Steering Self-Supervised Feature Learning Beyond Local Pixel Statistics

Simon Jenni[1] Hailin Jin[2] Paolo Favaro[1]
University of Bern[1] Adobe Research[2]
{simon.jenni,paolo.favaro}@inf.unibe.ch hljin@adobe.com

Abstract

We introduce a novel principle for self-supervised feature learning based on the discrimination of specific transformations of an image. We argue that the generalization capability of learned features depends on what image neighborhood size is sufficient to discriminate different image transformations: The larger the required neighborhood size and the more global the image statistics that the feature can describe. An accurate description of global image statistics allows to better represent the shape and configuration of objects and their context, which ultimately generalizes better to new tasks such as object classification and detection. This suggests a criterion to choose and design image transformations. Based on this criterion, we introduce a novel image transformation that we call limited context inpainting (LCI). This transformation inpaints an image patch conditioned only on a small rectangular pixel boundary (the limited context). Because of the limited boundary information, the inpainter can learn to match local pixel statistics, but is unlikely to match the global statistics of the image. We claim that the same principle can be used to justify the performance of transformations such as image rotations and warping. Indeed, we demonstrate experimentally that learning to discriminate transformations such as LCI, image warping and rotations, yields features with state of the art generalization capabilities on several datasets such as Pascal VOC, STL-10, CelebA, and ImageNet. Remarkably, our trained features achieve a performance on Places on par with features trained through supervised learning with ImageNet labels.

1. Introduction

The top-performance approaches to solve vision-based tasks, such as object classification, detection and segmentation, are currently based on supervised learning. Unfortunately, these methods achieve a high-performance only through a large amount of labeled data, whose collection is costly and error-prone. Learning through labels may also encounter another fundamental limitation, depending on the training procedure and dataset: It might yield features that describe mostly local statistics, and thus have limited generalization capabilities. An illustration of this issue is shown in Fig. 1. On the bottom row we show images that have been transformed such that local statistics of the corresponding image on the top row are preserved, but global statistics are not. We find experimentally that features pre-trained with ImageNet labels [6] have difficulties in telling real images apart from the transformed ones. This simple test shows that the classification task in ImageNet could be mostly solved by focusing on local image statistics. Such problem might not be noticed when evaluating these features on other tasks and datasets that can be solved based on similar local statistics. However, more general classification settings would certainly expose such a limitation. [16] also pointed out this problem and showed that training supervised models to focus on the global statistics (which they refer to as *shape*) can improve the generalization and the robustness of the learned

Figure 1: **The importance of global image statistics**. Top row: Natural images. Bottom row: Images transformed such that local statistics are preserved while global statistics are significantly altered.[1] An accurate image representation should be able to distinguish these two categories. A linear binary classifier trained to distinguish original versus transformed images on top of `conv5` features pre-trained on ImageNet labels yields an accuracy of 78%. If instead we use features pre-trained with our proposed self-supervised learning task the classifier achieves an accuracy of 85%. Notice that this transformation was not used in the training of our features and that the transformed images were built independently of either feature.

[1]The transformed images are obtained by partitioning an image into a 4×4 grid, by randomly permuting the tiles, and by training a network to inpaint a band of pixels across the tiles through adversarial training [19].

(a) (b) (c) (d)

Figure 2: **Selected image transformations**. Examples of local patches from images that were (a) warped, (b) locally inpainted, (c) rotated or (d) not transformed. The bottom row shows the original images, the middle row shows the corresponding transformed images and the top row shows a detail of the transformed image. By only observing a local patch (top row), is it possible in all of the above cases to tell if and how an image has been transformed or is it instead necessary to observe the whole image (middle row), *i.e.*, the global pixel statistics?

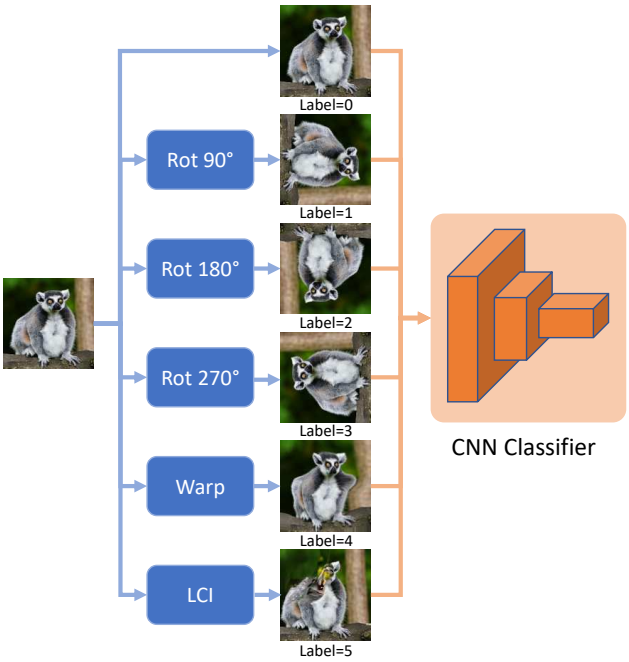


Figure 3: **Learning global statistics**. We propose to learn image representations by training a convolutional neural network to classify image transformations. The transformations are chosen such that local image statistics are preserved while global statistics are distinctly altered.

image representation.

Thus, to address this fundamental shortcoming and to limit the need for human annotation, we propose a novel self-supervised learning (SSL) method. SSL methods learn features without manual labeling and thus they have the potential to better scale their training and leverage large amounts of existing unlabeled data. The training task in our method is to *discriminate global image statistics*. To this end, we transform images in such a way that local statistics are largely unchanged, while global statistics are clearly altered. By doing so, we make sure that the discrimination of such transformations is not possible by working on just local patches, but instead it requires using the whole image. We illustrate this principle in Fig. 2. Incidentally, several existing SSL tasks can be seen as learning from such transformations, *e.g.*, spotting artifacts [25], context prediction [44], rotation prediction [17], and solving jigsaw puzzles [38].

