.1717	17.1	0.050	0.2315	0.099
🔴	CLIP	224	14	85.9	39.5	0.053	0.1665	19.0	0.043	0.1391	0.087
🔴	CLIP (ViT-L)	336	24	303.6	40.6	0.066	0.1543	24.3	0.056	0.2005	0.112
🔴	MetaCLIP	224	14	85.9	38.4	0.053	0.1669	19.1	0.043	0.1667	0.087
🔴	SigLIP-224	224	14	85.8	37.8	0.053	0.1718	17.2	0.043	0.1677	0.087
🔴	SigLIP-384	384	24	86.1	39.1	0.053	0.1564	23.3	0.043	0.1836	0.087
🔴	SigLIP-512	512	32	86.5	38.9	0.053	0.1496	26.2	0.043	0.1910	0.087
🔴	SigLIP-SO	512	36	413.7	40.7	0.073	0.1496	25.7	0.062	0.1952	0.125
🔴	Aim2	336	24	309.6	40.0	0.066	0.1556	23.5	0.056	0.2011	0.112
🔴	ViTamin	384	24	333.0	42.1	0.066	0.1535	24.4	0.056	0.1648	0.112

Table 3: Instance awareness results in all three categories. I: image size. F: Size of feature volume. P and P_{learn}: Number of all and learnable parameters, respectively, in millions. Metrics: Adjusted rand index (ARI), cross-entropy (CE), intersection over union (IoU), average precision for large objects (AP_{lg}).

parameter-less bicubic interpolation operation. The rest of the baseline, i.e. the MLP and CNN, is identical to dense attentive pooling.

Transposed Convolutions A common component in many dense prediction architectures are transposed convolutions. This operation reverses downsampling by applying learned filters that generate spatially larger outputs. Analogous to the bicubic interpolation baseline, we replace the masked cross-attention with a single transposed convolution layer, while we use the same CNN for upsampling. A disadvantage of this baseline is that the output resolution depends on the feature volume resolution, e.g. increasing the feature volume from by a factor of two would also increase the output size by a factor of two.

FeatUp Recently, the FeatUp (Fu et al., 2024) method was introduced that up-scales a feature volume under consideration (conditional on) of the input image. We employ this image-aware upsampling technique to up-scale the feature volumes $\mathbf{F}(\mathbf{x})$ and add a linear projection that maps to the task-defined output space.

5 Results

5.1 Comparison on backbone performance

A main goal of our work is to provide a fair comparison of different backbones regarding instance discrimination, local semantics and spatial understanding. We aim to highlight strengths and weaknesses of each backbone. Results (Tab. 3) indicate that DINOv2 has the best instance awareness. The backbone of SAM2 does not outperform other backbones considerably, this is surprising given that it was trained on instance discrimination. It suggests SAM2's mask decoder (which we did not use here) is a crucial component. Interestingly, self-supervised methods consistently outperform vision-language models in instance awareness (Fig. 3). Despite following the same reconstruction-based training, MAE performs better than Hiera in instance awareness while the opposite is true for semantic segmentation and depth. Among the vision-language models CLIP, MetaCLIP and SigLIP we did not find meaningful differences.

					Pascal VOC2012			COCO Stuff			Depth	
Cat.	Backbone	I	F	P	CE ↓	IoU ↑	P_{learn}	CE ↓	IoU ↑	P_{learn}	RMSE ↓	P_{learn}
⚪	Random (untrained)	224	14	85.8	2.7364	3.5	0.047	5.0889	0.1	0.088	1.253	0.043
🔵	ImageNet	224	14	85.8	0.3924	61.0	0.047	1.5140	28.1	0.088	0.769	0.043
🔵	SAM V2 B+	1024	64	80.8	0.6005	33.9	0.022	2.0630	12.6	0.062	0.754	0.018
🟢	MoCo V3	224	14	85.8	0.3274	63.3	0.047	1.4417	27.3	0.088	0.680	0.043
🟢	Dino	224	28	85.9	0.3397	63.8	0.047	1.3423	30.2	0.088	0.672	0.043
🟢	Dino V2	518	37	86.6	0.1259	83.4	0.047	1.0468	42.8	0.088	0.449	0.043
🟢	Dino V2 (ViT-L)	518	37	304.4	0.1173	**85.3**	0.060	**1.0347**	**43.8**	0.101	**0.429**	0.056
🟢	MAE	224	14	85.8	0.3471	60.2	0.047	1.4822	25.3	0.088	0.631	0.043
🟢	Hiera B+	224	7	69.1	0.3344	61.5	0.054	1.5007	25.7	0.094	0.556	0.050
🔴	CLIP	224	14	85.9	0.2710	68.3	0.047	1.3367	32.0	0.088	0.646	0.043
🔴	CLIP (ViT-L)	336	24	303.6	0.2228	74.9	0.060	1.2623	35.5	0.101	0.590	0.056
🔴	MetaCLIP	224	14	85.9	0.2580	69.4	0.047	1.3427	31.9	0.088	0.648	0.043
🔴	SigLIP-224	224	14	85.8	0.2898	67.1	0.047	1.3355	33.0	0.088	0.674	0.043
🔴	SigLIP-384	384	24	86.1	0.2137	73.9	0.047	1.2565	35.4	0.088	0.627	0.043
🔴	SigLIP-512	512	32	86.5	0.2203	75.7	0.047	1.2435	36.3	0.088	0.620	0.043
🔴	SigLIP-SO	512	36	413.7	0.1908	77.9	0.066	1.1737	38.8	0.107	0.556	0.062
🔴	Aim2	336	24	309.6	0.2244	75.2	0.060	1.1957	37.5	0.101	0.562	0.056
🔴	ViTamin	384	24	333.0	0.1843	77.9	0.060	1.1757	37.9	0.101	0.541	0.056

