Unsupervised Object-Level Representation Learning from Scene Images

Anonymous Author(s)
Affiliation
Address
email

Abstract

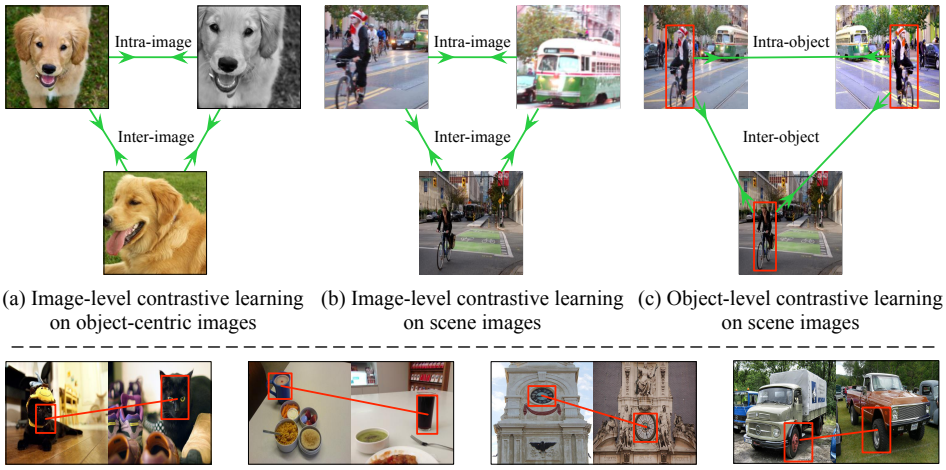
Contrastive self-supervised learning has largely narrowed the gap to the supervised pre-training on ImageNet. However, its success highly relies on the object-centric priors of ImageNet, *i.e.*, different augmented views of the same image correspond to the same object. Such a heavily curated constraint becomes immediately infeasible when pre-trained on more complex scene images with many objects. To overcome this limitation, we introduce a new self-supervised learning framework towards object-level representation learning from scene images. Our key insight is to leverage image-level self-supervised pre-training as the prior to discover object-level semantic correspondence, thus realizing object-level representation learning from scene images. Extensive experiments on COCO show that our framework significantly improves the performance of self-supervised learning from scene images. Furthermore, our framework improves the performance on downstream tasks when more unlabeled scene images are available, demonstrating its great potential of harnessing unlabeled data in the wild.

1 Introduction

Unsupervised visual representation learning aims at obtaining transferable features with abundant unlabeled data. Recent self-supervised learning (SSL) methods based on contrastive learning [57, 21, 36, 5, 18, 4, 7] have largely narrowed the gap and even surpassed the supervised counterpart on a number of downstream tasks [28, 45, 14, 44, 34, 22]. These methods build upon the instance discrimination task that maximizes the agreement between different data-augmented views of the same image. Despite their success, current SSL methods are primarily pre-trained on the unlabeled ImageNet [8] dataset that contains iconic images with single object as shown in Figure 1(a). The underlying object-centric constraint of ImageNet makes it hard to be applied in real world scenarios where more complex scene images with multiple objects are available. Meanwhile, naïvely adopting the off-the-shelf contrastive learning methods on scene images introduces inconsistent learning signals since random crops of the same image may correspond to different objects as shown in Figure 1(b). Indeed, it has been shown that current contrastive learning methods tend to struggle on more complex scene datasets [18, 47, 33, 55] like COCO [32] or Places365 [65]. Therefore, it is imperative to design an effective object-level representation learning paradigm as illustrated in Figure 1(c) to harness massive unlabeled scene images in the wild.

In this work, we are interested in going beyond ImageNet to obtain better representations on non-iconic images. Apparently, it is challenging to learn representations from scene-level images since they are entangled with many concepts including structures, objects, backgrounds and relationships. It remains an open question how to take advantages of spatial information of multiple objects naturally residing in the scene images when no object annotations are available, let alone further deriving object-level correspondence to construct positive object-instance pairs.

Submitted to 35th Conference on Neural Information Processing Systems (NeurIPS 2021). Do not distribute.