We cast our self-supervised learning approach as the task of discriminating changes in the global image statistics by classifying several image transformations (see Fig. 3). As a novel image transformation we introduce *limited context inpainting* (LCI). LCI selects a random patch from a natural image, substitutes the center with noise (thus, it preserves a small outer boundary of pixels), and trains a network to inpaint a realistic center through adversarial training. While LCI can inpaint a realistic center of the patch so that it seamlessly blends with the preserved boundaries, it is unlikely to provide a meaningful match with the rest of the original image. Hence, this mismatch can only be detected by learning global statistics of the image. Our formulation is also highly scalable and allows to easily incorporate more transformations as additional categories. In fact, we also include the classification of image warping and image rotations (see examples of such transformations in Fig. 2). An illustration of the proposed training scheme is shown in Fig. 3.

Contributions. Our proposed method has the following original contributions: 1) We introduce a novel self-supervised learning principle based on image transformations that can be detected only through global observations; 2) We introduce a novel transformation according to this principle and demonstrate experimentally its impact on feature learning; 3) We formulate the method so that it can easily scale with additional transformations; 4) Our proposed method achieves state of the art performance in transfer learning on several data sets; in particular, for the first time, we show that our trained features when transferred to Places achieve a performance on par with features trained through supervised learning with ImageNet labels. Code is available at `https://sjenni.github.io/LCI`.

2. Prior Work

Self-supervised Learning. Self-supervised learning is a feature learning method that avoids the use of data labels by introducing an artificial task. Examples of tasks defined on images are to find: the spatial configuration of parts [8, 38, 37], the color of a grayscale image [55, 56, 29], the image patch given its context [44], the image orientation [17], the artifacts introduced by a corruption process [25], the image instance up to data jittering [12, 51, 52], contrastive predictive coding [41, 20] or pseudo-labels obtained from a clustering process [40, 4, 60]. Self-supervised learning has also been applied to other data domains such as video [50, 43, 48, 36] and audio [42, 57, 15].

Several self-supervised tasks can be seen as the prediction of some form of image transformation applied to an image. Gidaris *et al*. [17] for example predict the number of 90° rotations applied to an image. Jenni and Favaro [25] predict the presence and position of artifacts introduced by a corruption process. Doersch *et al*. [8] predict transformations concerning image patches by predicting their relative location. Noroozi and Favaro [38] extend this idea to multiple patches by solving jigsaw puzzles. Recently Zhang *et al*. [54] proposed to predict the parameters of a relative projective transformation between two images using a Siamese architecture. In our work, we show that by predicting a combination of novel and previously explored image transformations we can form new and more challenging learning tasks that learn better features.

Some works have explored the combination of different self-supervised tasks via multi-task learning [46, 9]. Recently, Feng *et al*. [14] showed that a combination of the rotation prediction task by Gidaris *et al*. [17] with the instance recognition task by Wu *et al*. [51] achieve state-of-the-art results in transfer experiments. They do so by splitting the penultimate feature vector into two parts: One to predict the transformation and a second transformation agnostic part, used to discriminate between different training images. Note that our work is orthogonal to these approaches and thus it could be integrated in such multi-task formulations and would likely lead to further improvements.

Because in our LCI transformation we build an inpainting network through adversarial training, we briefly discuss works that exploit similar techniques.

Adversarial Feature Learning. Generative Adversarial Networks (GANs) [19] have been used for the purpose of representation learning in several works. Radford *et al*. [45] first showed that a convolutional discriminator can learn reasonably good features. Donahue *et al*. [10, 11] learn features by training an encoder to produce the inverse mapping of the generator. Pathak *et al*. [44] use an adversarial loss to train an autoencoder for inpainting. They use the trained encoder as a feature extractor. Denton *et al*. [7] also perform inpainting, but instead transfer the discriminator features. The work by Jenni and Favaro [25] has some similarity to our LCI transformation. They generate image artifacts by erasing and locally repairing features of an autoencoder. Our limited context inpainting is different from these methods in two important ways. First, we more strongly limit the context of the inpainter and put the inpainted patch back into a larger context to produce unrealistic global image statistics. Second, a separate patch discriminator allows stable adversarial training independent of the feature learning component.

Recognizing Image Manipulations. Many works have considered the detection of image manipulations in the context of image forensics [22, 49, 59, 2]. For example, Wang *et al*. [49] predict subtle face image manipulations based on local warping. Zhou *et al*. [59] detect image tampering generated using semantic masks. Transformations in these cases are usually subtle and do not change the global image statistics in a predictable way (images are manipulated to appear realistic). The aim is therefore antithetical to ours.

3. Learning Features by Discriminating Global Image Transformations

Our aim is to learn image representations without human annotation by recognizing variations in global image statistics. We do so by distinguishing between natural images and images that underwent several different image transformations. Our principle is to choose image transformations that: 1) Preserve local pixel statistics (*e.g*., texture), but alter the global image statistics of an image and 2) Can be recognized from a single transformed example in most cases. In this paper we choose the following transformations: limited context inpainting, warping, rotations and the identity. These transformations will be introduced in detail in the next sections.

Formally, given a set of unlabelled training images $\{x_i\}_{i=1,\dots,N}$ and a set of image transformations $\{T_j\}_{j=0,\dots,K}$, we train a classifier C to predict the transformation-label j given a transformed example $T_j \circ x_i$. In our case we set $K = 5$. We include the identity (no-transformation) case by letting $T_0 \circ x \doteq x$. We train the network C by minimizing the following self-supervised objective

$$\mathcal{L}_{\text{SSL}}(T_0, \dots, T_5) \doteq \min_C \frac{1}{6N} \sum_{i=1}^{N} \sum_{y=0}^{5} \ell_{\text{cls}}\Big(C\big(T_y \circ x_i\big), y\Big), \tag{1}$$

where ℓ_{cls} is the standard cross-entropy loss for a multi-class classification problem.

