A. Appendix

A.1. Pre-trained models

Note that for InsDis we use the model weights provided by the PyContrast GitHub repository which report higher ImageNet top-1 accuracy than originally reported (59.5 vs 54.0). As weights are not available for PIRL we likewise, take the ones provided by PyContrast which reports a slightly lower ImageNet accuracy of 61.7 (compared to 63.6). All other models are obtained from the original authors. We use the PyTorch framework in our code and therefore convert some of the models from their TensorFlow checkpoints. For most models we normalise the inputs by the mean and standard deviation on the ILSVRC12 train set, apart from SimCLR-v1/v2 which do not expect normalised inputs.

A.2. Many-shot evaluation details

The top-1 accuracy metric is reported on Food-101, CIFAR-10, CIFAR-100, SUN397, Stanford Cars, and DTD, mean per-class accuracy on FGVC Aircraft, Oxford-IIIT Pets, Caltech-101, and Oxford 102 Flowers and the 11-point mAP metric from [14] on Pascal VOC 2007. On Caltech-101 we randomly select 30 images per class to form the training set and we test on the rest. We use the first train/test split defined in DTD and SUN397. On FGVC Aircraft, Pascal VOC2007, DTD, and Oxford 102 Flowers we use the validation sets defined by the authors, and on the other datasets we randomly select 20% of the training set to form the validation set. The optimal hyperparameters were selected on the validation set, after which we retrained the model on all training and validation images. Finally, the accuracy is computed on the test set.

Linear We fit a multinomial logistic regression model on the extracted features of dimensionality 2048 from the frozen backbones. No augmentation was used and the images were resized to 224 pixels along the shorter side using bicubic resampling, followed by a center crop of 224 $\times$ 224. We select the $\ell 2$ regularisation constant on the validation set over 45 logarithmically spaced values between 10^{-6} and 10^{5}. The model is optimised using L-BFGS [36] on the softmax cross-entropy objective. As Pascal VOC2007 is a multi-label task, we fit one binary classifier for each class.

Finetuning We finetune the models following the protocol of [6] with minor modifications. We train for 5000 steps with a batch size of 64. The optimiser is SGD with Nesterov momentum and a momentum parameter of 0.9. The learning rate follows a cosine annealing schedule without restarts, and the initial learning rate is chosen from a grid of 4 logarithmically spaced values between 0.0001 and 0.1. The weight decay is similarly chosen from a grid of 4 logarithmically spaced values between 10^{-6} and 10^{-3}, along with no weight decay. These weight decay values are divided by the learning rate. We select the data augmentation from: random crop with resize and flip, or simply a center crop.

A.3. Few-shot evaluation details

For each few-shot learning episode we sample images from the combined sets of train, validation and test images. We fit a nearest centroid classifier on the extracted features of dimensionality 2048 from the frozen backbones. No augmentation was used and the images were resized to 224 pixels along the shorter side using bicubic resampling, followed by a center crop of 224 $\times$ 224. The fitted model is evaluated using 15 query images in each episode and the reported accuracies and errors are computed from 600 total episodes. In addition to the 20-shot results presented in the paper, we also report 5-shot and 50-shot results in Tables 6, 7 and 8. Note that in the original CD-FSL benchmark [19], models are only allowed to pre-train on mini-ImageNet and not the full version, so our results are not comparable to those of the original authors.

A.4. Detection evaluation details

We train the detectors on the VOC 2007 and 2012 trainval sets, and test on VOC 2007 test. When evaluating frozen backbones, we freeze all but the final residual block of the ResNets. In the full finetuning setup, we let the entire network be trainable. We extract features from the backbone using a Feature Pyramid Network [35] architecture and attach a Faster R-CNN [46] detector head to produce predictions. During training, the images are resized so the shorter side is one of [480, 512, 544, 576, 608, 640, 672, 704, 736, 768, 800] and during testing to 800 pixels. The models are trained for 144k iterations with a 100 iteration warm-up to an initial learning rate of 0.0025 which is decayed by a factor of 10 at iterations 96k and 128k. The batch size is 2 and we used a single GPU per model. Any other details of training uses the default values of the detectron2 [57] framework.

A.5. Surface normal estimation evaluation details

We use the implementation of [16], which is based on [70]. Each model is trained for 150 epochs, with the full backbone frozen. We use stochastic gradient descent with a momentum of 0.9, batch size of 4 and set the learning rate according to $(1 - \frac{t}{T})^{0.9}$, where t is the current epoch and T is the total number of epochs.

A.6. Semantic segmentation evaluation details

Models are trained (without freezing any layers) using stochastic gradient descent with an initial learning rate of 0.02, which is decayed by a factor of 0.9 every 500 iterations, and a constant momentum rate of 0.9. All models are trained with a batch size of two for 150k iterations in total.

A.7. Computing correlations

At many points in this work we analyse the statistical relationships between different results. This includes the correlation coefficients in Figs. 1, 2, 3, 5 and 6, those reported in the text and more summarised in Table 9. In order to

Table 6. 5-way 5-shot transfer on the Kornblith datasets. We report the average accuracy and 95% confidence interval over 600 test episodes. Results style: **best**, second best.