(a) Image-level contrastive learning on object-centric images (b) Image-level contrastive learning on scene images (c) Object-level contrastive learning on scene images

(d) Example region correspondence discovered across images

Figure 1: (a) Current image-level contrastive learning methods heavily rely on the object-centric bias of ImageNet, *i.e.*, different crops correspond to the same object. Prior works use either the different views of the same image [57, 21, 36, 5, 18, 7] (*i.e.*, intra-image) or similar images [67, 30, 58, 1, 10] (*i.e.*, inter-image) to form positive pairs. (b) Directly adopting image-level contrastive learning methods on scene images can cause inconsistent learning signals since different crops may correspond to different objects. (c) Object-level contrastive learning can overcome the limitation in (b) by enforcing object-level consistency. (d) We find that image-level contrastive learning encodes priors for region correspondence discovery across images, and high-response regions are usually objects or object parts (we show one discovered object-instance pair per image pair for clarity), which is useful for object-level representation learning.

To tackle these challenges, we introduce a novel object-level unsupervised representation learning framework tailored for scene images. Our framework is based on a key insight of the current contrastive learning methods: they can implicitly group different images with similar visual concepts together even though they are explicitly optimized to group different views of the same image. This phenomenon reveals that image-level contrastive learning has already induced a latent space with rich visual concepts. Though the latent space usually entangles other scene concepts like structures, backgrounds and relationships, it will be useful for object discovery if appropriately deployed. Through computing the similarity of sampled regions between kNN-retrieved images, we conclude two observations: 1) image-level contrastive learning encodes priors for region correspondence discovery across images; 2) high-response regions are usually objects or object parts.

Based on the observation above, we propose a multi-stage framework for unsupervised object-level representation learning. Specifically, we first extract potential object-based regions in scene images using the unsupervised region proposal algorithms (e.g., selective search [53]). We then propose a region correspondence generation scheme to leverage the off-the-shelf image-level contrastive learning pre-trained model to discover corresponding object-instance pairs for the proposed regions in the embedding space. Finally, we use the obtained object-instance pairs to construct positive sample pairs for object-level representation learning. Figure 1(d) shows several cross-image object-instance pairs discovered by our framework on COCO dataset using the latent prior of BYOL [18], the state-of-the-art image-level contrastive learning method. The discovered inter-corresponding pairs substantially provide diverse intra-class variances at the object-instance level to aid object-level representation learning.

Overall, our main contributions are summarized as follows:

1) We observe that existing image-level contrastive learning methods have priors to discover object-level correspondence across images. We leverage this prior for the first time for unsupervised cross-image object-level correspondence discovery.

2) With the obtained correspondence, we introduce a novel multi-stage self-supervised learning pipeline, termed as ORL, for object-level representation learning from scene images, going beyond object-centric ImageNet.

3) We contribute the first study for SSL beyond image-level contrastive learning. ORL shows substantial performance improvements over image-level contrastive learning approaches pre-trained on COCO dataset ($\sim$118k images with labels discarded), setting a new state of the art on this challenging dataset that contains diverse scenes in the wild. When SSL is conducted on a larger "COCO+" dataset (COCO train 2017 set plus the COCO unlabeled 2017 set, $\sim$241k images in total), ORL further improves the performance, demonstrating its potential to benefit from more unlabeled scene data.

2 Related work

Self-supervised learning. Self-supervised learning builds unsupervised representations by exploiting the internal priors or structures of data in the form of a pretext task. A wide range of pretext tasks have been proposed in the past few years. Examples include patch context prediction [9], jigsaw puzzles [38], inpainting [42], colorization [29, 63], cross-channel prediction [64], visual primitive counting [39], and rotation prediction [13]. Although good representations emerge with these pretext tasks, they are prone to lose generality due to their hand-crafted nature.