Table 4: Local semantics results on Pascal and COCO Stuff as well as spatial understanding on NYUv2 (right). I: image size. F: Size of feature volume. P and P_{learn}: Number of all and learnable parameters, respectively, in millions. Metrics: cross-entropy (CE), intersection over union (IoU), mean-squared error (MSE)

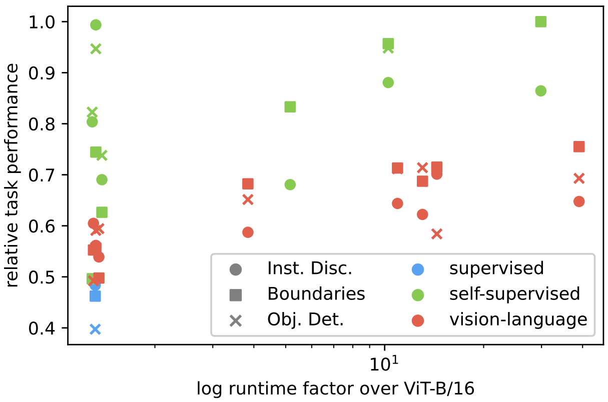

Figure 3: Self-supervised methods consistently outperform vision-language methods across all three instance awareness tasks. To show all tasks in a single figure, we report the relative task performance, for which the original scores were mapped linearly into the interval [0,1].

Vision-language models tend to show stronger local semantics (Tab. 4). While all vision-language models with 224px input size show similar performance, the larger versions of SigLIP (i. e. 384px and the shape optimized SO version) perform better but at a higher cost. Also in this evaluation, DINOv2 achieves the best scores. All things considered, possibly the most striking finding is the dominance of DINOv2. While one might argue that this is due to large image sizes and feature volumes, the mediocre performance of SigLIP-512, Hiera-B+ and (partly) SAM V2 show that it cannot be the only factor.

The evaluation on spatial understanding shows mixed results. Larger backbones tend to perform better, with the exception of Hiera-B+. Again, DINOv2 performs best, in this case by a large margin.

	Object Detection: $\mathrm{AP}_{\mathrm{lg}}$ ↑			Semantic Segmentation: IoU ↑			Depth: RMSE ↓		
names	Dino V2	SigLIP-384	MAE	Dino V2	SigLIP-384	MAE	Dino V2	SigLIP-384	MAE
base	**0.2666**	0.1836	0.2662	83.4	**73.9**	**60.2**	**0.4488**	**0.6274**	0.6307
8-dim	0.2448	0.1722	0.2529	82.8	70.4	55.6	0.4697	0.6626	0.6111
no-sigma	0.1791	0.1196	0.1712	50.5	38.1	35.1	0.6660	0.8362	0.7360
no-indiv-sigma	0.2409	0.1520	0.2429	**83.8**	70.2	58.6	0.4797	0.6552	0.6657
only-mask	0.2606	**0.1934**	**0.2673**	82.7	73.0	60.0	0.4712	0.6378	**0.5887**

Table 5: Ablation. The full model is the variant we use in all other experiments.