	Aircraft	Caltech101	Cars	CIFAR10	CIFAR100	DTD	Flowers	Food	Pets	SUN397
InsDis	42.59 ± 0.90	83.31 ± 0.65	46.42 ± 0.72	62.64 ± 0.64	68.06 ± 0.76	73.74 ± 0.67	89.55 ± 0.53	61.50 ± 0.75	73.21 ± 0.68	84.77 ± 0.60
MoCo-v1	42.74 ± 0.94	86.98 ± 0.57	44.63 ± 0.69	60.07 ± 0.64	66.10 ± 0.79	74.98 ± 0.70	89.13 ± 0.53	62.45 ± 0.78	74.68 ± 0.69	85.14 ± 0.57
PCL-v1	39.49 ± 0.87	84.35 ± 0.60	40.59 ± 0.76	62.75 ± 0.63	64.09 ± 0.79	64.48 ± 0.78	77.25 ± 0.75	57.45 ± 0.83	85.51 ± 0.64	80.89 ± 0.62
PIRL	42.91 ± 0.93	85.04 ± 0.62	46.87 ± 0.74	64.39 ± 0.63	69.32 ± 0.76	72.80 ± 0.69	89.52 ± 0.51	61.32 ± 0.77	74.05 ± 0.69	85.03 ± 0.59
PCL-v2	34.36 ± 0.75	86.33 ± 0.54	42.57 ± 0.70	70.96 ± 0.59	74.10 ± 0.69	72.84 ± 0.74	87.52 ± 0.52	61.00 ± 0.78	85.16 ± 0.66	84.80 ± 0.57
SimCLR-v1	48.11 ± 0.98	94.10 ± 0.36	53.46 ± 0.80	70.65 ± 0.66	77.10 ± 0.70	76.71 ± 0.65	93.10 ± 0.38	65.13 ± 0.77	86.52 ± 0.58	89.71 ± 0.47
MoCo-v2	35.97 ± 0.80	90.14 ± 0.48	49.55 ± 0.80	69.47 ± 0.62	75.62 ± 0.70	78.08 ± 0.67	91.12 ± 0.46	66.34 ± 0.80	87.91 ± 0.59	89.18 ± 0.48
SimCLR-v2	47.12 ± 0.96	94.92 ± 0.34	52.64 ± 0.77	71.90 ± 0.61	79.71 ± 0.66	79.06 ± 0.63	93.83 ± 0.37	69.85 ± 0.74	86.29 ± 0.58	90.99 ± 0.45
SeLa-v2	36.35 ± 0.77	89.85 ± 0.53	47.99 ± 0.78	71.27 ± 0.59	76.29 ± 0.72	77.81 ± 0.62	90.11 ± 0.51	67.69 ± 0.77	81.36 ± 0.67	90.80 ± 0.46
InfoMin	35.06 ± 0.75	87.03 ± 0.53	49.67 ± 0.79	67.28 ± 0.62	71.72 ± 0.72	73.43 ± 0.75	87.53 ± 0.57	65.95 ± 0.77	86.98 ± 0.57	86.54 ± 0.55
BYOL	53.88 ± 0.99	96.84 ± 0.28	58.77 ± 0.81	70.59 ± 0.62	79.19 ± 0.68	**81.33 ± 0.59**	**96.06 ± 0.30**	71.39 ± 0.72	92.20 ± 0.46	91.63 ± 0.43
DeepCluster-v2	47.73 ± 0.97	94.75 ± 0.35	58.17 ± 0.82	74.47 ± 0.61	80.52 ± 0.65	78.79 ± 0.59	95.44 ± 0.32	72.71 ± 0.72	89.13 ± 0.56	92.95 ± 0.41
SwAV	46.22 ± 0.91	94.43 ± 0.37	56.08 ± 0.82	72.73 ± 0.62	79.32 ± 0.67	79.80 ± 0.57	94.55 ± 0.37	69.65 ± 0.73	88.76 ± 0.56	93.00 ± 0.42
Supervised	**58.35 ± 0.96**	**97.61 ± 0.24**	**73.68 ± 0.84**	**77.50 ± 0.55**	**83.74 ± 0.61**	80.83 ± 0.59	94.19 ± 0.41	**76.23 ± 0.71**	**97.45 ± 0.28**	**93.78 ± 0.38**

Table 7. 5-way 50-shot transfer on the Kornblith datasets, apart from Caltech101, Cars and Flowers which do not have enough images per class for this setup. We report the average accuracy and 95% confidence interval over 600 test episodes. Results style: **best**, second best.

