SCAN: Learning to Classify Images without Labels

Wouter Van Gansbeke[1⋆] Simon Vandenhende[1⋆] Stamatios Georgoulis[2]
Marc Proesmans[1] Luc Van Gool[1,2]

[1]KU Leuven/ESAT-PSI [2]ETH Zurich/CVL, TRACE

A Smaller datasets

We include additional qualitative results on the smaller datasets, i.e. CIFAR10 [8], CIFAR100-20 [8] and STL10 [4]. We used the models from the state-of-the-art comparison.

A.1 Prototypical examples

Figure S1 visualizes a prototype image for every cluster on CIFAR10, CIFAR100-20 and STL-10. The object of interest is clearly recognizable in the images. It is worth noting that the prototypical examples on CIFAR10 and STL10 can be matched with the ground-truth classes of the dataset. This is not the case for CIFAR100-20, e.g. *bus* and *bicycle* belong to the *vehicles 1* ground-truth class. This behavior can be easily understood since CIFAR-20 makes use of superclasses. As a consequence, it is difficult to explain the intra-class variance from visual appearance alone. Interestingly, we can reduce this mismatch through overclustering (see Sec 3.4.).

A.2 Low confidence examples

Figure S2 shows examples for which the network produces low confidence predictions. In most cases, it is hard to determine the correct class label. The difficult examples include objects which are: only partially visible, occluded, under bad lighting conditions, etc.

B ImageNet

B.1 Training setup

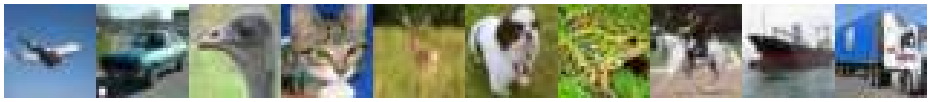
We summarize the training setup for ImageNet below.

Pretext Task Similar to our setup on the smaller datasets, we select instance discrimination as our pretext task. In particular, we use the implementation from MoCo [3]. We use a ResNet-50 model as backbone.

⋆ Authors contributed equally

Fig. S1: Prototype images on the smaller datasets.

(a) CIFAR10



(b) STL10

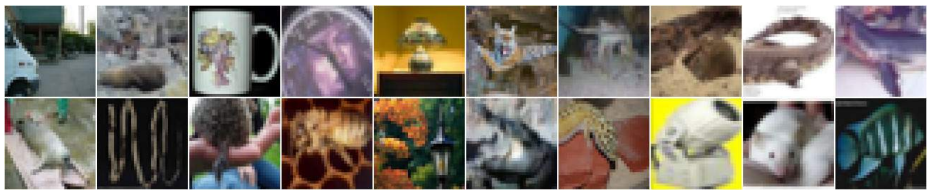
(c) CIFAR100-20

Fig. S2: Low confidence predictions.

(a) CIFAR10

(b) STL10

(c) CIFAR100-20



Clustering Step We freeze the backbone weights during the clustering step, and only train the final linear layer using the SCAN-loss. More specifically, we train ten separate linear heads in parallel. When initiating the self-labeling step, we select the head with the lowest loss to continue training. Every image is augmented using augmentations from SimCLR [2]. We reuse the entropy weight from before (5.0), and train with batches of size 512, 1024 and 1024 on the subsets of 50, 100 and 200 classes respectively. We use an SGD optimizer with momentum 0.9 and initial learning rate 5.0. The model is trained for 100 epochs. On the full ImageNet dataset, we increase the batch size and learning rate to 4096 and 30.0 respectively, and decrease the number of neighbors to 20.

Self-Labeling Step We use the strong augmentations from RandAugment to finetune the weights through self-labeling. The model weights are updated for 25 epochs using SGD with momentum 0.9. The initial learning rate is set to 0.03 and kept constant. Batches of size 512 are used. Importantly, the model weights are updated through an exponential moving average with $\alpha = 0.999$. We did not find it necessary to apply class balancing in the cross-entropy loss.

B.2 ImageNet - Subsets

Confusion matrix Figure S3 shows a confusion matrix on the ImageNet-50 dataset. Most of the mistakes can be found between classes that are hard to disentangle, e.g. *'Giant Schnauzer'* and *'Flat-coated Retriever'* are both black dog breeds, *'Guacamole'* and *'Mashed Potato'* are both food, etc.

Prototype examples Figure S4 shows a prototype image for every cluster on the ImageNet-50 subset. This figure extends Figure 9 from the main paper. Remarkably, the vast majority of prototype images can be matched with one of the ground-truth classes.

Low confidence examples Figure S5 shows examples for which the model produces low confidence predictions on the ImageNet-50 subset. In a number of cases, the low confidence output can be attributed to multiple objects being visible in the scene. Other cases can be explained by the partial visibility of the object, distracting elements in the scene, or ambiguity of the object of interest.