3.1. Limited Context Inpainting

The first transformation that we propose to use in eq. (1) is based on the Limited Context Inpainting (LCI). The aim

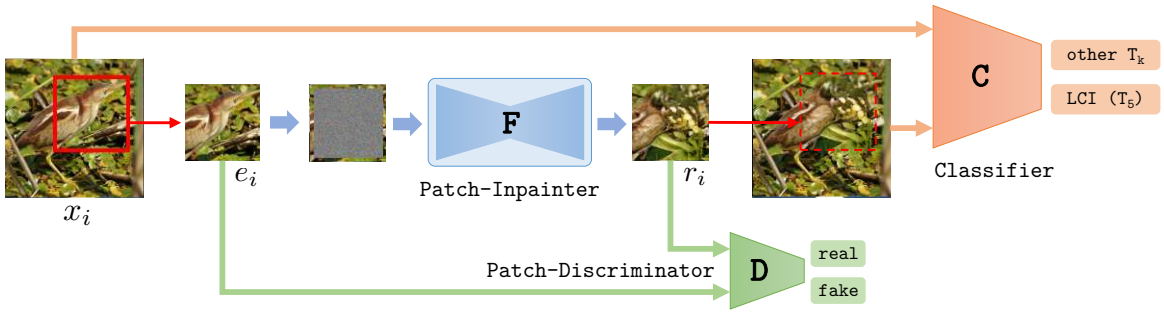


Figure 4: **Training of the Limited Context Inpainting (LCI) network**. A random patch is extracted from a training image x and all but a thin border of pixels is replaced by random noise. The inpainter network F fills the patch with realistic textures conditioned on the remaining border pixels. The resulting patch is replaced back into the original image, thus generating an image with natural local statistics, but unnatural global statistics.

of LCI is to modify images only locally, *i.e.*, at the scale of image patches. We train an inpainter network F conditioned only on a thin border of pixels of the patch (see Fig. 4). The inpainted patch should be realistic on its own and blend in at the boundary with the surrounding image, but should not meaningfully match the content of the whole image (see an example in Fig. 2 (b)). The inpainter F is trained using adversarial training against a patch discriminator D (which ensures that we match the local statistics) as well as the transformation classifier C. The patch to be inpainted is randomly selected at a uniformly sampled location $\Delta \in \Omega$, where Ω is the image domain. Then, $\mathcal{W}_\Delta \subset \Omega$ is a square region of pixels around Δ. We define e_i as the original patch of pixels at $\mathcal{W}_\Delta$ and r_i as the corresponding inpainted patch

$$e_i(p - \Delta) \doteq x_i(p), \quad \forall p \in \mathcal{W}_\Delta \tag{2}$$

$$r_i \doteq F(e_i \odot (1 - m) + z \odot m) \tag{3}$$

with m a mask that is 1 in the center of the patch and 0 at the boundary (2 to 4 pixels in our baseline), $z \sim \mathcal{N}(0, I)$ is a zero-mean Gaussian noise and $\odot$ denotes the Hadamard (pixel-to-pixel) product. The LCI transformation T_5 is then defined as

$$(T_5 \circ x_i)(p) \doteq \begin{cases} x_i(p) & \text{if } p \notin \mathcal{W}_\Delta \\ r_i(p - \Delta) & \text{if } p \in \mathcal{W}_\Delta. \end{cases} \tag{4}$$

Finally, to train the inpainter F we minimize the cost

$$\mathcal{L}_{\text{inp}} = \frac{1}{N} \sum_{i=1}^{N} \ell_{\text{GAN}}(r_i, e_i) + \lambda_{\text{border}} \left|(r_i - e_i) \circ (1 - m)\right|^2 - \mathcal{L}_{\text{SSL}}(T_0, \ldots, T_5), \tag{5}$$

where $\lambda_{\text{border}} = 50$ is a tuning parameter to regulate the importance of autoencoding the input boundary, and $\ell_{\text{GAN}}(\cdot, \cdot)$ is the hinge loss for adversarial training [30], which also includes the maximization in the discriminator D.

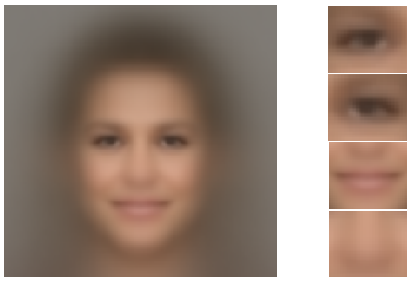
Remark. In contrast to prior SSL methods [25, 44, 7] , here we do not take the features from the networks that we used to learn the transformation (*e.g.*, D or F). Instead, here we take features from a separate classifier C that has only a partial role in the training of F. This separation has several advantages: 1) A separate tuning of training parameters is possible, 2) GAN tricks can be applied without affecting the classifier C, (3) GAN training can be stable even when the classifier wins ($\mathcal{L}_{\text{SSL}}$ saturates w.r.t. F).

3.2. Random Warping

In addition to the LCI, which is a local image transformation, we consider random global warping as our T_4 transformation. A warping is a smooth deformation of the image coordinates defined by n pixel coordinates $\{(u_i, v_i)\}_{i=1,\ldots,n}$, which act as control points. We place the control points on an uniform grid of the image domain and then randomly offset each control point by sampling the shifts from a rectangular range $[-d, d] \times [-d, d]$, where d is typically $1/10$-th of the image size. The dense flow field for warping is then computed by interpolating between the offsets at the control points using a polyharmonic spline [13]. Warping affects the local image statistics only minimally: In general, it is difficult to distinguish a warped patch from a patch undergoing a change in perspective. Therefore, the classifier needs to learn global image statistics to detect image warping.