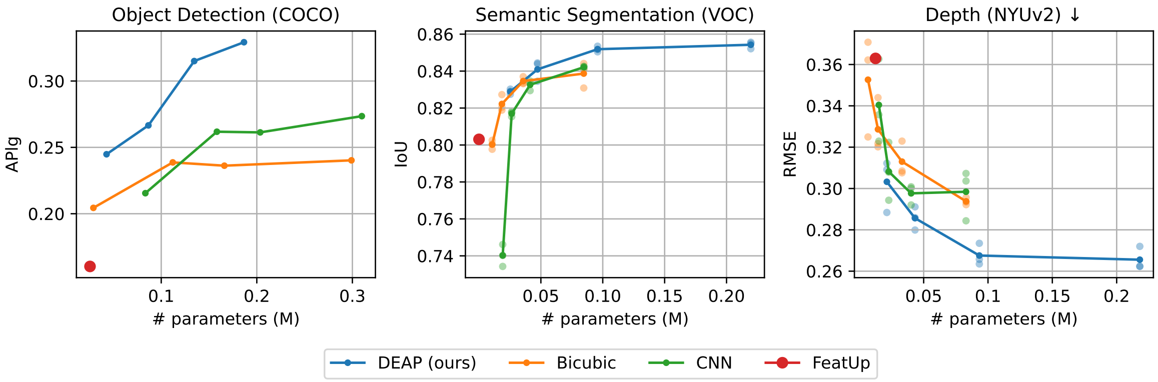

Figure 4: Dense attentive probing outperforms CNN, interpolation and FeatUp-based readouts on DinoV2 features* (instances: ↑ = better; semantic segmentation ↑; depth ↓). For semantic segmentation and depth the lines represent averages over three runs.

* For the largest bicubic run on object detection we report the best score of three runs due to training instability.

5.2 Readout design

Ablation We next explore design choices of our dense attentive probing readout by varying relevant hyperparameters (Tab. 5). The introduction of the σ parameter and its adaptivity per head is crucial for good performance. This suggests that information is organized at different levels of locality in the feature volumes.

Alternative readout architectures Next, we ask whether alternative architectures (introduced in Sec. 4.2) with a similar parameter budget could perform competitively to dense attentive probing. We find the latter to be more parameter efficient and to achieve better scores than the baselines across all three tasks (Fig. 4). An additional advantage of our method over CNNs is decoupling input and output resolution, in a CNN, a larger feature volume size would result in a larger output. FeatUp is highly parameter efficient but has high memory demands and requires long computation times (factor 4 compared to dense attentive probing).

Variable output size To ensure a fair comparison, the readout size is fixed in the previous experiments. However, it is possible to generate outputs at an arbitrary resolution, because the adapter takes positions as inputs (similar to implicit neural fields). Instead of using the standard query position grid for a 224 px output, we can sample different query coordinates at test time (Fig. 5). The results show that our method generalizes well to higher resolutions, with some details being resolved better at higher resolutions (see for example the head of the horse and the chair).

5.3 DeAP performance correlates with performance of specialized models on downstream tasks

To assess the reliability of our findings, we consider previous work in object-centric representation learning and a classification-based evaluation (Fig. 6). Object-centric learning shares the goal of disentangling in-