Recently, contrastive learning [19] that performs instance discrimination [57, 21, 36, 5, 18, 4, 7] has shown great potential in this filed, largely narrowing the gap to fully supervised learning. The core idea of contrastive learning is to gather positive pairs and separate negative pairs in the embedding space. A positive pair is usually formed with two transformed views of the same image while the negative pairs are formed with different images. Typically, contrastive learning methods require a large number of negative samples to avoid mode collapse. These samples can be maintained within a mini-batch [41, 25, 59, 24, 2, 5], a memory bank [57, 50, 67, 36] or a queue [21, 6]. BYOL [18] and SwAV [4] further remove the necessity of involving negative pairs. BYOL directly predicts the features of one view from another view, while SwAV predicts the cluster assignments between multiple views of the same image. Despite their improved performance, the existing image-level contrastive learning methods are largely confined to the underlying object-centric bias of ImageNet.

There are also a few prior attempts [17, 3, 15] for self-supervised learning on non-curated scene images. As opposed to our work, most of them consider larger models and datasets to explore the limit of current self-supervised learning methods without further considering the object-level information residing in scene images. More recently, DenseCL [55] and Self-EMD [33] perform pixel-level representation learning tailored for dense prediction tasks. CAST [47] applies saliency maps to constrain random crops and uses Grad-CAM [46]-based attention loss to improve the localization ability of self-supervised learning. As opposed to their works, we leverage image-level contrastive learning prior for high-level semantic correspondence discovery across images and use the obtained correspondence to form positive object-instance pairs for object-level representation learning.

Visual correspondence. Visual correspondence aims at finding pairwise pixels or regions across images that result from the same scene [60], which can be regarded as similarity learning of visual descriptors among matched points or patches. While early efforts learn dense correspondence with labeled data [20, 61, 27, 52, 40], some recent works learn the similarity between the parts or landmarks of the data in an unsupervised manner [49, 48]. Our work substantially differs from this line of research from original intention. Previous works aim at accurately detect all correspondence given two images, whereas our work focuses on retrieving high-quality correspondence to improve representation learning.

3 Methodology

We propose a new multi-stage self-supervised learning framework, *i.e.*, ORL, for object-level representation learning from scene images. ORL extends the existing image-level contrastive learning framework to object level by leveraging priors from image-level instance discrimination. The overall pipeline of ORL is illustrated in Figure 2. It contains three stages: image-level pre-training, correspondence discovery, and object-level pre-training. We detail each stage as follows.

3.1 ORL pipeline

Preliminary: Contrastive learning. Our pipeline contains several contrastive learning modules in Stage 1 and 3. Without loss of generality, we consider BYOL [18] as our basic contrastive learning

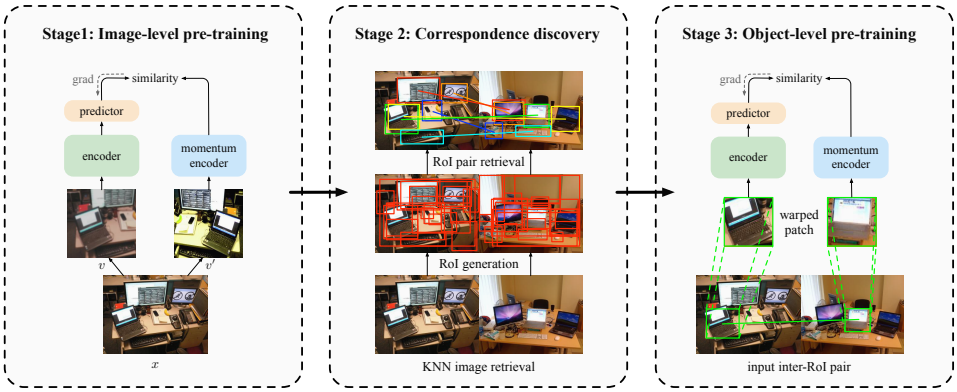


Figure 2: Overview of our three-stage pipeline. In Stage 1, we pre-train an image-level contrastive learning model. In Stage 2, we first use the pre-trained model to retrieve KNNs for each image in the embedding space to obtain image-level visually similar pairs. We then use unsupervised region proposal algorithms (*e.g.*, selective search) to generate rough RoIs for each image pair. Afterwards, we reuse the pre-trained model to retrieve the top-ranked RoI pairs, *i.e.*, correspondence. We find these pairs of RoIs are almost objects or object parts. In Stage 3, with the corresponding RoI pairs discovered across images, we finally perform object-level contrastive learning using the same architecture as BYOL.

module. BYOL uses two neural networks: the *online* network $f_\theta(x)$ and the *target* network $g_\xi(x)$. The target network provides the regression target to train the online network while its weights ξ are updated by an exponential moving average of the online parameters θ with a decay rate $\tau \in [0, 1]$ following BYOL. Given two input images x_1 and x_2, the loss function is defined as:

$$\mathcal{L}(x_1, x_2) \triangleq \|f_\theta(x_1) - g_\xi(x_2)\|_2^2, \tag{1}$$

We name it an "intra-" version of BYOL if x_1 and x_2 are two augmented views from the same image, otherwise an "inter-" one.