3.3. Image Rotations

Finally, we consider as T_1, T_2, and T_3 image rotations of 90°, 180°, and 270°respectively. This choice is inspired by Gidaris *et al.* [17] who proposed RotNet, a network to predict image rotations by multiples of 90°. This was shown to be a simple yet effective SSL pretext task. These transformations are predictable because the photographer bias introduces a canonical reference orientation for many natural images. They also require global statistics as local patches



(a) (b)

Figure 5: **Image statistics on CelebA**. (a) The mean image obtained from 8000 samples from CelebA. (b) Four local patches extracted from the mean image. Because these patterns appear always with the same orientation in the dataset, it is possible to distinguish rotated images by using only these local statistics.

of rotated images often do not indicate the orientation of the image, because similar patches can be found in the untransformed dataset.

Remark. There exist, however, several settings in which the prediction of image rotations does not result in good features. Many natural images for example do not have a canonical image orientation. Thus, in these cases the prediction of image rotations is an ill-posed task. There also exist entire data domains of interest, where the image orientation is ambiguous, such as satellite and cell imaging datasets. Even when a clear upright image orientation exists, this method alone can lead to non-optimal feature learning. As an example, we show that the prediction of image rotations on CelebA [31], a dataset of face images, leads to significantly worse features than can be learned through the prediction of other transformations (see Table 3). The main reason behind this limitation is that local patches can be found in the dataset always with the same orientation (see Fig. 5). For instance, the classifier can easily distinguish rotated faces by simply detecting one eye or the mouth.

3.4. Preventing Degenerate Learning

As was observed by Doersch *et al*. [8], networks trained to solve a self-supervised task might do so by using very local statistics (*e.g*., localization by detecting the chromatic aberration). Such solutions are called *shortcuts* and are a form of degenerate learning as they yield features with poor generalization capabilities. When introducing artificial tasks, such as the discrimination of several image transformations, it is important to make sure that the trained network cannot exploit (local) artifacts introduced by the transformations to solve the task. For example, the classifier could learn to recognize processing artifacts of the inpainter F in order to recognize LCI transformed images. Although adversarial training should help to prevent this behavior, we find experimentally that it is not sufficient on its own. To further prevent such failure cases, we also train the network F to autoencode image patches by modifying the loss $\mathcal{L}_{\text{inp}}$ in eq. (5) as $\mathcal{L}_{\text{inp,AE}} = \mathcal{L}_{\text{inp}} + \lambda_{\text{AE}} \frac{1}{N} \sum_{i=1}^{N} |F(e_i) - e_i|^2$, where $\lambda_{\text{AE}} = 50$ is a tuning parameter to regulate the importance of autoencoding image patches. We create also artificial untransformed images by substituting a random patch with its autoencoded version. In each mini-batch to the classifier we replace half of the untransformed images with these patch-autoencoded images. In this manner the classifier will not focus on the small artifacts (which could even be not visible to the naked eye) as a way to discriminate the transformations. During training we also replace half of the original images in a minibatch with these patch-autoencoded images before applying the rotation.

4. On the Choice of Transformations

Our goal is to learn features by discriminating images undergoing different transformations. We pointed out that this approach should use transformations that can be distinguished only by observing large regions of pixels, and is scalable, *i.e*., it can be further refined by including more transformations. In this section, we would like to make these two aspects clearer.

Determining suitable transformations. We find that the choice of what transformations to use depends on the data distribution. An example of such dependency in the case of RotNet on CelebA is shown in Fig. 5. Intuitively, *an ideal transformation is such that any transformed local patch should be found in the original dataset, but any transformed global patch should not be found in the dataset*. This is also the key idea behind the design of LCI.

Introducing additional transformations. As we will show in the Experiments section, adding more transformations (as specified above) can improve the performance. An important aspect is that the classifier must be able to distinguish the different transformations. Otherwise, its task is ambiguous and can lead to degenerate learning. Put in simple terms, *a transformed global patch should be different from any other global patch (including itself) transformed with a different transformation*. We verify that our chosen transformations satisfy this principle, as LCI and image warping cannot produce rotated images and warping is a global deformation, while LCI is a local one.

5. Experiments

We perform an extensive experimental evaluation of our formulation on several established unsupervised feature learning benchmarks. For a fair comparison with prior work we implement the transformation classifier C with a standard AlexNet architecture [28]. Following prior work, we remove the local response normalization layers and add batch normalization [23] to all layers except for the fi-

Table 1: Ablation experiments for different design choices of Limited Context Inpainting (LCI) on STL-10 [5]. We pre-train an AlexNet to predict if an image has been transformed with LCI or not and transfer the frozen `conv5` features for linear classification.

Ablation	**Accuracy**
(a) 32×32 patches	61.2%
(b) 40×40 patches	70.6%
(c) 56×56 patches	75.1%
(d) Pre-trained and frozen F	63.7%
(e) No adversarial loss w.r.t. C	68.0%
(f) No patch autoencoding	69.5%
Baseline (48×48 patches)	76.2%

nal one. No other modifications to the original architecture were made (we preserve the two-stream architecture). For experiments on lower resolution images we remove the max-pooling layer after `conv5` and use SAME padding throughout the network. The standard data-augmentation strategies (random cropping and horizontal flipping) were used. Self-supervised pre-training of the classifier was performed using the AdamW optimizer [34] with parameters $\beta_1 = 0.5$, $\beta_2 = 0.99$ and a weight decay of 10^{-4}. We decayed the learning rate from $3 \cdot 10^{-4}$ to $3 \cdot 10^{-7}$ over the course of training using cosine annealing [33]. The training of the inpainter network F and patch-discriminator D was done using the Adam optimizer [26] with a fixed learning rate of $2 \cdot 10^{-4}$ and $\beta_1 = 0.5$. The size of the patch boundary is set to 2 pixels in experiments on STL-10 and CelebA. On ImageNet we use a 4 pixel boundary. Details for the network architectures and additional results are provided in the supplementary material.