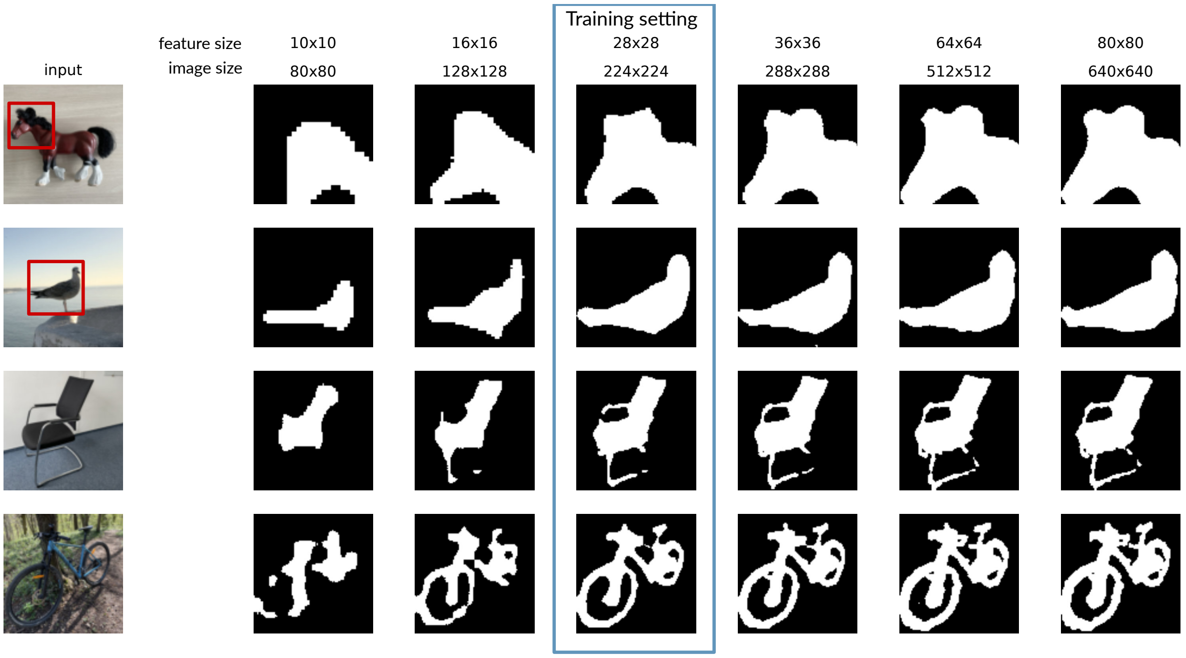

Figure 5: After training, our readout can be queried to output different resolutions from the same backbone. Here we use a DINOv2 backbone trained on Pascal VOC.

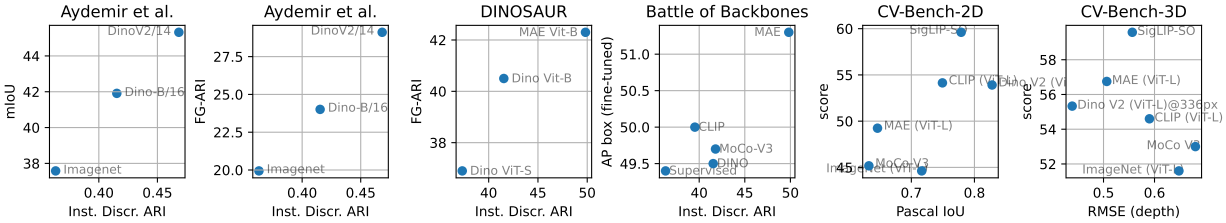

Figure 6: We compare our results with reported scores on two object-centric learning methods (three leftmost panels) (Aydemir et al., 2023; Seitzer et al., 2023)), fine-tuned Faster R-CNN backbones (Goldblum et al., 2024) and CV-Bench 2D and 3D (Tong et al., 2024).

stances but achieves this through specific model architectures, whereas we evaluate model-agnostic features for instance-specific signals. In these experiments we compare how well the object-centric scores match our instance discrimination scores for the same backbones. Relating to the instance clustering foreground adjusted rand-index, which indicates how well instances are disentangled, by Aydemir et al. (2023) we find an almost linear relationship between their and our instance discrimination scores. Also comparing to DINOSAUR (Seitzer et al., 2023), we find an almost linear relationship over three backbones (Dino ViT-S, Dino ViT-B and MAE).

The evaluation of Goldblum et al. (2024) shares the goal of characterizing current backbones with our work but put more emphasis on out-of-distribution and backbone architecture. While a positive relation between their and our scores is recognizable, the trend is less pronounced that in previous experiments. CV-Bench-2D and CV-Bench-3D (Tong et al., 2024) were recently introduced to assess the capabilities of multi-modal language models and provide scores for several vision backbones. While the variance is larger, we can see the same trends again: For the semantic score of CV-Bench-2D there is a positive correlation with our semantic segmentation measurements while there is a negative correlation between the spatial scores of CV-Bench-3D and our monocular depth prediction errors (where lower is better). In summary, our method can be used to obtain similar insights on relative backbone performance as more expensive and slower evaluation methods.

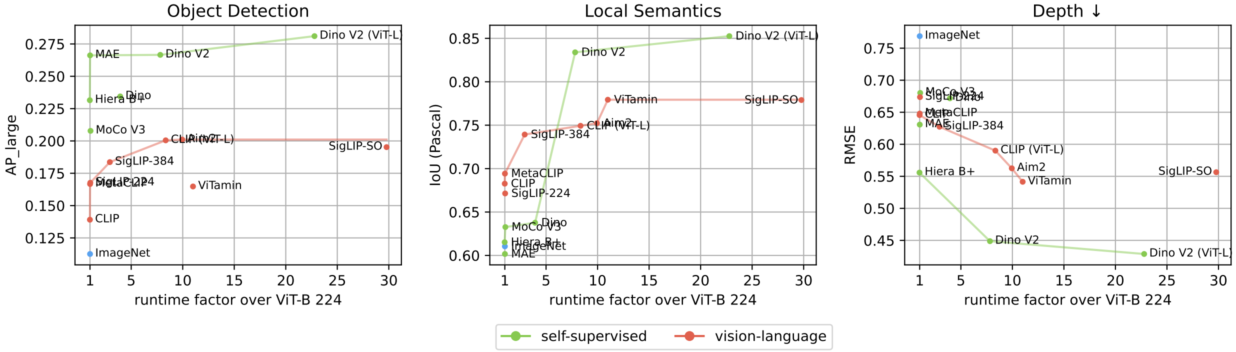

Figure 7: Inference speed over performance on three tasks, relative to the ViT-B/16 with 224px image input (the fastest and most frequent architecture in our evaluation). The lines depict the pareto fronts.