Stage 1: Image-level pre-training. The foremost stage is to obtain an unsupervised pre-trained model from image-level tasks. As shown in Figure 2 Stage 1, given two augmented views v and v' from the same input image x, we pre-train the network following the loss function $\mathcal{L}_{\text{image}} = \mathcal{L}(v, v')$, constituting a standard image-level BYOL pre-training. This stage can be freely replaced with other image-level contrastive learning methods, we adopt BYOL here for its simplicity and effectiveness.

Stage 2: Correspondence discovery. We employ the pre-trained image-level contrastive learning model in Stage 1 to mine object-level correspondence for the whole dataset. As shown in Figure 2 Stage 2, the overall discovery process is comprised of three steps.

(i) Image-level nearest-neighbor retrieval. Specifically, for each query image x in the training set $\mathcal{D}$, we first retrieve its top K nearest neighbors $\mathcal{N}_k$, $k = 1, ..., K$, by cosine distance in the embedding space using the features learned from the first stage to form image-level pairs that contain similar visual context.

(ii) Region-of-interest (RoI) generation. To generate object-based RoIs, we apply unsupervised region proposal algorithms, *e.g.*, selective search [53] for each image in the pair. Considering the redundancy of generated proposals (each image can have thousands of proposals), we filter certain number of them with some pre-defined thresholds[1] including the minimal scale, the range of aspect ratio, and the maximal intersection-over-union (IoU) among the filtered boxes. After the filtering operation, we select the top 100 proposals ranked with objectiveness as the candidate RoI set for subsequent RoI pair retrieval. To extract features with the equally-sized input that is compatible with the backbone, we crop and resize each RoI to 224×224. Note that even top RoIs ranked with objectiveness are still very noisy, containing a large proportion of non-object regions.

(iii) Top-ranked RoI pair retrieval. For each query RoI from x, we compute its cosine similarity in the embedding space with all RoIs from its nearest-neighbor image $\mathcal{N}_k$ using the features learned from

[1] In practice, we filter bounding boxes with the minimal scale as 96 pixels, the aspect ratio between $1/3$ and $3/1$, and the maximal IoU as 0.5.

Stage 1 again. Within the calculated cosine similarity matrix $\mathbf{M}_k \in \mathbb{R}^{100\times100}$, we retrieve top-ranked N RoI pairs to construct the set of object-level corresponding pairs $\{\mathcal{B}_k^n\}$, where $n = \{1, ..., N\}$. These high-response corresponding regions are almost objects or object parts. Finally, We save the nearest-neighbor image id and bounding box coordinate information of each corresponding pair.

Stage 3: Object-level pre-training. With the corresponding inter-image RoI (inter-RoI) pairs obtained in Stage 2, we perform object-level representation learning following the BYOL framework. Specifically, given an input image x, we first randomly select one nearest-neighbor image $\mathcal{N}_k$ to obtain the corresponding set of inter-RoI pairs $\{\mathcal{B}_k^n\}$. We then randomly select one inter-RoI pair $\mathcal{B}_k^n$ as a positive pair. With the bounding box coordinate stored in $\mathcal{B}_k^n$, we crop the corresponding inter-RoIs from x and $\mathcal{N}_k$ respectively, and resize each patch to 96×96, constituting two patches p_1 and p_2. We feed the two patches to the online network and target network separately to compute the loss $\mathcal{L}_{\text{inter-RoI}} = \mathcal{L}(p_1, p_2)$.