5.1. Ablation Experiments

Limited Context Inpainting. We perform ablation experiments on STL-10 [5] to validate several design choices for the joint inpainter and classifier training. We also illustrate the effect of the patch-size on the performance of the learned features. We pre-train the transformation classifier for 200 epochs on 64×64 crops of the unlabelled training set. The mini-batch size was set to 64. We then transfer the frozen `conv5` features by training a linear classifier for 500 epochs on randomly cropped 96×96 images of the small labelled training set. Only LCI was used as transformation in these experiments. The results of the following ablations are reported in Table 1:

(a)-(c) Varying Patch-Size: We vary the size of the inpainted patches. We observe that small patches lead to a significant drop in feature performance. Smaller patches are easy to inpaint and the results often do not alter the global image statistics;

Table 2: We report the test set accuracy of linear classifiers trained on frozen features for models trained to predict different combinations of image transformations on STL-10.

Initialization	`conv1`	`conv2`	`conv3`	`conv4`	`conv5`
Random	48.4%	53.3%	51.1%	48.7%	47.9%
Warp	57.2%	64.2%	62.8%	58.8%	55.3%
LCI	58.8%	67.2%	67.4%	68.1%	68.0%
Rot	58.2%	67.3%	69.3%	69.9%	70.1%
Warp + LCI	**59.3%**	68.1%	69.5%	68.5%	67.2%
Rot + Warp	57.4%	69.2%	70.7%	70.5%	70.6%
Rot + LCI	58.5%	69.2%	71.3%	72.8%	72.3%
Rot + Warp + LCI	59.2%	**69.7%**	**71.9%**	**73.1%**	**73.7%**

Table 3: We report the average precision of linear classifiers trained to predict facial attributes on frozen features of models trained to predict different combinations of image transformations on CelebA.

Initialization	`conv1`	`conv2`	`conv3`	`conv4`	`conv5`
Random	68.9%	70.1%	66.7%	65.3%	63.2%
Warp	71.7%	73.4%	71.2%	68.8%	64.3%
LCI	71.3%	73.0%	72.0%	71.1%	68.0%
Rot	70.3%	70.9%	67.8%	65.6%	62.1%
Warp + LCI	**72.0%**	73.9%	73.3%	72.1%	69.0%
Rot + Warp	71.6%	73.6%	72.0%	70.1%	66.4%
Rot + LCI	71.3%	72.7%	71.9%	70.8%	66.7%
Rot + Warp + LCI	71.8%	**74.0%**	**73.5%**	**72.5%**	**69.2%**

(d)-(f) Preventing Shortcuts: Following sec. 3.4, we show how adversarial training of F is necessary to achieve a good performance by removing the feedback of both D and C in (d) and only C in (e). We also demonstrate the importance of adding autoencoded patches to the non-transformed images in (f);

Combination of Image Transformations. We perform additional ablation experiments on STL-10 and CelebA [31] where C is trained to predict different combinations of image transformations. These experiments illustrate how our formulation can scale with the number of considered image transformations and how the effectiveness of transformations can depend on the data domain.

We pre-train the AlexNet to predict image transformations for 200 epochs on 64×64 crops on STL-10 and for 100 epochs on 96×96 crops on CelebA using the standard data augmentations. For transfer we train linear classifiers on top of the frozen convolutional features (without resizing of the feature-maps) to predict the 10 object categories in the case of STL-10 and to predict the 40 face attributes in the case of CelebA. Transfer learning is performed for 700 epochs on 64×64 crops in the case of STL-10 and for 100

Table 4: Transfer learning results for classification, detection and segmentation on PASCAL compared to state-of-the-art feature learning methods (* use a bigger AlexNet).

Model	[Ref]	Classification (mAP)	Detection (mAP)	Segmentation (mIoU)
Krizhevsky *et al*. [28]	[55]	79.9%	59.1%	48.0%
Random	[44]	53.3%	43.4%	19.8%
Agrawal *et al*. [1]	[10]	54.2%	43.9%	-
Bojanowski *et al*. [3]	[3]	65.3%	49.4%	-
Donahue *et al*. [10]	[10]	60.1%	46.9%	35.2%
Feng *et al*. [14]	[14]	74.3%	**57.5%**	**45.3%**
Gidaris *et al*. [17]	[17]	73.0%	54.4%	39.1%
Jayaraman & Grauman [24]	[24]	-	41.7%	-
Jenni & Favaro [25]	[25]	69.8%	52.5%	38.1%
Krähenbühl *et al*. [27]	[27]	56.6%	45.6%	32.6%
Larsson *et al*. [29]	[29]	65.9%	-	38.0%
Noroozi & Favaro [38]	[38]	67.6%	53.2%	37.6%
Noroozi *et al*. [39]	[39]	67.7%	51.4%	36.6%
Noroozi *et al*. [40]	[40]	72.5%	56.5%	42.6%
Mahendran *et al*. [35]	[35]	64.4%	50.3%	41.4%
Mundhenk *et al*. [37]	[37]	69.6%	55.8%	41.4%
Owens *et al*. [42]	[42]	61.3%	44.0%	-
Pathak *et al*. [44]	[44]	56.5%	44.5%	29.7%
Pathak *et al*. [43]	[43]	61.0%	52.2%	-
Wang & Gupta [50]	[27]	63.1%	47.4%	-
Zhan *et al*. [53]	[53]	-	-	44.5%
Zhang *et al*. [55]	[55]	65.9%	46.9%	35.6%
Zhang *et al*. [56]	[56]	67.1%	46.7%	36.0%
Doersch *et al*. [8]*	[10]	65.3%	51.1%	-
Caron *et al*. [4]*	[4]	73.7%	55.4%	45.1
Ours	-	**74.5%**	56.8%	44.4

Table 5: Validation set accuracy on ImageNet with linear classifiers trained on frozen convolutional layers. † indicates multi-crop evaluation and * use a bigger AlexNet.