5.4 Speed-performance tradeoff

For practitioners, runtime limitations can constrain the selection of viable backbones, for example when a minimal latency is required. Therefore, we explore the relationship between task performance and inference speed in more detail (Fig. 7). We measure the time to run ten batches with eight samples each in inference mode (i. e. without gradient computation). The fastest model in our evaluation set is the ViT-B/16 at a resolution of 224 pixels. Therefore we indicate the factor by which the runtime is extended with respect to this model. For example, the slowest model, SigLIP-SO takes around 30 times as long as the ViT-B/16 reference model. The results show that, depending on the task, there are fast models that achieve good performance, namely MAE for object detection and Hiera B+ for depth prediction. For local semantics we find a stronger dependency between model sizes and performance, i.e. larger models are required for good performance.

6 Conclusion

In this work we proposed dense attentive probing, a fast and parameter-efficient readout for evaluating the representational expressiveness of trained backbones on dense prediction. For example, our standard training for a readout on a ViT-B/16 224 pixel backbone on Pascal VOC adds less than 70,000 parameters and trains in less than 16 minutes (using a single Nvidia RXT2080 GPU). We used dense attentive probing to systematically analyze common vision backbones with respect to the three complementary aspects: instance awareness, depth and local semantics. Our results suggest that the backbones of the DINOv2 family are highly capable. It is the best ViT-B model across all experiments. Using DINOv2 with a ViT-L backbone improves performance further but at the cost of a three times longer runtime. Classic supervised pretraining on ImageNet results in fairly poor performance. Excluding DINOv2, we identified the trend that local semantics is better captured by language-vision models while reconstruction-based self-supervised learning leads to features with better instance awareness.

For practitioners, DINOv2 is a natural choice if enough compute is available. For compute-constrained cases, the decision is more complex: MAE is recommendable for instance related tasks while CLIP-based models (e.g. MetaCLIP) show good local semantics. For spatial understanding Hiera-B represents a good trade-off between performance and inference speed. We plan to retain an online leaderboard where new backbones can easily be incorporated to help tracking future progress of dense prediction performance.

7 Limitations

While we use a fairly small readout (in terms of parameters) that can adapt to multiple locality scales in the features through the learnable masks, even this readout has inductive biases and can favor certain backbones such that results might get distorted. We deliberately opted for a small readout to directly measure the feature performance. Consequently, this limits the degree to which the features can be re-combined and processed. In fact, this could be the reason why the SAM2 backbone performs comparably poorly in instance awareness in our hands: it does rely on a relatively "heavy" decoder. A more direct comparison to object-centric approaches would be interesting, but is challenging as these approaches explicitly encode objects (e. g. in attention slots) which can be compared to ground truth.

The current selection of tasks we evaluated is limited to three broad categories and a few instances of those. Adding additional task categories (e. g. as in Taskonomy (Zamir et al., 2018)) would be desirable in the future for a broader characterization of backbones.

References

Görkay Aydemir, Weidi Xie, and Fatma Guney. Self-supervised object-centric learning for videos. *Advances in Neural Information Processing Systems*, 36:32879–32899, 2023.

Randall Balestriero and Yann LeCun. Learning by reconstruction produces uninformative features for perception. *arXiv preprint arXiv:2402.11337*, 2024.

Randall Balestriero, Mark Ibrahim, Vlad Sobal, Ari S. Morcos, Shashank Shekhar, Tom Goldstein, Florian Bordes, Adrien Bardes, Grégoire Mialon, Yuandong Tian, Avi Schwarzschild, Andrew Gordon Wilson, Jonas Geiping, Quentin Garrido, Pierre Fernandez, Amir Bar, Hamed Pirsiavash, Yann LeCun, and Micah Goldblum. A cookbook of self-supervised learning. *ArXiv*, abs/2304.12210, 2023.

Adrien Bardes, Jean Ponce, and Yann LeCun. Vicreg: Variance-invariance-covariance regularization for self-supervised learning. *arXiv preprint arXiv:2105.04906*, 2021.

