Supplementary Material

1 Clustering Combination

In this section we provide an illustration figure for the effects of combining multiple clusterings in Figure 1, which is also mentioned in Section 5.2. Additionally, we show the nearest neighbor validation performances in Table 1 to support our hyper-parameter choices for H, m in different architectures.

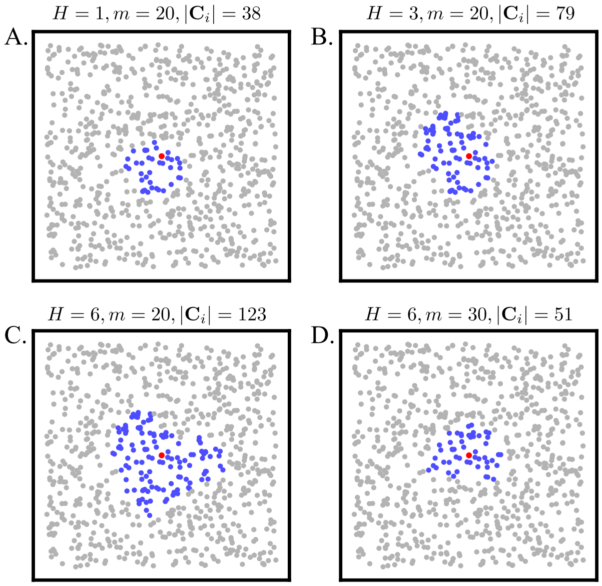

Figure 1: Illustration of the effect of combining across multiple clusterings to achieve robustness. The target embedded vector $\mathbf{v}_i$ is represented by the red dot, while blue dots represent close neighbors $\mathbf{C}_i$ under the specified hyperparameter settings.

Network Setting	A	V	R-18	R-50
$(1, 1\text{k})$	–	–	35.2	–
$(1, 10\text{k})$	30.6	38.9	35.7	40.2
$(1, 20\text{k})$	–	–	35.0	–
$(3, 10\text{k})$	**31.1**	–	36.2	–
$(6, 10\text{k})$	30.4	**39.7**	37.3	42.4
$(10, 10\text{k})$	–	–	36.1	42.3
$(10, 30\text{k})$	–	–	**37.9**	**43.4**

Table 1: Nearest neighbor validation performances of different architectures trained with different choices of $\mathbf{C}_i$. "A" means "AlexNet". "V" means VGG16. "R" means "ResNet". Similarly to Table 5 in the main text, $(1, 10\text{k})$ means clustering-based $\mathbf{C}_i$ with $H = 1$ and $m = 10000$.

2 Results Details

2.1 Transfer Learning Details

Besides the settings listed in the main paper, there are additional settings for data augmentation during our transfer learning training to ImageNet and Places 205 datasets. In general, we use random crop and random horizontal flip as data augmentation techniques during transfer learning for all architectures on both ImageNet and Places 205 datasets, where the specific random crop implementation varies across networks and datasets. For AlexNet on ImageNet and all architectures on Places 205, we use the AlexNet style random crop [5], which is first resizing the image so that its smallest side is 256 and then randomly cropping a 224 × 224 patch. For VGG16, ResNet-18, and ResNet-50 on ImageNet, we use the ResNet style random crop [4], which is first randomly choosing a patch whose aspect ratio and area suffice two conditions and then resizing that path to 224×224. The two sufficed conditions are: its area is at least 20% of the overall area and at most 100% of the overall area; its aspect ratio ranges from $3/4$ to $4/3$. We use the same data augmentation techniques for the same architecture trained with different methods.

2.2 DeepCluster Results Details

The DeepCluster [1] VGG16, ResNet-18, and ResNet-50 results are produced by us, where the DC-VGG16 network is provided by the authors and the DC-ResNet-18 and DC-ResNet-50 networks are trained by us using the provided source codes.

More specifically, for ResNet-18, two implementations of DC-ResNet-18 network are trained. Both of them modifies the standard ResNet-18 architecture by removing the final pooling and final fully connected layer and then adding additional fully connected layers, where the last layer has 10000 units. One implementation (DC-ResNet-18-A) only has that 10000-unit fully connected layer and the other implementation (DC-ResNet-18-B) has two more 4096-unit fully connected layers before that. We find that DC-ResNet-18-B performs slightly better than DC-ResNet-18-A and thus report the performances of DC-ResNet-18-B in the main paper.

Similarly for ResNet-50, two implementations (DC-ResNet-50-A and DC-ResNet-50-B) are trained. However, we find it impossible to train DC-ResNet-50-B as the k-means clustering results always become trivial at the third epoch. So the results reported in the paper are from DC-ResNet-50-A, which should only be slightly worse than DC-ResNet-50-B.