Model\Layer	conv1	conv2	conv3	conv4	conv5
ImageNet Labels	19.3%	36.3%	44.2%	48.3%	50.5%
Random	11.6%	17.1%	16.9%	16.3%	14.1%
Donahue *et al*. [10]	17.7%	24.5%	31.0%	29.9%	28.0%
Feng *et al*. [14]	19.3%	33.3%	**40.8%**	41.8%	**44.3%**
Gidaris *et al*. [17]	18.8%	31.7%	38.7%	38.2%	36.5%
Huang *et al*. [21]	15.6%	27.0%	35.9%	39.7%	37.9%
Jenni & Favaro [25]	19.5%	33.3%	37.9%	38.9%	34.9%
Noroozi & Favaro [38]	18.2%	28.8%	34.0%	33.9%	27.1%
Noroozi *et al*. [39]	18.0%	30.6%	34.3%	32.5%	25.7%
Noroozi *et al*. [40]	19.2%	32.0%	37.3%	37.1%	34.6%
Tian *et al*. [47]	18.4%	33.5%	38.1%	40.4%	42.6%
Wu *et al*. [51]	16.8%	26.5%	31.8%	34.1%	35.6%
Zhang *et al*. [55]	13.1%	24.8%	31.0%	32.6%	31.8%
Zhang *et al*. [56]	17.7%	29.3%	35.4%	35.2%	32.8%
Zhang *et al*. [54]	19.2%	32.8%	40.6%	39.7%	37.7%
Doersch *et al*. [8]*	16.2%	23.3%	30.2%	31.7%	29.6%
Caron *et al*. [4]*	12.9%	29.2%	38.2%	39.8%	36.1%
Zhuang *et al*. [60]*†	18.7%	32.7%	38.1%	42.3%	42.4%
Ours	**20.8%**	**34.5%**	40.2%	**43.1%**	41.4%
Ours†	22.0%	36.4%	42.4%	45.4%	44.4%

Table 6: Validation set accuracy on Places with linear classifiers trained on frozen convolutional layers. † indicates multi-crop evaluation and * the use of a bigger AlexNet.

Model\Layer	conv1	conv2	conv3	conv4	conv5
Places Labels	22.1%	35.1%	40.2%	43.3%	44.6%
ImageNet Labels	22.7%	34.8%	38.4%	39.4%	38.7%
Random	15.7%	20.3%	19.8%	19.1%	17.5%
Donahue *et al*. [10]	22.0%	28.7%	31.8%	31.3%	29.7%
Feng *et al*. [14]	22.9%	32.4%	36.6%	37.3%	**38.6%**
Gidaris *et al*. [17]	21.5%	31.0%	35.1%	34.6%	33.7%
Jenni & Favaro [25]	23.3%	**34.3%**	36.9%	37.3%	34.4%
Noroozi & Favaro [38]	23.0%	31.9%	35.0%	34.2%	29.3%
Noroozi *et al*. [39]	23.3%	33.9%	36.3%	34.7%	29.6%
Noroozi *et al*. [40]	22.9%	34.2%	37.5%	37.1%	34.4%
Owens *et al*. [42]	19.9%	29.3%	32.1%	28.8%	29.8%
Pathak *et al*. [44]	18.2%	23.2%	23.4%	21.9%	18.4%
Wu *et al*. [51]	18.8%	24.3%	31.9%	34.5%	33.6%
Zhang *et al*. [55]	16.0%	25.7%	29.6%	30.3%	29.7%
Zhang *et al*. [56]	21.3%	30.7%	34.0%	34.1%	32.5%
Zhang *et al*. [54]	22.1%	32.9%	37.1%	36.2%	34.7%
Doersch *et al*. [8]*	19.7%	26.7%	31.9%	32.7%	30.9%
Caron *et al*. [4]*	18.6%	30.8%	37.0%	37.5%	33.1%
Zhuang *et al*. [60]*†	18.7%	32.7%	38.2%	40.3%	39.5%
Ours	**24.1%**	33.3%	**37.9%**	**39.5%**	37.7%
Ours†	25.0%	34.8%	39.7%	41.1%	39.4%

epochs on 96×96 crops in the case of CelebA. We report results for STL-10 in Table 2 and for CelebA in Table 3.

We can observe that the discrimination of a larger number of image transformations generally leads to better feature performance on both datasets. When considering each of the transformations in isolation we see that not all of them generalize equally well to different data domains. Rotation prediction especially performs significantly worse on CelebA than on STL-10. The performance of LCI on the other hand is good on both datasets.

5.2. Unsupervised Feature Learning Benchmarks

We compare our proposed model to state-of-the-art methods on the established feature learning benchmarks. We pre-train the transformation classifier for 200 epochs on the ImageNet training set. Images were randomly cropped to 128×128 and the last max-pooling layer was removed during pre-training to preserve the size of the feature map before the fully-connected layers. We used a batch-size of 96 and trained on 4 GPUs.

Pascal VOC. We finetune our transformation classifier features for multi-label classification, object detection and semantic segmentation on the Pascal VOC dataset. We follow the established experimental setup and use the framework provided by Krähenbühl *et al*. [27] for multilabel classification, the Fast-RCNN [18] framework for detection and the FCN [32] framework for semantic segmentation. We absorb the batch-normalization parameters into the param-

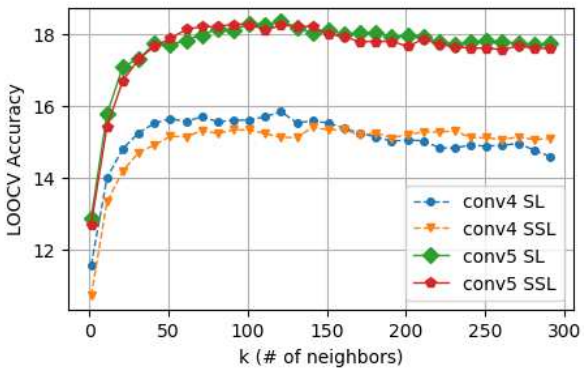


Figure 6: We report leave-one-out cross validation (LOOCV) accuracy for k-nearest neighbor classifiers on the Places validation set. We compare the performance of our self-supervised transformation classifier against features of a supervised network for different values of k. Both networks were pre-trained on ImageNet.

eters of the associated layers in the AlexNet and apply the data-dependent rescaling by Krähenbühl *et al*. [27], as is common practice. The results of these transfer learning experiments are reported in Table 4. We achieve state-of-the-art performance in classification and competitive results for detection and segmentation.