B.3 ImageNet - Full

We include additional qualitative results on the full ImageNet dataset. In particular, Figures S6, S7 and S8 show images from the validation set that were assigned to the same cluster. These can be viewed together with Figure 11 in the main paper. Additionally, we show some mistakes in Figure S9. The failure cases occur when the model focuses too much on the background, or when the network cannot easily discriminate between pairs of similarly looking images. However, in most cases, we can still attach some semantic meaning to the clusters, e.g. animals in cages, white fences.

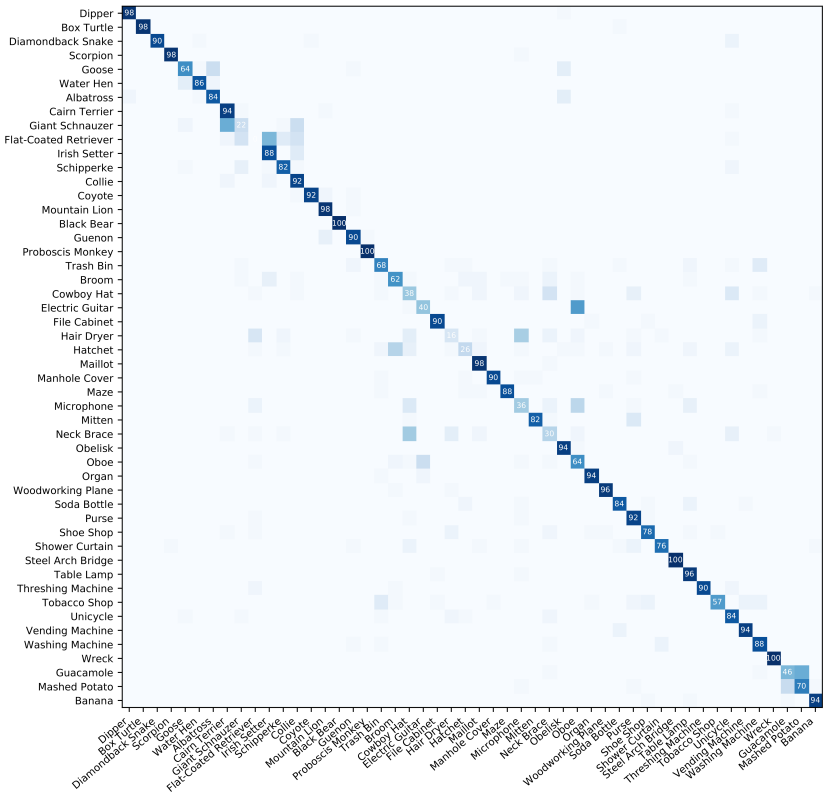


Fig. S3: Confusion matrix on ImageNet-50.

C Experimental setup

C.1 Datasets

Different from prior work [7, 1, 9, 10], we do not train and evaluate on the full datasets. Differently, we use the standard train-val splits to study the generalization properties of our models. Additionally, we report the mean and standard deviation on the smaller datasets. We would like to encourage future works to adopt this procedure as well. Table S1 provides an overview of the number of classes, the number of images and the aspect ratio of the used datasets. The selected classes on ImageNet-50, ImageNet-100 and ImageNet-200 can be found in our git repository.

C.2 Augmentations

As shown in our experiments, it is beneficial to apply strong augmentations during training. The strong augmentations were composed of four randomly selected

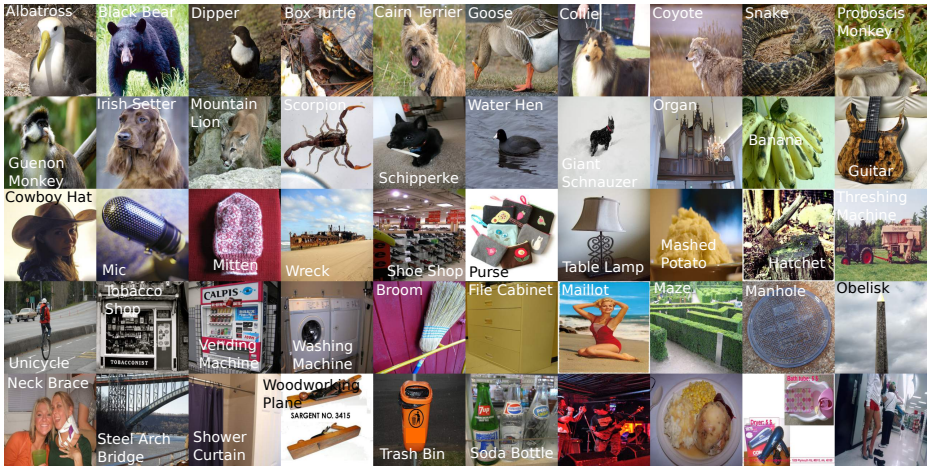


Fig. S4: Prototype images on ImageNet-50.

