Supplementary Material for: Steering Self-Supervised Feature Learning Beyond Local Pixel Statistics

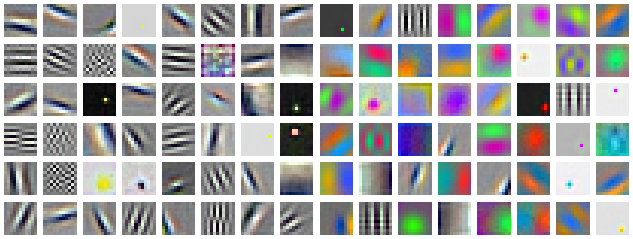

Figure 1: Features learned in the first convolutional layer of an AlexNet trained to recognize image transformations on ImageNet with our method.

1. Implementation Details for Limited Context Inpainting

We provide additional details regarding the implementation of our Limited Context Inpainting (LCI). The network architecture of the inpainter network F is depicted in Table 2. We used a standard autoencoder architecture with leaky-ReLU activations [11] and batch normalization [7]. The architecture of the patch discriminator D is shown in Table 3. We use spectral normalization [12] in all the layers of the discriminator. We feed a pair of real or generated patches as input to the discriminator by concatenating them along the channel dimension. We found this to result in more diverse patch inpaintings and more stable training. This technique was also proposed by [10].

2. Details of the Evaluation Protocol

For the linear classifier experiments on ImageNet and Places we followed the protocol established by [15] and train linear classifiers on fixed features extracted at different layers of the network. Feature maps are spatially resized via average-pooling such that they contain approximately 9K units. Training parameters of the linear classifiers are identical to the prior SotA [3]. Concretely, linear classifiers are trained for 65 epochs using SGD+Momentum with an initial learning rate of 0.1 which we decay to 0.01 after 5 epochs, 0.002 after 25 epochs and finally 0.0004 after 45 epochs.

Table 1: Comparison of test-set accuracy on STL-10 with other published results. Note that the methods do not all use the same network architecture.

Method	**Accuracy**
Dosovitskiy *et al*. [1]	74.2%
Dundar *et al*. *et al*. [2]	74.1%
Hjelm *et al*. [5]	77.0%
Huang *et al*. [6]	76.8%
Jenni & Favaro [8]	80.1%
Ji *et al*. [9]	88.8%
Oyallon *et al*. [13]	87.6%
Swersky *et al*. [14]	70.1%
Zhao *et al*. [16]	74.3%
Ours	**91.8%**

3. ResNet Experiments on STL-10

We performed additional experiments with a more modern network architecture on STL-10. We followed the setup of [9] and trained a ResNet-34 [4] for 200 epochs on the 100K unlabelled training images of STL-10. We then fine-tuned the network for 300 epochs on the 5K labelled training images and evaluate on the 8K test images. The training parameters are the same as in our experiments with AlexNet. We used data augmentation and multi-crop evaluation similar to [9]. Results and a comparison to prior work is shown in Table 1.

4. Additional Qualitative Results

We visualize the filters learned in the first convolutional layer of an AlexNet after our self-supervised pre-training in Figure 1. We provide additional results for nearest neighbor retrieval on the ImageNet validation set in Figure 2. Additionally, we show some examples of LCI transformed images in Figure 3. Note that although the patch-border is in some cases visible, the transformation classifier can not rely on solely detecting these borders, since the examples with autoencoded patches will have similar processing footprints.

Figure 2: Additional results for nearest neighbor retrieval. The left-most column shows the query image. Odd rows: Retrievals with our features. Even rows: Retrievals with features learned using ImageNet labels.