Adrien Bardes, Quentin Garrido, Jean Ponce, Xinlei Chen, Michael Rabbat, Yann LeCun, Mido Assran, and Nicolas Ballas. V-jepa: Latent video prediction for visual representation learning. 2023.

Deblina Bhattacharjee, Sabine Süsstrunk, and Mathieu Salzmann. Vision transformer adapters for generalizable multitask learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 19015–19026, 2023.

Tyler Bonnen, Stephanie Fu, Yutong Bai, Thomas O'Connell, Yoni Friedman, Nancy Kanwisher, Joshua B Tenenbaum, and Alexei A Efros. Evaluating multiview object consistency in humans and image models. *arXiv preprint arXiv:2409.05862*, 2024.

Christopher P. Burgess, Loïc Matthey, Nicholas Watters, Rishabh Kabra, Irina Higgins, Matthew M. Botvinick, and Alexander Lerchner. Monet: Unsupervised scene decomposition and representation. *ArXiv*, abs/1901.11390, 2019.

Holger Caesar, Jasper Uijlings, and Vittorio Ferrari. Coco-stuff: Thing and stuff classes in context. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1209–1218, 2018.

Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 9650–9660, 2021.

Hao Chen, Ran Tao, Han Zhang, Yidong Wang, Xiang Li, Wei Ye, Jindong Wang, Guosheng Hu, and Marios Savvides. Conv-adapter: Exploring parameter efficient transfer learning for convnets. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 1551–1561, 2024a.

Jieneng Chen, Qihang Yu, Xiaohui Shen, Alan Yuille, and Liang-Chieh Chen. Vitamin: Designing scalable vision models in the vision-language era. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 12954–12966, 2024b.

Shoufa Chen, Chongjian Ge, Zhan Tong, Jiangliu Wang, Yibing Song, Jue Wang, and Ping Luo. Adaptformer: Adapting vision transformers for scalable visual recognition. *Advances in Neural Information Processing Systems*, 35:16664–16678, 2022a.

Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pp. 1597–1607. PMLR, 2020.

Xinlei Chen, Saining Xie, and Kaiming He. An empirical study of training self-supervised vision transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 9640–9649, 2021.

Zhe Chen, Yuchen Duan, Wenhai Wang, Junjun He, Tong Lu, Jifeng Dai, and Yu Qiao. Vision transformer adapter for dense predictions. *arXiv preprint arXiv:2205.08534*, 2022b.

Timothée Darcet, Maxime Oquab, Julien Mairal, and Piotr Bojanowski. Vision transformers need registers. *International Conference on Learning Representations (ICLR)*, 2024.

Aniket Didolkar, Andrii Zadaianchuk, Anirudh Goyal, Mike Mozer, Yoshua Bengio, Georg Martius, and Maximilian Seitzer. Zero-shot object-centric representation learning. *arXiv preprint arXiv:2408.09162*, 2024.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *ICLR*, 2021.

Mohamed El Banani, Amit Raj, Kevis-Kokitsi Maninis, Abhishek Kar, Yuanzhen Li, Michael Rubinstein, Deqing Sun, Leonidas Guibas, Justin Johnson, and Varun Jampani. Probing the 3d awareness of visual foundation models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 21795–21806, 2024.

Mark Everingham, SM Ali Eslami, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The pascal visual object classes challenge: A retrospective. *International journal of computer vision*, 111:98–136, 2015.

Alex Fang, Albin Madappally Jose, Amit Jain, Ludwig Schmidt, Alexander Toshev, and Vaishaal Shankar. Data filtering networks. *International Conference on Learning Representations (ICLR)*, 2024.

Enrico Fini, Mustafa Shukor, Xiujun Li, Philipp Dufter, Michal Klein, David Haldimann, Sai Aitharaju, Victor Guilherme Turrisi da Costa, Louis Béthune, Zhe Gan, Alexander T Toshev, Marcin Eichner, Moin Nabi, Yinfei Yang, Joshua M. Susskind, and Alaaeldin El-Nouby. Multimodal autoregressive pre-training of large vision encoders, 2024.

Stephanie Fu, Mark Hamilton, Laura E. Brandt, Axel Feldmann, Zhoutong Zhang, and William T. Freeman. Featup: A model-agnostic framework for features at any resolution. In *International Conference on Learning Representations (ICLR)*, 2024.

Samir Yitzhak Gadre, Gabriel Ilharco, Alex Fang, Jonathan Hayase, Georgios Smyrnis, Thao Nguyen, Ryan Marten, Mitchell Wortsman, Dhruba Ghosh, Jieyu Zhang, et al. Datacomp: In search of the next generation of multimodal datasets. *Advances in Neural Information Processing Systems*, 36, 2024.