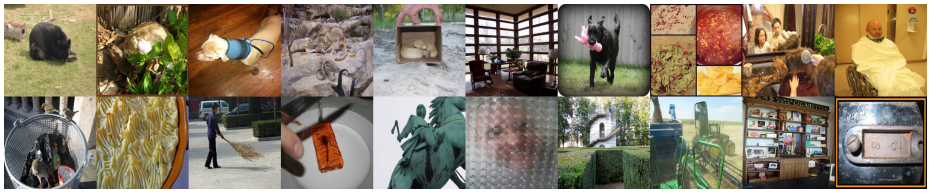


Fig. S5: Low confidence examples on ImageNet-50.

Table S1: Datasets overview

Dataset	Classes	Train images	Val images	Aspect ratio
CIFAR10	10	50,000	10,000	32 x 32
CIFAR100-20	20	50,000	10,000	32 x 32
STL10	10	5,000	8,000	96 x 96
ImageNet-50	50	64,274	2,500	224 x 224
ImageNet-100	100	128,545	5,000	224 x 224
ImageNet-200	200	256,558	10,000	224 x 224
ImageNet	1000	1,281,167	50,000	224 x 224

transformations from RandAugment [5], followed by Cutout [6]. The transformation parameters were uniformly sampled between fixed intervals. Table S2 provides a detailed overview. We applied an identical augmentation strategy across all datasets.

Fig. S6: Example clusters of ImageNet-1000 (1).

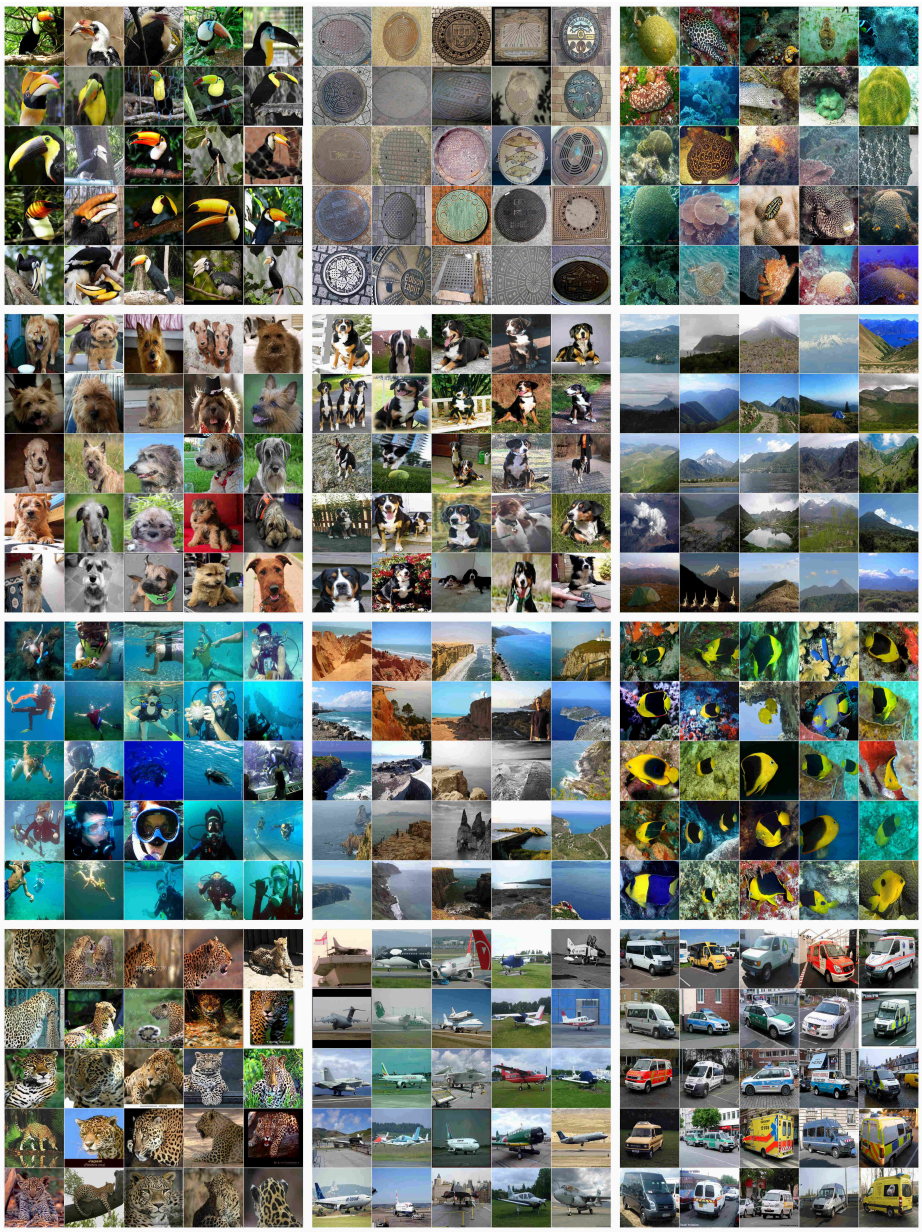


Fig. S7: Example clusters of ImageNet-1000 (2).