To make full use of discovered objects, we introduce the intra-RoI contrastive learning via augmenting object patches. Specifically, we randomly select one filtered bounding box from x obtained in Stage 2, and spatially jitter the box around its original location with the following operations[2]: (i) a random box center shifting within 50% of its width and height, (ii) a random area scaling between 50% and 200% of the original box, and (iii) a random aspect ratio between $1/2$ and $2/1$. Similarly, we crop the two intra-RoIs p and p', and resize each patch to 96×96 for forward propagation to compute the loss $\mathcal{L}_{\text{intra-RoI}} = \mathcal{L}(p, p')$. The diverse spatial jittering of the bounding box encourages the network to preserve common object information and disregard the background, thus further improving the localization ability.

We also keep the two original global views in BYOL since they preserve the global image-level information compared with the local patches. The final loss for our ORL can thus be formulated as:

$$\mathcal{L}_{\text{ORL}} = \lambda_1 \mathcal{L}_{\text{image}} + \lambda_2 \mathcal{L}_{\text{intra-RoI}} + \lambda_3 \mathcal{L}_{\text{inter-RoI}}, \tag{2}$$

where λ_1, λ_2, λ_3 are the loss weights to balance each term. We set all loss weights to 1 by default. We also compute the symmetric loss $\widetilde{\mathcal{L}}_{\text{ORL}}$ following BYOL.

3.2 Implementation details

Dataset. We pre-train our models on the COCO train2017 set that contains ∼118k images without using labels. Compared with the heavily curated object-centric ImageNet dataset, COCO contains more natural and diverse scenes in the wild, which is closer to real-world scenarios. We also perform self-supervised learning on a larger "COCO+" dataset (COCO train2017 set plus COCO unlabeled2017 set) to verify whether our method can benefit from more unlabeled scene data.

Image augmentations. The global image augmentation setting is the same as BYOL [18]: a 224×224-pixel random resized crop with a random horizontal flip, followed by a random color distortion, random grayscale conversion, random Gaussian blur and solarization. For the local patch augmentation, we directly crop the corresponding intra-RoI and inter-RoI on the input images, and resize each cropped patch to 96×96 to take place of the random resized cropping. The subsequent augmentations exactly follow the global ones.

Network architecture. We adopt ResNet-50 [23] as the default backbone. We use the same MLP projector g_θ and predictor q_θ as in BYOL: a linear layer with output size 4096 followed by batch normalization (BN) [26], rectified linear units (ReLU) [37], and a final linear layer with output dimension 256. We share the backbone and projector weights among the global and two local branches while the weights of predictor are not shared.

Optimization. For pre-training in Stage 1 and Stage 3, we use the same training hyper-parameters. Specifically, we use the SGD optimizer with a weight decay of 0.0001 and a momentum of 0.9. We adopt the cosine learning rate decay schedule [35] with a base learning rate of 0.2, linearly scaled[16] with the batch size ($lr = 0.2 \times \text{BatchSize}/256$). The batch size is set to 512 by default, which is friendly to typical 8-GPU implementations. To keep the training iterations comparable with the ImageNet supervised pre-training, we train our models for 800 epochs with a warm-up period of 4 epochs. The exponential moving average parameter τ starts from 0.99 and is increased to 1

[2]Inspired by the anchor technique used in object detection, we adopt the similar operations for our bounding box augmentations but in a continuous fashion.

during training, following [18]. For correspondence generation in Stage 2, we retrieve top $K = 10$ nearest neighbors for each image and select top-ranked $N = 10\%$ RoI pairs for each image-level nearest-neighbor pair.

4 Experiments

4.1 Transferring to downstream tasks

We evaluate the quality of learned representations by transferring them to multiple downstream tasks. Following common protocol [17, 36], we use two evaluation setups: (i) the pre-trained network is *frozen* as a feature extractor, and (ii) the network parameters are fully *fine-tuned* as weight initialization. We provide more experimental details in the supplementary material.