Micah Goldblum, Hossein Souri, Renkun Ni, Manli Shu, Viraj Prabhu, Gowthami Somepalli, Prithvijit Chattopadhyay, Mark Ibrahim, Adrien Bardes, Judy Hoffman, et al. Battle of the backbones: A large-scale comparison of pretrained models across computer vision tasks. *Advances in Neural Information Processing Systems*, 36, 2024.

Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 16000–16009, 2022.

Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin De Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. Parameter-efficient transfer learning for nlp. In *International conference on machine learning*, pp. 2790–2799. PMLR, 2019.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*, 2021.

Menglin Jia, Luming Tang, Bor-Chun Chen, Claire Cardie, Serge Belongie, Bharath Hariharan, and Ser-Nam Lim. Visual prompt tuning. In *European Conference on Computer Vision*, pp. 709–727. Springer, 2022.

LAION-AI. Clip benchmark. https://github.com/LAION-AI/CLIP_benchmark, 2022.

Yanghao Li, Hanzi Mao, Ross Girshick, and Kaiming He. Exploring plain vision transformer backbones for object detection. In *European conference on computer vision*, pp. 280–296. Springer, 2022.

Dongze Lian, Daquan Zhou, Jiashi Feng, and Xinchao Wang. Scaling & shifting your features: A new baseline for efficient model tuning. *Advances in Neural Information Processing Systems*, 35:109–123, 2022.

Tsung-Yi Lin, Michael Maire, Serge J. Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. Microsoft coco: Common objects in context. In *European Conference on Computer Vision*, 2014.

Zhuang Liu, Hanzi Mao, Chao-Yuan Wu, Christoph Feichtenhofer, Trevor Darrell, and Saining Xie. A convnet for the 2020s. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 11976–11986, 2022.

Francesco Locatello, Dirk Weissenborn, Thomas Unterthiner, Aravindh Mahendran, Georg Heigold, Jakob Uszkoreit, Alexey Dosovitskiy, and Thomas Kipf. Object-centric learning with slot attention. *Advances in neural information processing systems*, 33:11525–11538, 2020.

Lars Mescheder, Michael Oechsle, Michael Niemeyer, Sebastian Nowozin, and Andreas Geiger. Occupancy networks: Learning 3d reconstruction in function space. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 4460–4470, 2019.

Pushmeet Kohli Nathan Silberman, Derek Hoiem and Rob Fergus. Indoor segmentation and support inference from rgbd images. In *ECCV*, 2012.

David Novotny, Samuel Albanie, Diane Larlus, and Andrea Vedaldi. Semi-convolutional operators for instance segmentation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 86–102, 2018.

Maxime Oquab, Timoth'ee Darcet, Th'eo Moutakanni, Huy Q. Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, Mahmoud Assran, Nicolas Ballas, Wojciech Galuba, Russ Howes, Po-Yao Huang, Shang-Wen Li, Ishan Misra, Michael G. Rabbat, Vasu Sharma, Gabriel Synnaeve, Huijiao Xu, Hervé Jégou, Julien Mairal, Patrick Labatut, Armand Joulin, and Piotr Bojanowski. Dinov2: Learning robust visual features without supervision. *ArXiv*, abs/2304.07193, 2023.

Jeong Joon Park, Peter Florence, Julian Straub, Richard Newcombe, and Steven Lovegrove. Deepsdf: Learning continuous signed distance functions for shape representation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 165–174, 2019.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32, 2019.

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 *International Conference on Machine Learning*, 2021.

Nikhila Ravi, Valentin Gabeur, Yuan-Ting Hu, Ronghang Hu, Chaitanya Ryali, Tengyu Ma, Haitham Khedr, Roman Rädle, Chloe Rolland, Laura Gustafson, et al. Sam 2: Segment anything in images and videos. *arXiv preprint arXiv:2408.00714*, 2024.

Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael S. Bernstein, Alexander C. Berg, and Li Fei-Fei. Imagenet large scale visual recognition challenge. *International Journal of Computer Vision*, 115:211–252, 2014.

Chaitanya Ryali, Yuan-Ting Hu, Daniel Bolya, Chen Wei, Haoqi Fan, Po-Yao Huang, Vaibhav Aggarwal, Arkabandhu Chowdhury, Omid Poursaeed, Judy Hoffman, et al. Hiera: A hierarchical vision transformer without the bells-and-whistles. *International Conference on Machine Learning (ICML)*, 2023.