Linear Classifier Experiments on ImageNet and Places. To measure the quality of our self-supervised learning task we use the transformation classifier as a fixed feature extractor and train linear classifiers on top of each convolutional layer. These experiments are performed both on ImageNet (the dataset used for pre-training) and Places [58] (to measure how well the features generalize to new data). We follow the same setup as the state-of-the-art methods and report the accuracy achieved on a single crop. Results for ImageNet are shown in Table 5 and for Places in Table 6. Our learned features achieve state-of-the-art performance for `conv1`, `conv2` and `conv4` on ImageNet. On Places we achieve the best results on `conv1`, `conv3` and `conv4`. Our results on `conv4` in particular are the best overall and even slightly surpass the performance of an AlexNet trained on ImageNet using supervision.

Nearest Neighbor Evaluation. Features learned in deep CNNs through supervised learning tend to distribute so that their Euclidean distance relates closely to the semantic *visual similarity* of the images they correspond to. We want to see if also our SSL features enjoy the same property. Thus, we compute the nearest neighbors of our SSL and of SL features in `conv5` features space on the validation set of ImageNet. Results are shown in Fig. 7. We also show a quantitative comparison of k-nearest neighbor classification on the Places validation set in Figure 6. We report the leave-one-out cross validation (LOOCV) accuracy for different values of k. This can be done efficiently by computing $(k+1)$-nearest neighbors using the complete dataset and by excluding the closest neighbor for each query. The concatenation of features from five 128×128 crops (extracted at the resolution the networks were trained on) is used for nearest neighbors. The features are standardized and cosine similarity is used for nearest neighbor computation.

Figure 7: Comparison of 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. Nearest neighbors were computed on the validation set of ImageNet with `conv5` features using cosine similarity.

6. Conclusions

We introduced the self-supervised feature learning task of discriminating natural images from images transformed through local inpainting (LCI), image warping and rotations, based on the principle that trained features generalize better when their task requires detecting global natural image statistics. This principle is supported by substantial experimental evaluation: Trained features achieve SotA performance on several transfer learning benchmarks (Pascal VOC, STL-10, CelebA, and ImageNet) and even slightly outperform supervised training on Places.

Acknowledgements. This work was supported by the Swiss National Science Foundation (SNSF) grant number 200021_169622 and an Adobe award.

References

[1] Pulkit Agrawal, Joao Carreira, and Jitendra Malik. Learning to see by moving. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 37–45, 2015.

[2] Jawadul H Bappy, Amit K Roy-Chowdhury, Jason Bunk, Lakshmanan Nataraj, and BS Manjunath. Exploiting spatial structure for localizing manipulated image regions. In *Proceedings of the IEEE international conference on computer vision*, pages 4970–4979, 2017.

[3] Piotr Bojanowski and Armand Joulin. Unsupervised learning by predicting noise. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 517–526, 2017.

[4] 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*, pages 132–149, 2018.

[5] Adam Coates, Andrew Ng, and Honglak Lee. An analysis of single-layer networks in unsupervised feature learning. In *Proceedings of the fourteenth international conference on artificial intelligence and statistics*, pages 215–223, 2011.

[6] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 248–255, 2009.

[7] Emily Denton, Sam Gross, and Rob Fergus. Semi-supervised learning with context-conditional generative adversarial networks. *arXiv preprint arXiv:1611.06430*, 2016.

[8] Carl Doersch, Abhinav Gupta, and Alexei A Efros. Unsupervised visual representation learning by context prediction. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1422–1430, 2015.

[9] Carl Doersch and Andrew Zisserman. Multi-task self-supervised visual learning. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2051–2060, 2017.

[10] Jeff Donahue, Philipp Krähenbühl, and Trevor Darrell. Adversarial feature learning. *arXiv preprint arXiv:1605.09782*, 2016.

[11] Jeff Donahue and Karen Simonyan. Large scale adversarial representation learning. *arXiv preprint arXiv:1907.02544*, 2019.

[12] 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.

[13] Jean Duchon. Splines minimizing rotation-invariant semi-norms in sobolev spaces. In *Constructive theory of functions of several variables*, pages 85–100. Springer, 1977.

[14] 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.

[15] Ruohan Gao, Rogerio Feris, and Kristen Grauman. Learning to separate object sounds by watching unlabeled video. In *Proceedings of the European Conference on Computer Vision*, pages 35–53, 2018.

[16] Robert Geirhos, Patricia Rubisch, Claudio Michaelis, Matthias Bethge, Felix A Wichmann, and Wieland Brendel. Imagenet-trained cnns are biased towards texture; increasing shape bias improves accuracy and robustness. *arXiv preprint arXiv:1811.12231*, 2018.

[17] Spyros Gidaris, Praveer Singh, and Nikos Komodakis. Unsupervised representation learning by predicting image rotations. *arXiv preprint arXiv:1803.07728*, 2018.

[18] Ross Girshick. Fast r-cnn. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1440–1448, 2015.

[19] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in Neural Information Processing Systems*, pages 2672–2680, 2014.

[20] Olivier J Hénaff, Ali Razavi, Carl Doersch, SM Eslami, and Aaron van den Oord. Data-efficient image recognition with contrastive predictive coding. *arXiv preprint arXiv:1905.09272*, 2019.