	Aircraft	CIFAR10	CIFAR100	DTD	Food	Pets	SUN397
InsDis	51.06 ± 0.88	71.77 ± 0.52	77.57 ± 0.63	83.97 ± 0.47	73.43 ± 0.63	84.78 ± 0.56	92.10 ± 0.39
MoCo-v1	51.20 ± 0.89	68.22 ± 0.54	75.22 ± 0.70	84.76 ± 0.49	74.19 ± 0.60	85.65 ± 0.55	92.31 ± 0.38
PCL-v1	44.78 ± 0.82	69.35 ± 0.53	72.07 ± 0.70	77.18 ± 0.58	67.46 ± 0.67	90.76 ± 0.46	87.59 ± 0.47
PIRL	52.17 ± 0.88	72.23 ± 0.52	78.43 ± 0.64	83.94 ± 0.51	73.05 ± 0.62	85.58 ± 0.53	92.44 ± 0.39
PCL-v2	38.48 ± 0.78	79.51 ± 0.45	82.86 ± 0.53	83.79 ± 0.48	72.30 ± 0.65	89.96 ± 0.48	90.19 ± 0.42
SimCLR-v1	55.29 ± 0.93	79.72 ± 0.49	84.43 ± 0.55	86.24 ± 0.43	77.24 ± 0.59	92.83 ± 0.40	94.34 ± 0.33
MoCo-v2	41.22 ± 0.79	78.01 ± 0.45	83.01 ± 0.57	86.42 ± 0.46	77.17 ± 0.60	92.25 ± 0.42	92.98 ± 0.36
SimCLR-v2	56.33 ± 0.91	81.36 ± 0.48	87.79 ± 0.49	87.99 ± 0.42	81.65 ± 0.53	93.51 ± 0.38	95.51 ± 0.28
SeLa-v2	43.04 ± 0.83	79.16 ± 0.50	84.11 ± 0.59	87.77 ± 0.43	80.10 ± 0.56	89.84 ± 0.44	95.11 ± 0.29
InfoMin	39.91 ± 0.76	74.23 ± 0.53	79.16 ± 0.57	83.09 ± 0.49	76.12 ± 0.59	91.61 ± 0.42	91.05 ± 0.42
BYOL	65.69 ± 0.88	80.49 ± 0.47	87.57 ± 0.50	89.12 ± 0.42	83.04 ± 0.51	96.18 ± 0.30	95.89 ± 0.26
DeepCluster-v2	57.84 ± 0.93	82.56 ± 0.47	88.11 ± 0.46	**89.34 ± 0.40**	84.38 ± 0.49	94.62 ± 0.36	96.57 ± 0.24
SwAV	55.88 ± 0.89	80.30 ± 0.49	86.93 ± 0.51	89.13 ± 0.41	81.94 ± 0.54	94.58 ± 0.36	**96.64 ± 0.24**
Supervised	**71.97 ± 0.83**	**85.80 ± 0.40**	**90.24 ± 0.42**	88.23 ± 0.44	**85.26 ± 0.48**	**98.54 ± 0.16**	96.61 ± 0.24

capture the fact that an absolute increase of 1% in accuracy has varying significance depending on if, e.g., the accuracy goes from 50% to 51% or if it goes from 98% to 99%, we apply a logit-transformation to any metric that is bounded in the range 0 to 1.

All correlations computed against ImageNet performance use the logit-transformed ImageNet top-1 accuracy. Additionally, we logit-transform all recognition accuracies, AP metrics from detection, $11.25°$, $22.5°$ and $30°$ in surface normal estimation, and both mean-IOU and accuracy in semantic segmentation. The only metrics not transformed in this way are the Mean and Median errors in surface normal estimation. We negate these two error metrics before computing correlations in Fig. 2 so reading the figure is easier.

For correlations in Fig. 1, we average the logit-transformed accuracies across datasets in all many-shot and few-shot settings to produce a single correlation coefficient for each setting. For both detection settings we report the correlation of the logit-transformed AP50 metric and for the two dense settings we report correlations of the logit-transformed $11.25°$ and mean-IOU metrics.

For calibration (Fig. 3), perceptual similarity and attentive diffusion (Table 9), we similarly use logit-transformed values when computing correlations. For the red, green and blue colour channel errors in our image reconstruction, we report correlations of their raw values.

A.8. Image reconstruction by feature inversion

To see what information is retained by the models, we evaluate how well an image can be reconstructed from an extracted feature. We follow the deep image prior [55] protocol of feature inversion. Given an image I, we first extract its feature vector $f(I)$ by passing it through the pre-trained model backbone f. Next, we initialise a reconstruction network g_θ, parameterised by θ, which maps from a fixed noise input z to an image $g_\theta(z)$. The reconstruction network is then trained to output an image which, when passed through our pre-trained backbone, produces a feature close to that of I. The optimisation problem is:

$$\arg\min_\theta f(g_\theta(z)) - f(I). \quad (1)$$

We extract the features from our pre-trained backbone from the 4th residual block, giving a vector size of $2048 \times 7 \times 7$. The reconstruction network is trained for 3000 iterations using the Adam optimiser with a learning rate of 0.001. The architecture of the reconstruction network is the same as in the original deep image prior paper [55] and the study in [69].

A.9. Computing the saliency maps

We use an occlusion mask of 10×10 pixels and pass it over images resized to 242×242 which we then crop to 224×224 to ensure all pixels are occluded the same number of times. The attention values are computed as the root relative squared error (RRSE) of the original features and

Table 8. Few-shot transfer of pre-trained models using prototypical networks. Here, we present few-shot transfer results for 5-way 5-shot and 5-way 50-shot settings on CD-FSL. We report the average accuracy and 95% confidence interval over 600 test episodes. Results style: **best**, second best.