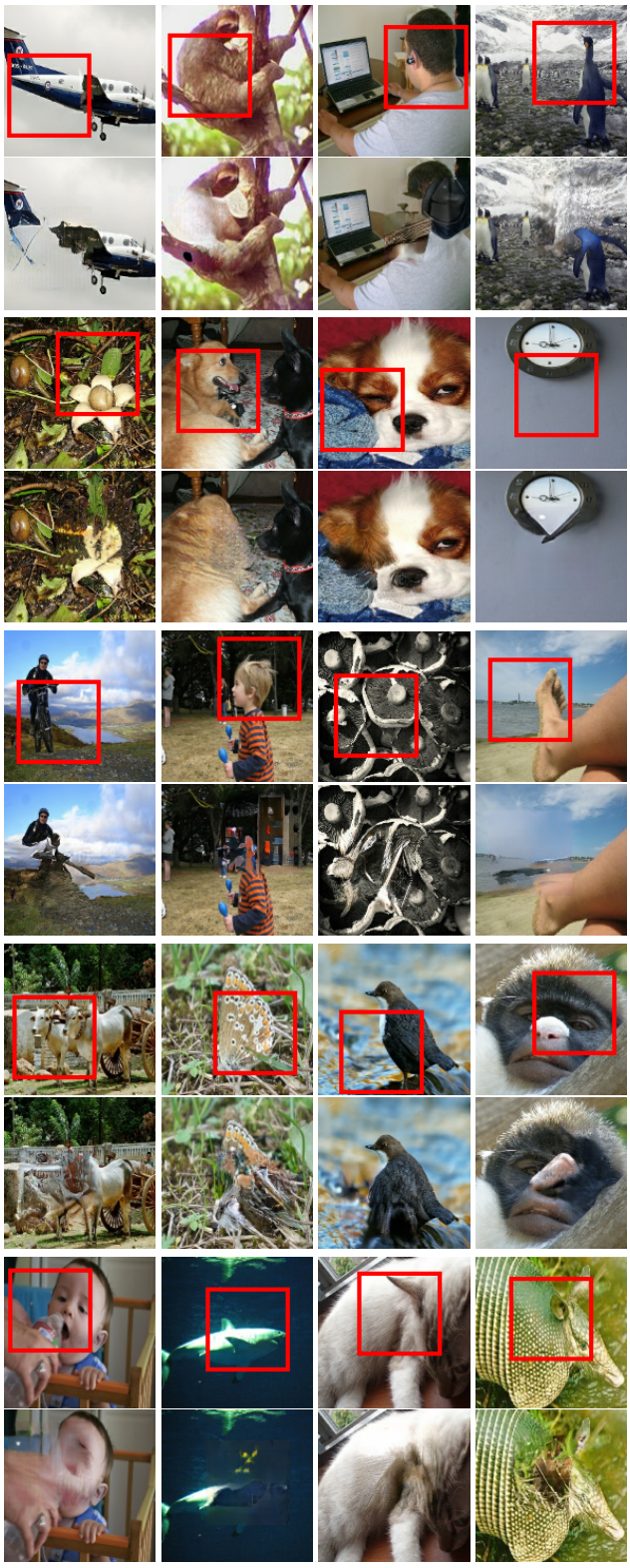

Figure 3: We show examples of images transformed with Limited Context Inpainting (LCI). Odd rows: The original training images with the patch used for LCI indicated in red. Even rows: The images after applying LCI.

Table 2: Network architecture of our inpainter network F used for LCI. The layers in parenthesis are included for pre-training on ImageNet and excluded for the experiments on STL-10 and CelebA.

Inpainter Network F
conv 3×3 stride=1 `leaky-ReLU` 48
conv 4×4 stride=2 BN `leaky-ReLU` 96
conv 4×4 stride=2 BN `leaky-ReLU` 192
(conv 4×4 stride=2 BN `leaky-ReLU` 384)
(deconv 4×4 stride=2 BN `leaky-ReLU` 192)
deconv 4×4 stride=2 BN `leaky-ReLU` 96
deconv 4×4 stride=2 BN `leaky-ReLU` 48
deconv 3×3 stride=1 `tanh` 3

Table 3: Network architecture of our patch discriminator network D used for LCI. The layers in parenthesis are included for pre-training on ImageNet and excluded for the experiments on STL-10 and CelebA.

Patch Discriminator D
conv 3×3 stride=1 SN `leaky-ReLU` 64
conv 4×4 stride=2 SN `leaky-ReLU` 64
conv 3×3 stride=1 SN `leaky-ReLU` 128
conv 4×4 stride=2 SN `leaky-ReLU` 128
conv 3×3 stride=1 SN `leaky-ReLU` 256
(conv 4×4 stride=2 SN `leaky-ReLU` 256)
(conv 3×3 stride=1 SN `leaky-ReLU` 512)
Global 2D Average Pooling
fully-connected SN `linear` 1

References

[1] Alexey Dosovitskiy, Jost Tobias Springenberg, Martin Riedmiller, and Thomas Brox. Discriminative unsupervised feature learning with convolutional neural networks. In *Advances in Neural Information Processing Systems*, pages 766–774, 2014.

[2] Aysegul Dundar, Jonghoon Jin, and Eugenio Culurciello. Convolutional clustering for unsupervised learning. *arXiv preprint arXiv:1511.06241*, 2015.

[3] Zeyu Feng, Chang Xu, and Dacheng Tao. Self-supervised representation learning by rotation feature decoupling. In *The IEEE Conference on Computer Vision and Pattern Recognition*, 2019.

[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.

[5] R Devon Hjelm, Alex Fedorov, Samuel Lavoie-Marchildon, Karan Grewal, Phil Bachman, Adam Trischler, and Yoshua Bengio. Learning deep representations by mutual information estimation and maximization. *arXiv preprint arXiv:1808.06670*, 2018.

[6] Chen Huang, Chen Change Loy, and Xiaoou Tang. Unsupervised learning of discriminative attributes and visual representations. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5175–5184, 2016.

[7] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *International Conference on Machine Learning*, pages 448–456, 2015.

[8] Simon Jenni and Paolo Favaro. Self-supervised feature learning by learning to spot artifacts. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2733–2742, 2018.

[9] Xu Ji, João F Henriques, and Andrea Vedaldi. Invariant information clustering for unsupervised image classification and segmentation. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 9865–9874, 2019.

[10] Zinan Lin, Ashish Khetan, Giulia Fanti, and Sewoong Oh. Pacgan: The power of two samples in generative adversarial networks. In *Advances in Neural Information Processing Systems*, pages 1498–1507, 2018.

[11] Andrew L Maas, Awni Y Hannun, and Andrew Y Ng. Rectifier nonlinearities improve neural network acoustic models. In *Proc. ICML*, volume 30, 2013.

[12] Takeru Miyato, Toshiki Kataoka, Masanori Koyama, and Yuichi Yoshida. Spectral normalization for generative adversarial networks. *arXiv preprint arXiv:1802.05957*, 2018.

[13] Edouard Oyallon, Eugene Belilovsky, and Sergey Zagoruyko. Scaling the scattering transform: Deep hybrid networks. In *Proceedings of the IEEE international conference on computer vision*, pages 5618–5627, 2017.

[14] Kevin Swersky, Jasper Snoek, and Ryan P Adams. Multi-task bayesian optimization. In *Advances in neural information processing systems*, pages 2004–2012, 2013.

[15] Richard Zhang, Phillip Isola, and Alexei A Efros. Colorful image colorization. In *European Conference on Computer Vision*, pages 649–666. Springer, 2016.

[16] Junbo Zhao, Michael Mathieu, Ross Goroshin, and Yann Lecun. Stacked what-where auto-encoders. *arXiv preprint arXiv:1506.02351*, 2015.