Shunsuke Saito, Zeng Huang, Ryota Natsume, Shigeo Morishima, Angjoo Kanazawa, and Hao Li. Pifu: Pixel-aligned implicit function for high-resolution clothed human digitization. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 2304–2314, 2019.

Christoph Schuhmann, Richard Vencu, Romain Beaumont, Robert Kaczmarczyk, Clayton Mullis, Aarush Katta, Theo Coombes, Jenia Jitsev, and Aran Komatsuzaki. Laion-400m: Open dataset of clip-filtered 400 million image-text pairs. *arXiv preprint arXiv:2111.02114*, 2021.

Maximilian Seitzer, Max Horn, Andrii Zadaianchuk, Dominik Zietlow, Tianjun Xiao, Carl-Johann Simon-Gabriel, Tong He, Zheng Zhang, Bernhard Schölkopf, Thomas Brox, et al. Bridging the gap to real-world object-centric learning. 2023.

Jan-Martin O Steitz and Stefan Roth. Adapters strike back. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 23449–23459, 2024.

Shengbang Tong, Ellis Brown, Penghao Wu, Sanghyun Woo, Manoj Middepogu, Sai Charitha Akula, Jihan Yang, Shusheng Yang, Adithya Iyer, Xichen Pan, et al. Cambrian-1: A fully open, vision-centric exploration of multimodal llms. *arXiv preprint arXiv:2406.16860*, 2024.

Marissa A. Weis, Laura Pede, Timo Lüddecke, and Alexander S. Ecker. Self-supervised representation learning of neuronal morphologies. *ArXiv*, abs/2112.12482, 2021.

Ross Wightman. Pytorch image models. https://github.com/rwightman/pytorch-image-models, 2019.

Hu Xu, Saining Xie, Xiaoqing Ellen Tan, Po-Yao Huang, Russell Howes, Vasu Sharma, Shang-Wen Li, Gargi Ghosh, Luke Zettlemoyer, and Christoph Feichtenhofer. Demystifying clip data. 2024.

Xuan Yang, Liangzhe Yuan, Kimberly Wilber, Astuti Sharma, Xiuye Gu, Siyuan Qiao, Stephanie Debats, Huisheng Wang, Hartwig Adam, Mikhail Sirotenko, et al. Polymax: General dense prediction with mask transformer. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 1050–1061, 2024.

Alex Yu, Vickie Ye, Matthew Tancik, and Angjoo Kanazawa. pixelnerf: Neural radiance fields from one or few images. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 4578–4587, 2021.

Bruce XB Yu, Jianlong Chang, Haixin Wang, Lingbo Liu, Shijie Wang, Zhiyu Wang, Junfan Lin, Lingxi Xie, Haojie Li, Zhouchen Lin, et al. Visual tuning. *ACM Computing Surveys*, 56(12):1–38, 2024.

Jiahui Yu, Zirui Wang, Vijay Vasudevan, Legg Yeung, Mojtaba Seyedhosseini, and Yonghui Wu. Coca: Contrastive captioners are image-text foundation models. 2022.

Andrii Zadaianchuk, Maximilian Seitzer, and Georg Martius. Object-centric learning for real-world videos by predicting temporal feature similarities. *Advances in Neural Information Processing Systems*, 36, 2024.

Amir R Zamir, Alexander Sax, William Shen, Leonidas J Guibas, Jitendra Malik, and Silvio Savarese. Taskonomy: Disentangling task transfer learning. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 3712–3722, 2018.

Xiaohua Zhai, Joan Puigcerver, Alexander Kolesnikov, Pierre Ruyssen, Carlos Riquelme, Mario Lucic, Josip Djolonga, Andre Susano Pinto, Maxim Neumann, Alexey Dosovitskiy, et al. A large-scale study of representation learning with the visual task adaptation benchmark. *arXiv preprint arXiv:1910.04867*, 2019.

Xiaohua Zhai, Basil Mustafa, Alexander Kolesnikov, and Lucas Beyer. Sigmoid loss for language image pre-training. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 11975–11986, 2023.

Zihan Zhong, Zhiqiang Tang, Tong He, Haoyang Fang, and Chun Yuan. Convolution meets lora: Parameter efficient finetuning for segment anything model. *arXiv preprint arXiv:2401.17868*, 2024.

Jinghao Zhou, Chen Wei, Huiyu Wang, Wei Shen, Cihang Xie, Alan Yuille, and Tao Kong. ibot: Image bert pre-training with online tokenizer. *arXiv preprint arXiv:2111.07832*, 2021.

Xingyi Zhou, Dequan Wang, and Philipp Krähenbühl. Objects as points. *arXiv preprint arXiv:1904.07850*, 2019.