	CropDiseases		EuroSAT		ISIC		ChestX	
	5-shot	50-shot	5-shot	50-shot	5-shot	50-shot	5-shot	50-shot
InsDis	88.01 ± 0.58	92.70 ± 0.43	81.29 ± 0.63	88.25 ± 0.47	43.90 ± 0.55	55.76 ± 0.50	25.67 ± 0.42	31.77 ± 0.44
MoCo-v1	87.87 ± 0.58	92.87 ± 0.42	81.32 ± 0.61	87.72 ± 0.46	**44.42 ± 0.55**	56.81 ± 0.52	25.92 ± 0.45	32.74 ± 0.43
PCL-v1	72.89 ± 0.69	82.83 ± 0.55	66.56 ± 0.76	76.41 ± 0.63	33.21 ± 0.48	39.77 ± 0.45	23.33 ± 0.40	27.40 ± 0.42
PIRL	86.22 ± 0.63	92.18 ± 0.44	82.14 ± 0.63	88.55 ± 0.44	43.89 ± 0.54	56.89 ± 0.52	25.60 ± 0.41	31.44 ± 0.47
PCL-v2	87.57 ± 0.60	93.57 ± 0.40	81.10 ± 0.54	89.23 ± 0.37	37.47 ± 0.52	46.82 ± 0.46	24.87 ± 0.42	30.56 ± 0.43
SimCLR-v1	90.29 ± 0.52	94.49 ± 0.37	82.78 ± 0.56	90.55 ± 0.36	43.99 ± 0.55	56.16 ± 0.53	26.36 ± 0.44	33.16 ± 0.47
MoCo-v2	87.62 ± 0.60	93.61 ± 0.40	84.15 ± 0.52	89.83 ± 0.37	42.60 ± 0.55	55.68 ± 0.53	25.26 ± 0.44	32.20 ± 0.43
SimCLR-v2	90.80 ± 0.52	95.80 ± 0.29	86.45 ± 0.49	92.07 ± 0.30	43.66 ± 0.58	56.83 ± 0.54	26.34 ± 0.44	33.23 ± 0.47
SeLa-v2	90.96 ± 0.54	95.40 ± 0.33	84.56 ± 0.57	88.51 ± 0.59	39.97 ± 0.55	51.31 ± 0.52	25.60 ± 0.44	32.81 ± 0.44
InfoMin	87.77 ± 0.61	92.93 ± 0.40	81.68 ± 0.59	87.61 ± 0.43	39.03 ± 0.55	51.58 ± 0.51	25.78 ± 0.44	31.58 ± 0.44
BYOL	92.71 ± 0.47	96.69 ± 0.27	83.64 ± 0.54	90.46 ± 0.35	43.09 ± 0.56	**58.03 ± 0.52**	26.39 ± 0.43	**34.17 ± 0.45**
DeepCluster-v2	**93.63 ± 0.44**	**97.04 ± 0.27**	**88.39 ± 0.49**	93.07 ± 0.31	40.73 ± 0.59	53.65 ± 0.54	26.51 ± 0.45	**34.17 ± 0.48**
SwAV	93.49 ± 0.46	96.72 ± 0.28	87.29 ± 0.54	**93.36 ± 0.31**	39.66 ± 0.54	51.10 ± 0.50	**26.54 ± 0.48**	33.86 ± 0.46
Supervised	89.37 ± 0.55	94.32 ± 0.36	83.81 ± 0.55	89.62 ± 0.37	39.38 ± 0.58	52.54 ± 0.56	25.22 ± 0.41	32.34 ± 0.45

Table 9. Numerical values for the results presented in Figs 3-4. Columns 1-4: Expected calibration error (ECE) using 15 bins for unscaled models and models further calibrated using temperature scaling. Columns 5-7: Average perceptual distance computed on reconstructed images, using three different measures of the Learned Perceptual Image Patch Similarity (LPIPS) metric [66]. Columns 8-10: Mean squared errors between the colour channels of reconstructed and original images. Column 11: Attentive diffusion measured as the percentage of attention values above the mean attention over an image. Higher value means wider attention. Results style: **lowest**, second lowest.

	Many-shot (Linear)		Many-shot (Finetune)		Perceptual Distance			Colour Error			Attention
	Unscaled	Scaled	Unscaled	Scaled	AlexNet	VGG	SqueezeNet	Red	Green	Blue	Diffusion
InsDis	12.68	2.72	8.15	2.18	0.58	0.71	0.48	3971	2734	3394	48.48
MoCo-v1	14.15	2.58	8.21	2.28	0.62	0.74	0.53	4073	3044	3512	47.92
PCL-v1	14.06	3.71	7.29	2.63	0.74	0.81	0.65	4598	3954	4141	**41.43**
PIRL	15.68	2.68	8.37	2.12	0.59	0.72	0.50	3607	3070	3435	48.12
PCL-v2	11.07	2.85	5.04	2.34	0.56	0.66	0.47	3008	2807	3101	43.91
SimCLR-v1	8.45	2.13	5.29	2.46	0.56	0.70	0.47	3224	2667	3223	46.07
MoCo-v2	9.25	2.67	6.01	2.25	0.54	0.67	0.45	3179	2514	2695	45.39
SimCLR-v2	9.71	2.19	6.06	2.45	0.55	0.68	0.46	3655	2855	3404	47.91
SeLa-v2	11.52	2.81	5.20	2.10	0.69	0.72	0.57	3962	3775	4315	47.68
InfoMin	**7.05**	2.99	5.32	2.23	0.49	0.60	0.39	**2592**	**2403**	**2594**	43.73
BYOL	10.23	**1.93**	5.82	1.96	0.59	0.71	0.48	3765	3268	3471	48.81
DeepCluster-v2	8.69	2.17	4.94	**1.85**	0.58	0.67	0.48	3527	3170	3804	48.69
SwAV	8.25	2.16	4.80	1.86	0.57	0.67	0.46	3560	3186	3565	49.47
Supervised	10.35	2.22	**4.48**	1.90	**0.47**	**0.55**	**0.37**	2788	2917	2903	43.88
Correlation to ImageNet	-0.77	-0.59	-0.90	-0.59	-0.51	-0.69	-0.57	-0.56	-0.11	-0.22	0.09

the occluded features, averaged over all times a pixel is occluded (10^2). The RRSE ensures that the distances are invariant to the scale of the original features.