Other hyper-parameters for network training are mostly the same as used in the provided source codes. Meanwhile, all hyper-parameters for transfer learning to ImageNet and Places 205 are also the same as provided, except the data augmentation techniques which are the same as described in Section 2.1.

2.3 Places KNN Results

We run models on center crops of training images in Places 205 [7] dataset to generate the memory bank $\bar{\mathbf{V}}$. We then run the KNN validation similarly to the ImageNet [3] KNN procedure, which is described in the main paper. The results are shown in Table 2.

Network	KNN
IR with BN - A	36.9
LA - A	**37.5**
IR with BN - V	40.1
LA - V	**41.9**
IR - R18	38.6
LA - R18	**40.3**
IR - R50	41.6
LA - R50	**42.4**

Table 2: KNN results for Places 205 dataset. “A” means “AlexNet”. “V” means VGG16. “R” means “ResNet”.

2.4 Faster RCNN Details

Our Faster RCNN [6] implementations are based on tf-faster-rcnn. We use SGD with momentum of 0.9, batch size 256, and weight decay 0.0001. Learning rate is initialized as 0.001 and dropped by a factor of 10 after 50000 steps. We train the models for 70000 steps. In particular, we set the number of total RoIs for training the region classifier to be 128 to reproduce the original Faster RCNN results, as indicated by [2]. For AlexNet, we fine-tune all layers. For VGG16, we fix “conv1” and “conv2” while fine-tuning others. For ResNet-50, we fix the first convolution layer and the first three blocks while fine-tuning others. Other hyper-parameters are the same as the default settings in tf-faster-rcnn.

2.5 Single-crop Results

Table 3 includes the transfer learning performance on ImageNet and Places 205 using single crop.

3 Other Hyperparameters

There are several other adjustable hyper-parameters in LA training procedure, such as the updating frequency for the clustering results, the parameter k in $\mathcal{N}_k$ for $\mathbf{B}_i$, and whether doing clustering on $\bar{\mathbf{V}}$ or network outputs on center crops of $\mathbf{I}$. In this section, we show results of experiments illustrating the influences of these parameters in Table 4.

Method	conv1	conv2	conv3	conv4	conv5
ImageNet					
IR - A	16.4	28.1	32.4	37.3	38.5
LA - A	14.9	30.1	35.7	39.4	40.2
IR - V	11.2	16.9	25.2	37.3	49.5
LA - V	11.1	19.3	21.2	40.2	55.4
LA - R18	8.4	16.3	27.7	45.0	50.7
LA - R50	9.3	23.2	38.0	48.6	58.8
Places205					
IR - A	19.3	31.1	34.5	37.4	36.8
LA - A	17.1	32.2	36.5	38.3	37.8
IR - V	14.9	21.2	24.6	36.7	43.9
LA - V	17.8	23.2	30.0	42.1	48.1
LA - R18	17.3	22.4	29.9	41.6	44.1
LA - R50	18.3	31.5	39.2	46.3	49.1

Table 3: Top-1 classification accuracy on ImageNet and Places205 with single crop. "A" means "AlexNet". "V" means VGG16. "R" means "ResNet". "IR" means "IR with BN".

References

[1] Mathilde Caron, Piotr Bojanowski, Armand Joulin, and Matthijs Douze. Deep clustering for unsupervised learning of visual features. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 132–149, 2018. 2

[2] Xinlei Chen and Abhinav Gupta. An implementation of faster rcnn with study for region sampling. *arXiv preprint arXiv:1702.02138*, 2017. 2

[3] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. 2009. 2

[4] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 1

[5] Geoffrey Hinton, Li Deng, Dong Yu, George Dahl, Abdel-rahman Mohamed, Navdeep Jaitly, Andrew Senior, Vincent Vanhoucke, Patrick Nguyen, Brian Kingsbury, et al. Deep neural networks for acoustic modeling in speech recognition. *IEEE Signal processing magazine*, 29, 2012. 1

[6] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster R-CNN: Towards real-time object detection with region proposal networks. In *Advances in Neural Information Processing Systems (NIPS)*, 2015. 2

Setting	NN perf.
Baseline	35.7
$k = 2048$	35.4
$k = 8192$	35.8
center_crop	35.8
more_freq	35.7

Table 4: Nearest neighbor validation performances for ResNet-18 trained with different settings. "Baseline" uses $H = 1, m = 10000$, and $k = 4096$. Other settings change one of the hyper-parameters while keeping the others the same. "center_crop" represents the experiment with clustering result acquired on the center crops rather than $\bar{\mathbf{V}}$. "more_freq" represents the experiment with clustering result updated every 1000 steps.

[7] Bolei Zhou, Agata Lapedriza, Jianxiong Xiao, Antonio Torralba, and Aude Oliva. Learning deep features for scene recognition using places database. In *Advances in neural information processing systems*, pages 487–495, 2014. 2