Method	Pre-train data	VOC07 mAP	ImageNet Top-1	Places205 Top-1	iNat. Top-1
Random [17]	-	9.6	13.7	16.6	4.8
Supervised [36]	ImageNet	87.5	75.9	51.5	45.4
SimCLR [5]	COCO	78.1	50.9	48.0	22.7
MoCo v2 [6]	COCO	82.2	55.1	48.8	27.8
BYOL [18]	COCO	84.5	57.8	50.5	29.5
ORL (ours)	COCO	**86.7**	**59.0**	**52.7**	**31.8**
BYOL [18]	COCO+	87.0	59.6	52.7	30.9
ORL (ours)	COCO+	**88.6**	**60.7**	**54.1**	**32.0**

Table 1: Image classification with linear models. All unsupervised methods are based on 800-epoch pre-training on COCO(+) with ResNet-50. We report mAP on the VOC07 dataset and top-1 center-crop accuracy on all other datasets. Numbers for all other methods are reproduced by us.

Method	Pre-train data	VOC07 low-shot (mAP)							
		1	2	4	8	16	32	64	96
Random	-	9.2	9.4	11.1	12.3	14.3	17.4	21.3	23.8
Supervied	ImageNet	53.0	63.6	73.7	78.8	81.8	83.8	85.2	86.0
SimCLR [5]	COCO	33.3	43.5	52.5	61.1	66.7	70.5	73.7	75.0
MoCo v2 [6]	COCO	39.5	49.3	60.4	69.3	74.1	76.8	79.1	80.1
BYOL [18]	COCO	39.4	50.9	62.2	71.7	76.6	79.2	81.3	82.2
ORL (ours)	COCO	**39.6**	**51.2**	**63.4**	**72.6**	**78.2**	**81.3**	**83.6**	**84.7**
BYOL [18]	COCO+	41.1	54.3	66.6	75.2	80.1	82.6	84.6	85.4
ORL (ours)	COCO+	**42.1**	**54.9**	**67.4**	**75.7**	**81.3**	**83.7**	**85.8**	**86.7**

Table 2: Low-shot image classification on VOC07 using linear SVMs trained on the fixed representations. All unsupervised methods are pre-trained on COCO(+) for 800 epochs with ResNet-50. We report mAP for each case across five runs.

Method	Pre-train data	1% labels		10% labels	
		Top-1	Top-5	Top-1	Top-5
Random	-	1.6	5.0	21.8	44.2
Supervised [62]	ImageNet	25.4	48.4	56.4	80.4
SimCLR [5]	COCO	23.4	46.4	52.2	77.4
MoCo v2 [6]	COCO	28.2	54.7	57.1	81.7
BYOL [18]	COCO	28.4	55.9	58.4	82.7
ORL (ours)	COCO	**31.0**	**58.9**	**60.5**	**84.2**
BYOL [18]	COCO+	28.3	56.0	59.4	83.6
ORL (ours)	COCO+	**31.8**	**60.1**	**60.9**	**84.4**

Table 3: Semi-supervised learning on ImageNet. All unsupervised methods are pre-tained on COCO(+) for 800 epochs with ResNet-50. We fine-tune all models with 1% and 10% ImageNet labels, and report both top-1 and top-5 center-crop accuracy on the ImageNet validation set.

Image classification with linear models Following [17, 36], we assess the quality of features by training linear classifiers on top of the *fixed* representations extracted from different depths of the network for four datasets: VOC07 [11], ImageNet [8], Places205 [66], and iNaturalist18 [54]. These datasets involve diverse classification tasks ranging from object classification, scene recognition to fine-grained recognition. For VOC07, we train linear SVMs using LIBLINEAR package [12] following the setup in [17, 36]. We train on `trainval` split of VOC07 and evaluate mAP on `test` split. For ImageNet, Places205 and iNaturalist18, we follow [64, 17, 36] and train a 1000-way,