Table 10. Training details as reported by original authors for all models used in this paper. Asterisks (*) note models we obtain from PyContrast instead of original authors.

	Epochs	Batch size	Target net	Mom. enc.	Mem. bank	Proj. head	Jigsaw	Grayscale	Colour jitter	Solarize	Blur	Random crop	Horiz. flip	Normalize
InsDis*	200	256			✓			✓	✓			✓	✓	✓
MoCo-v1	200	256		✓				✓	✓			✓	✓	✓
PCL-v1	200	256		✓				✓	✓			✓	✓	✓
PIRL*	200	1024			✓		✓		✓			✓	✓	✓
PCL-v2	200	256		✓		✓		✓	✓		✓	✓	✓	✓
SimCLR-v1	1000	4096				✓		✓	✓		✓	✓	✓	
MoCo-v2	800	256		✓		✓		✓	✓		✓	✓	✓	✓
SimCLR-v2	800	4096		✓		✓		✓	✓		✓	✓	✓	
SeLa-v2	400	4096			✓	✓		✓	✓		✓	multi	✓	✓
InfoMin	800	256		✓		✓	✓	✓	✓		✓	✓	✓	✓
BYOL	1000	4096	✓			✓		✓	✓	✓	✓	✓	✓	✓
DeepCluster-v2	800	4096			✓	✓		✓	✓		✓	multi	✓	✓
SwAV	800	4096				✓		✓	✓		✓	multi	✓	✓
Supervised	120	256							PCA			✓	✓	✓