[21] Jiabo Huang, Qi Dong, Shaogang Gong, and Xiatian Zhu. Unsupervised deep learning by neighbourhood discovery. *arXiv preprint arXiv:1904.11567*, 2019.

[22] Minyoung Huh, Andrew Liu, Andrew Owens, and Alexei A Efros. Fighting fake news: Image splice detection via learned self-consistency. In *Proceedings of the European Conference on Computer Vision*, pages 101–117, 2018.

[23] 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.

[24] Dinesh Jayaraman and Kristen Grauman. Learning image representations tied to ego-motion. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1413–1421, 2015.

[25] 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.

[26] Diederik Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[27] Philipp Krähenbühl, Carl Doersch, Jeff Donahue, and Trevor Darrell. Data-dependent initializations of convolutional neural networks. *arXiv preprint arXiv:1511.06856*, 2015.

[28] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012.

[29] Gustav Larsson, Michael Maire, and Gregory Shakhnarovich. Colorization as a proxy task for visual understanding. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6874–6883, 2017.

[30] Jae Hyun Lim and Jong Chul Ye. Geometric gan. *arXiv preprint arXiv:1705.02894*, 2017.

[31] Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Deep learning face attributes in the wild. In *Proceedings of International Conference on Computer Vision*, 2015.

[32] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3431–3440, 2015.

[33] Ilya Loshchilov and Frank Hutter. Sgdr: Stochastic gradient descent with warm restarts. *arXiv preprint arXiv:1608.03983*, 2016.

[34] Ilya Loshchilov and Frank Hutter. Fixing weight decay regularization in adam. *arXiv preprint arXiv:1711.05101*, 2017.

[35] Aravindh Mahendran, James Thewlis, and Andrea Vedaldi. Cross pixel optical-flow similarity for self-supervised learning. In *Asian Conference on Computer Vision*, pages 99–116. Springer, 2018.

[36] Ishan Misra, C Lawrence Zitnick, and Martial Hebert. Shuffle and learn: unsupervised learning using temporal order verification. In *European Conference on Computer Vision*, pages 527–544. Springer, 2016.

[37] Mundhenk et al. Improvements to context based self-supervised learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018.

[38] Mehdi Noroozi and Paolo Favaro. Unsupervised learning of visual representations by solving jigsaw puzzles. In *European Conference on Computer Vision*, pages 69–84. Springer, 2016.

[39] Mehdi Noroozi, Hamed Pirsiavash, and Paolo Favaro. Representation learning by learning to count. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 5898–5906, 2017.

[40] Mehdi Noroozi, Ananth Vinjimoor, Paolo Favaro, and Hamed Pirsiavash. Boosting self-supervised learning via knowledge transfer. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 9359–9367, 2018.

[41] Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018.

[42] Andrew Owens, Jiajun Wu, Josh H McDermott, William T Freeman, and Antonio Torralba. Ambient sound provides supervision for visual learning. In *European Conference on Computer Vision*, pages 801–816. Springer, 2016.

[43] Deepak Pathak, Ross Girshick, Piotr Dollár, Trevor Darrell, and Bharath Hariharan. Learning features by watching objects move. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017.

[44] Deepak Pathak, Philipp Krahenbuhl, Jeff Donahue, Trevor Darrell, and Alexei A Efros. Context encoders: Feature learning by inpainting. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2536–2544, 2016.

[45] Alec Radford, Luke Metz, and Soumith Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks. *arXiv preprint arXiv:1511.06434*, 2015.

[46] Zhongzheng Ren and Yong Jae Lee. Cross-domain self-supervised multi-task feature learning using synthetic imagery. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 762–771, 2018.

[47] Yonglong Tian, Dilip Krishnan, and Phillip Isola. Contrastive multiview coding. *arXiv preprint arXiv:1906.05849*, 2019.

[48] Carl Vondrick, Abhinav Shrivastava, Alireza Fathi, Sergio Guadarrama, and Kevin Murphy. Tracking emerges by colorizing videos. In *Proceedings of the European Conference on Computer Vision*, pages 391–408, 2018.

[49] Sheng-Yu Wang, Oliver Wang, Andrew Owens, Richard Zhang, and Alexei A Efros. Detecting photoshopped faces by scripting photoshop. *arXiv preprint arXiv:1906.05856*, 2019.

[50] Xiaolong Wang and Abhinav Gupta. Unsupervised learning of visual representations using videos. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2794–2802, 2015.

[51] Zhirong Wu, Yuanjun Xiong, Stella X Yu, and Dahua Lin. Unsupervised feature learning via non-parametric instance discrimination. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3733–3742, 2018.

[52] Ye et al. Unsupervised embedding learning via invariant and spreading instance feature. In *CVPR*, 2019.

[53] Zhan et al. Self-supervised learning via conditional motion propagation. In *CVPR*, 2019.

[54] Liheng Zhang, Guo-Jun Qi, Liqiang Wang, and Jiebo Luo. Aet vs. aed: Unsupervised representation learning by auto-encoding transformations rather than data. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2547–2555, 2019.

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

[56] Richard Zhang, Phillip Isola, and Alexei A Efros. Split-brain autoencoders: Unsupervised learning by cross-channel prediction. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1058–1067, 2017.

[57] Hang Zhao, Chuang Gan, Andrew Rouditchenko, Carl Vondrick, Josh McDermott, and Antonio Torralba. The sound of pixels. In *Proceedings of the European Conference on Computer Vision*, pages 570–586, 2018.

[58] 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.

[59] Peng Zhou, Xintong Han, Vlad I Morariu, and Larry S Davis. Learning rich features for image manipulation detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1053–1061, 2018.

[60] Chengxu Zhuang, Alex Lin Zhai, and Daniel Yamins. Local aggregation for unsupervised learning of visual embeddings. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 6002–6012, 2019.