Method	Pre-train data	COCO detection			COCO instance seg.		
		AP^{bb}	AP^{bb}_{50}	AP^{bb}_{75}	AP^{mk}	AP^{mk}_{50}	AP^{mk}_{75}
Random [55]	-	32.8	50.9	35.3	29.9	47.9	32.0
Supervised [55]	ImageNet	39.7	59.5	43.3	35.9	56.6	38.6
SimCLR [5]	COCO	37.0	56.8	40.3	33.7	53.8	36.1
MoCo v2 [6]	COCO	38.5	58.1	42.1	34.8	55.3	37.3
Self-EMD [33]	COCO	39.3	60.1	42.8	-	-	-
DenseCL [55]	COCO	39.6	59.3	43.3	35.7	56.5	38.4
BYOL [18]	COCO	39.5	59.3	43.2	35.6	56.5	38.2
ORL (ours)	COCO	**40.3**	**60.2**	**44.4**	**36.3**	**57.3**	**38.9**
BYOL [18]	COCO+	40.0	60.1	44.0	36.2	57.1	39.0
ORL (ours)	COCO+	**40.6**	**60.8**	**44.5**	**36.7**	**57.9**	**39.3**

Table 4: Object detection and instance segmentation fine-tuned on COCO. All unsupervised methods are based on 800-epoch pre-training on COCO(+). We use Mask R-CNN R50-FPN ($1\times$ schedule), and report bounding-box AP (AP^{bb}) and mask AP (AP^{mk}). Numbers for MoCo v2 are adopted from [55].

205-way and 8142-way linear classifier, respectively. We train on `train` split of each dataset, and report top-1 center-crop accuracy on the respective `val` split. Table 1 reports the results for the best-performing layer of each method. Our ORL substantially outperforms the BYOL baseline on all four datasets. We also observe that the COCO pre-trained ORL surpasses the supervised ImageNet pre-trained counterpart on Places205 by 1.2% in top-1 accuracy. This is the first time that a self-supervised learner outperforms the ImageNet pre-training using only $\sim$1/10 images compared with ImageNet. When pre-trained on a larger COCO+ dataset, ORL again outperforms BYOL. Note that apart from Places205 (2.6% gains), ORL also surpasses the supervised ImageNet counterpart on VOC07 by 1.1% mAP, using merely $\sim$1/5 images.

Low-shot image classification. We perform low-shot image classification with few training examples per class on VOC07 dataset following the same setup in [17]. We vary the number of labeled examples per category used to train linear SVMs on `train` split of VOC07 and report the average mAP across five independent samples for each low-shot case evaluated on `test` split. Table 2 shows the results (the standard deviation is provided in the supplementary material). ORL shows consistent performance improvement over BYOL for each low-shot value, with larger gains achieved as the number of labeled examples per class is increasing. ORL also gradually bridges the gap to the supervised ImageNet pre-training under this scenario. We observe the consistent performance boost when pre-training on the COCO+ dataset. Note that the COCO+ pre-trained ORL again outperforms the supervised ImageNet pre-training when the low-shot values are 64 and 96.

Semi-supervised learning. We perform semi-supervised learning on ImageNet following the protocol of [57, 24, 36, 5, 18]. Specifically, we first randomly select 1% and 10% labeled data from ImageNet `train` split. We then fine-tune our models on these two training subsets and report both top-1 and top-5 accuracy on the official `val` split of ImageNet in Table 3. Again, ORL outperforms BYOL as well as the supervised ImageNet counterpart by large margins.

Object detection and segmentation. We train a Mask R-CNN model [22] with R50-FPN backbone [31] implemented in Detectron2 [56]. We *fine-tune* all layers end-to-end on COCO `train2017` split with the standard $1\times$ schedule and evaluate on COCO `val2017` split. We follow the same setup in [51], with batch normalization layers synchronized across GPUs [43]. As shown in Table 4, ORL yields 0.8% AP and 0.7% AP improvements over BYOL for object detection and instance segmentation, respectively. The improvements are consistent over all evaluation metrics. When pre-trained on the COCO+ dataset, ORL again outperforms BYOL. It should be well noted that ORL even outperforms the most recent Self-EMD and DenseCL that are specifically designed for dense prediction downstream tasks. More importantly, either COCO or COCO+ pre-trained ORL can surpass the supervised ImageNet pre-training on all metrics. This further demonstrates the superiority of learning unsupervised representations at object level.

4.2 Ablation study

In this subsection, we conduct extensive ablation experiments to examine the effect of each component that contributes to ORL. We pre-train our models on COCO and observe the downstream performance of all ablations on VOC07 SVM classification benchmark as introduced in Section 4.1.