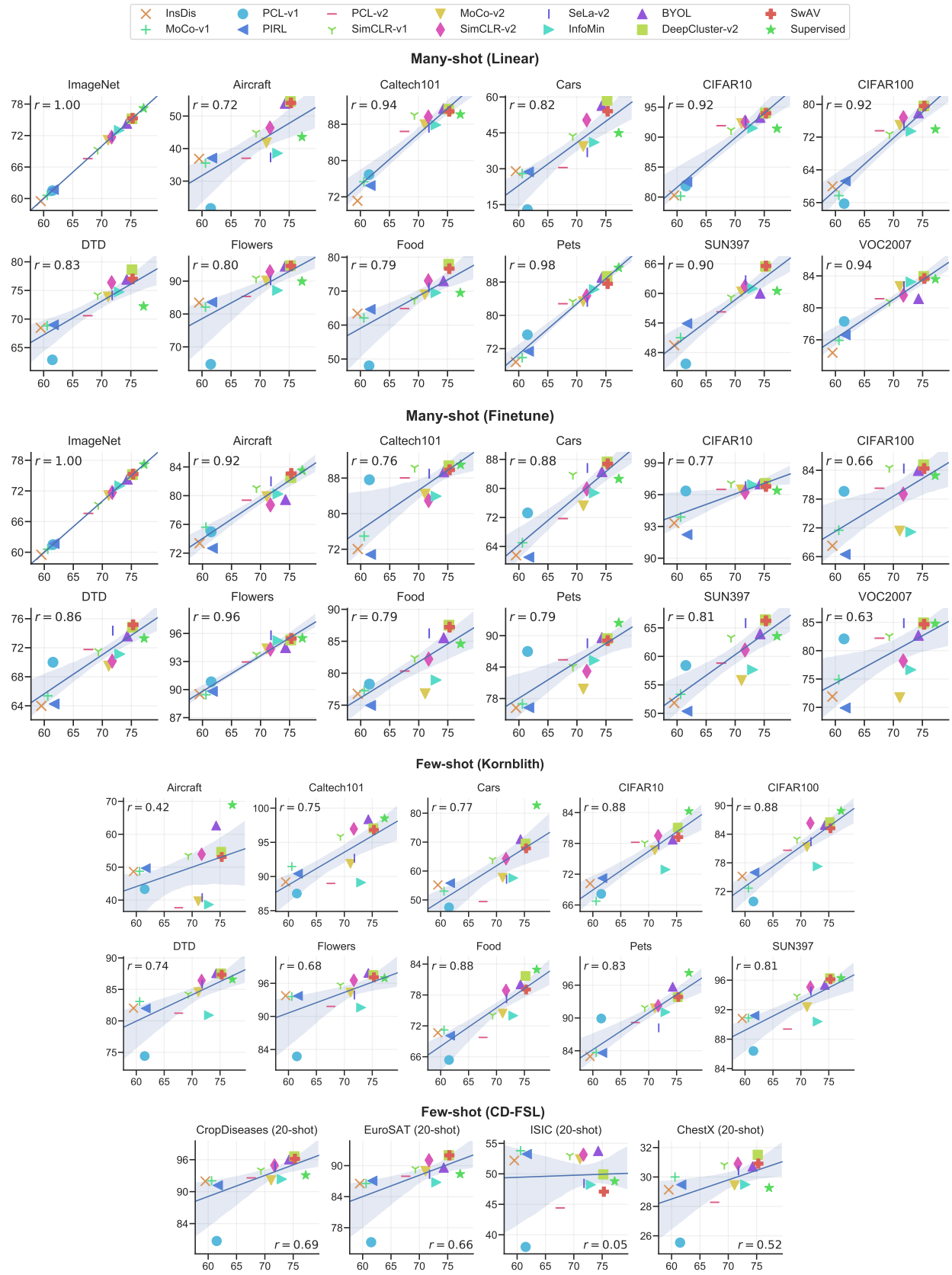

Figure 5. Individual plots of transfer correlations between ImageNet accuracy on the x-axis and target performance on the y-axis.

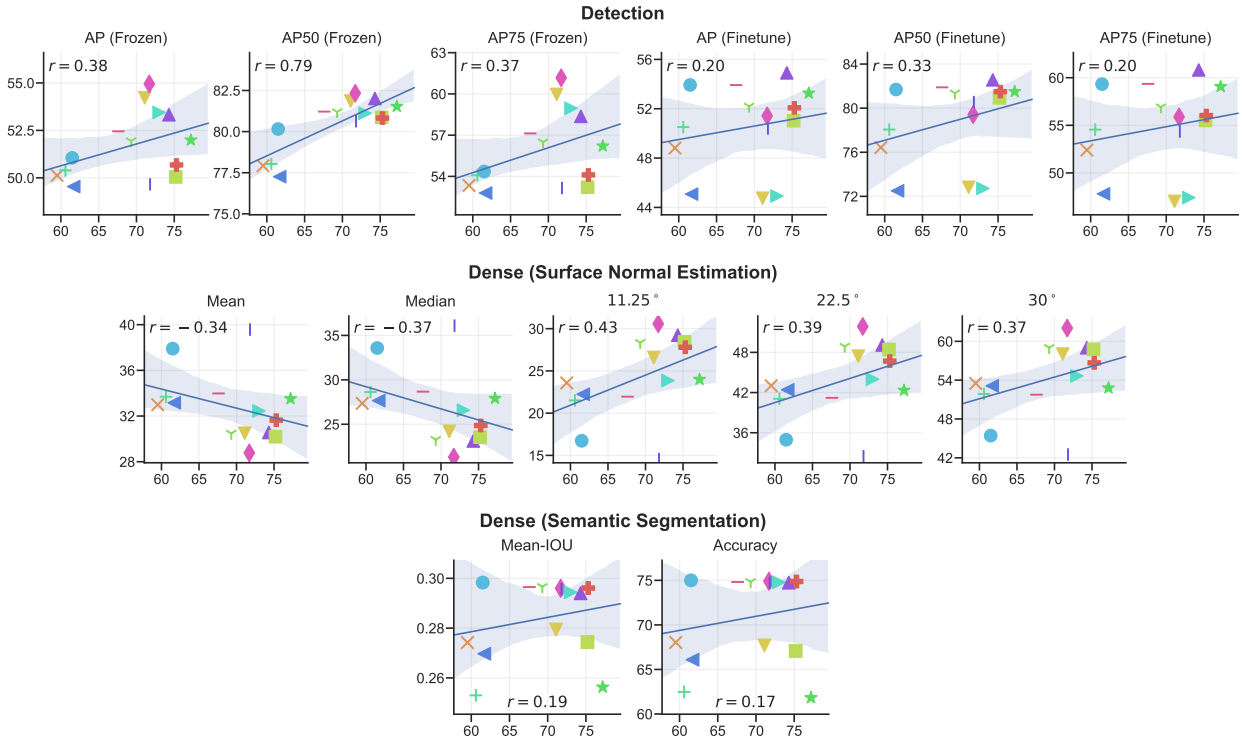

Figure 6. Individual plots of transfer correlations between ImageNet accuracy on the x-axis and target performance on the y-axis.