Fig. S8: Example clusters of ImageNet-1000 (3).

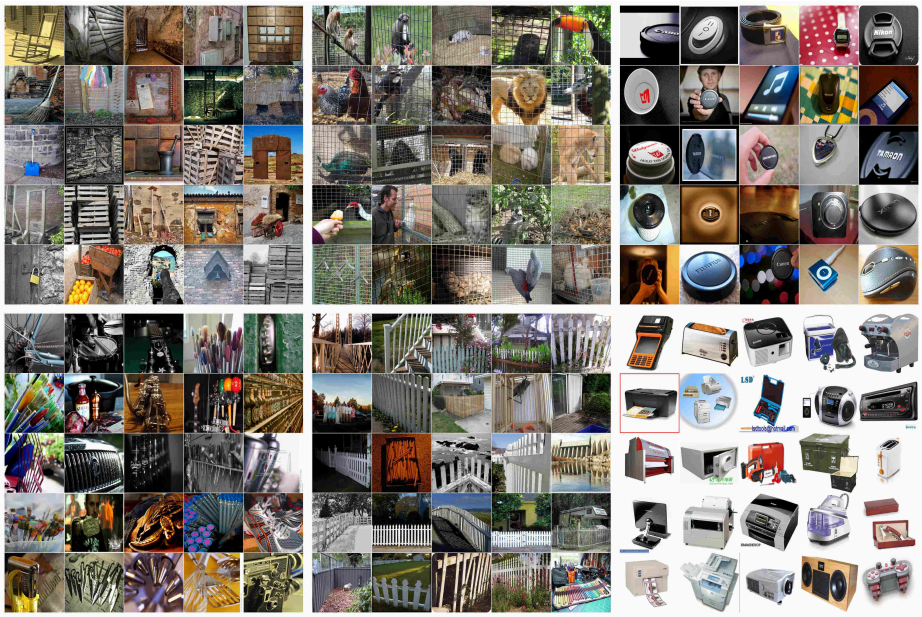


Fig. S9: Incorrect clusters of ImageNet-1000 predicted by our model.

Table S2: List of transformations. The strong transformations are composed by randomly selecting four transformations from the list, followed by Cutout.

Transformation	**Parameter**	**Interval**
Identity	-	-
Autocontrast	-	-
Equalize	-	-
Rotate	θ	$[-30, 30]$
Solarize	T	$[0, 256]$
Color	C	$[0.05, 0.95]$
Contrast	C	$[0.05, 0.95]$
Brightness	B	$[0.05, 0.95]$
Sharpness	S	$[0.05, 0.95]$
Shear X	R	$[-0.1, 0.1]$
Translation X	λ	$[-0.1, 0.1]$
Translation Y	λ	$[-0.1, 0.1]$
Posterize	B	$[4, 8]$
Shear Y	R	$[-0.1, 0.1]$

References

1. Chang, J., Wang, L., Meng, G., Xiang, S., Pan, C.: Deep adaptive image clustering. In: ICCV (2017)
2. Chen, T., Kornblith, S., Norouzi, M., Hinton, G.: A simple framework for contrastive learning of visual representations. arXiv preprint arXiv:2002.05709 (2020)
3. Chen, X., Fan, H., Girshick, R., He, K.: Improved baselines with momentum contrastive learning. arXiv preprint arXiv:2003.04297 (2020)
4. Coates, A., Ng, A., Lee, H.: An analysis of single-layer networks in unsupervised feature learning. In: JMLR (2011)
5. Cubuk, E.D., Zoph, B., Shlens, J., Le, Q.V.: Randaugment: Practical automated data augmentation with a reduced search space. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops. pp. 702–703 (2020)
6. DeVries, T., Taylor, G.W.: Improved regularization of convolutional neural networks with cutout. arXiv preprint arXiv:1708.04552 (2017)
7. Ji, X., Henriques, J.F., Vedaldi, A.: Invariant information clustering for unsupervised image classification and segmentation. In: ICCV (2019)
8. Krizhevsky, A., Hinton, G., et al.: Learning multiple layers of features from tiny images (2009)
9. Xie, J., Girshick, R., Farhadi, A.: Unsupervised deep embedding for clustering analysis. In: ICML (2016)
10. Yang, J., Parikh, D., Batra, D.: Joint unsupervised learning of deep representations and image clusters. In: CVPR (2016)