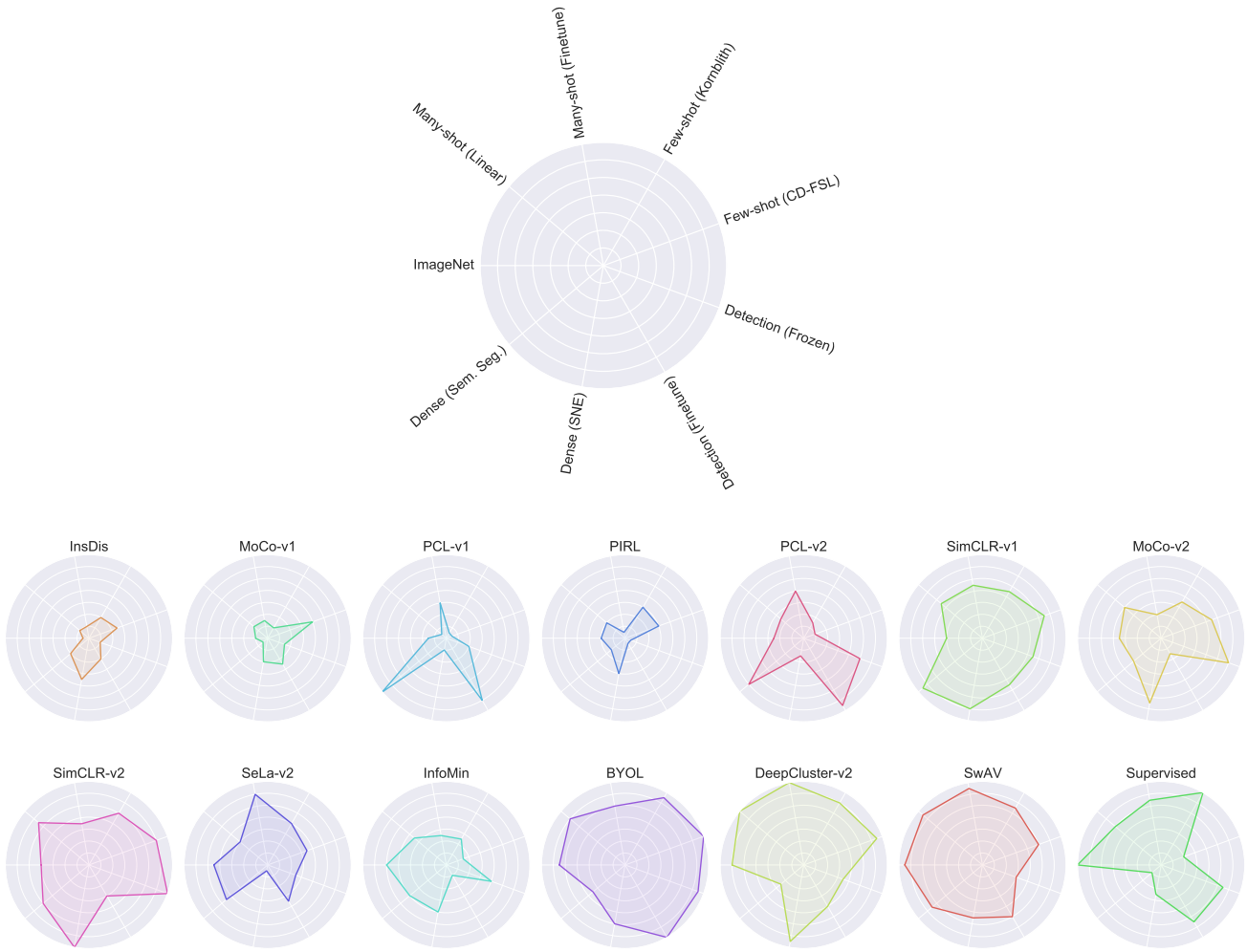

Figure 7. Radar charts of model performance on ImageNet and our eight different evaluation settings. In each setting we compute the rankings of the models (from averaged performance where there are multiple datasets). In each plot above, a higher rank (better performance) places the line closer to the outer edge of the circle. A larger total area roughly corresponds to better performance across a wide range of transfer settings. The rankings are based on average accuracy in the many-shot and few-shot settings, AP50 for frozen and finetuned detection, mean error for surface normal estimation and mean IOU for semantic segmentation.

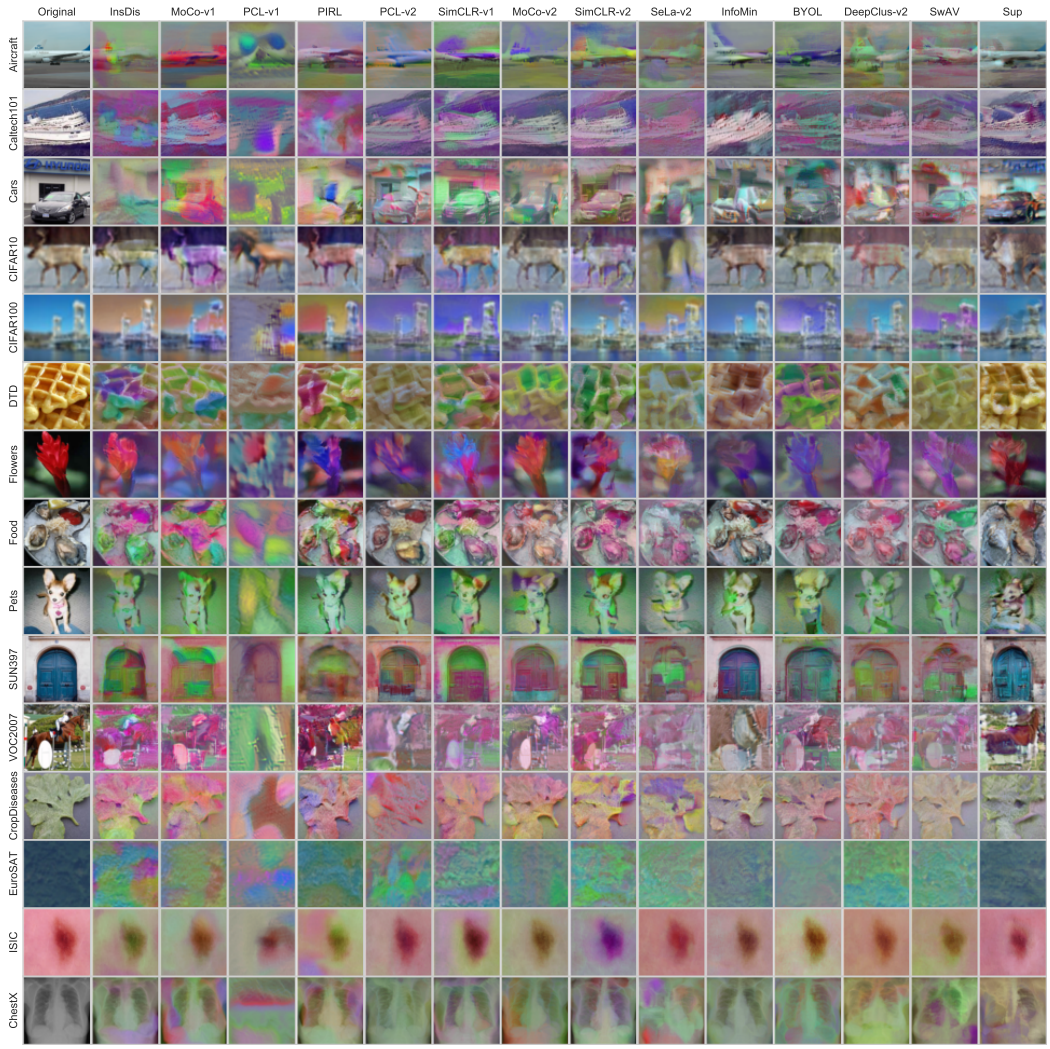

Figure 8. Deep image prior reconstructions on one image for each of 15 datasets.

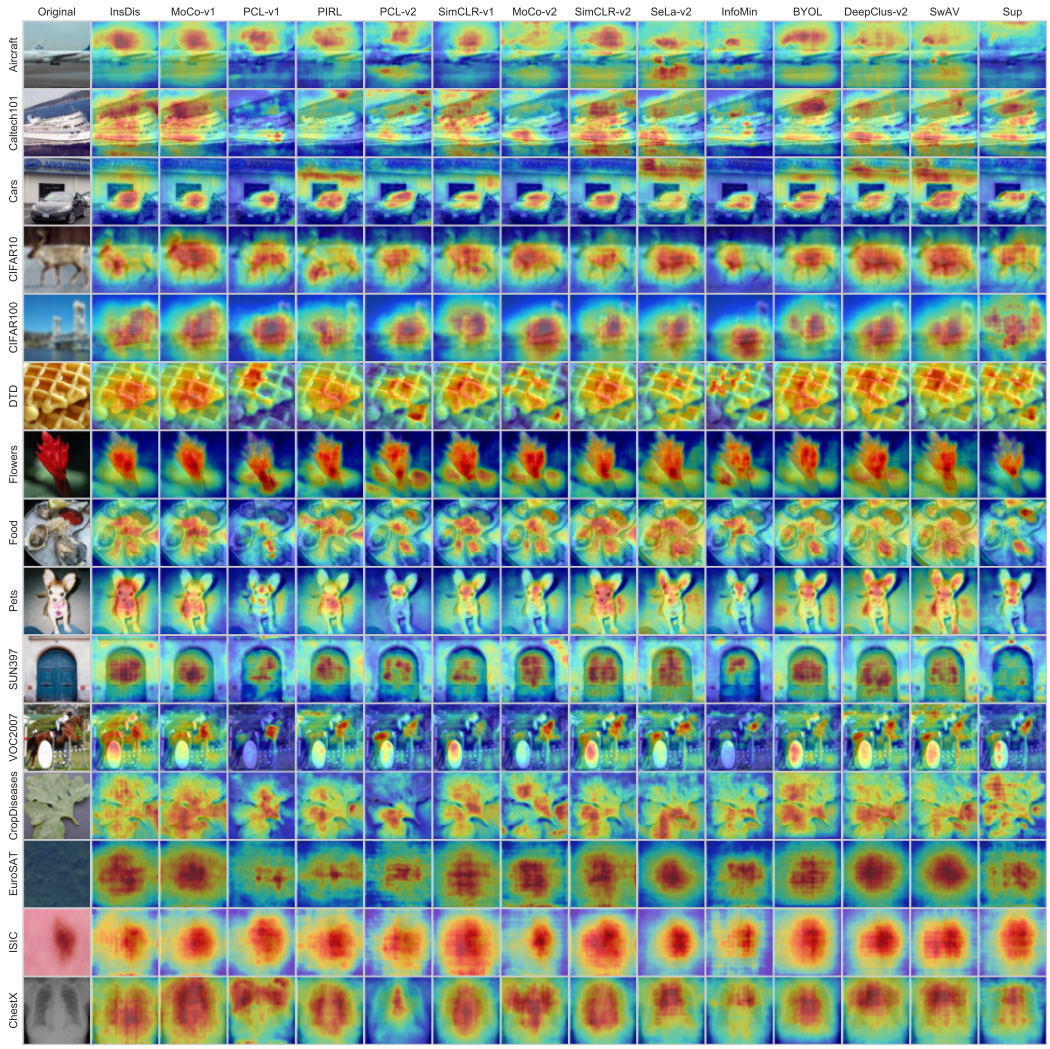

Figure 9. Saliency maps for all models on one image for each of 15 datasets.