Table 5: Ablations for ORL. (a) Effect of intra-RoI and inter-RoI losses. (b) Effect of NNs and RoI pairs. (c) Comparison with multi-crop BYOL. (d) Comparison with ground truth bounding boxes. (e) Pre-trainig schedules. We report mAP of linear SVMs on VOC07 classification benchmark.

(a)

pre-train	intra-RoI	inter-RoI	VOC07
BYOL			84.5
ORL	✓		85.7
		✓	85.9
	✓	✓	86.7

(b)

	# of NNs			# of RoI pairs		
	1	10	20	5%	10%	20%
VOC07	84.7	86.7	87.0	86.1	86.7	86.4

(c)

pre-train	input views	VOC07
BYOL	$2 \times 224 + 4 \times 96$	84.0
ORL	$2 \times 224 + 4 \times 96$	86.7

(d)

boxes	VOC07
selective search	85.4
ground truth	86.7

(e)

pre-train	100	200	400	800
BYOL	77.1	81.8	83.7	84.5
ORL	83.5	85.2	86.3	86.7

Effect of intra-RoI and inter-RoI losses. Table 5a ablates the effect of our introduced $\mathcal{L}_{\text{intra-RoI}}$ and $\mathcal{L}_{\text{inter-RoI}}$ losses in Equation 2. Adding either $\mathcal{L}_{\text{intra-RoI}}$ or $\mathcal{L}_{\text{inter-RoI}}$ can improve the performance, with the best results obtained by adding both terms.

Effect of nearest neighbors and RoI pairs. Table 5b ablates the effect of the number of nearest neighbors K and RoI pairs N used for generating inter-RoI pairs in Stage 2 of ORL. We set $N = 10\%$ when ablating K, and set $K = 10$ when ablating N. We observe that retrieving more nearest neighbors leads to better performance since more nearest neighbors provide more diverse image-level pairs to the subsequent generation of inter-RoI pairs. Although setting $K = 20$ produces a slightly better performance, we choose $K = 10$ by default as a trade-off considering the tendency of the saturated performance. Our method is more robust to the number of retrieved top-ranked RoI pairs after image-level nearest-neighbor retrieval, with $N = 10\%$ performing slightly better.

Comparison with multi-crop BYOL. Prior work [4] has indicated that cropping multiple views of the same image can improve the performance of self-supervised learning methods pre-trained on ImageNet. To investigate whether our gains are due to more accurate object-instance comparison or simply more number of mixed views, we randomly crop four additional smaller views for BYOL to ensure the number and size of the input patches are equal (*i.e.*, $2 \times 224 + 4 \times 96$). As shown in Table 5c, different from the observation on ImageNet, simply adding more low-resolution crops tend to hurt the performance since it will further intensify the inconsistent noise on scene images. In contrast, ORL substantially outperforms this multi-crop variant, validating that the gains are truly due to our object-level representation learning mechanism.

Comparison with ground truth bounding boxes. In Stage 2, ORL requires an unsupervised region proposal algorithm to extract approximate object-based regions, which is inaccurate to some extent. We further investigate whether the performance can be improved when more accurate object regions are available, *i.e.*, with bounding box annotations. To this end, we replace our selective-search generated object proposals with ground truth bounding boxes provided from COCO `train2017` set, while keeping other procedures unchanged. As shown in Table 5d, adopting ground truth bounding boxes perform inferior to selective search. This is mainly due to that although the ground truth bounding boxes can provide more accurate object location, their numbers are too scarce compared with a large amount of region proposals generated by selective search. The more diverse region proposals can potentially induce more unknown object or object-part discovery beyond the manually annotated objects. In this case, the diversity can make up for the inaccuracy.

Pre-training schedules. Table 5e shows the results with different pre-training schedules. The performance of both ORL and BYOL improves when pre-trained for longer epochs, while ORL consistently outperforms BYOL by at least 2.2% mAP. Note that our 200-epoch ORL has already surpassed the 800-epoch BYOL, demonstrating the efficiency of ORL.

4.3 Visualization

Correspondence pairs. Figure 3 visualizes some top-ranked region correspondence discovered in Stage 2 of ORL based on the latent prior of BYOL. We observe that each generated inter-RoI pair

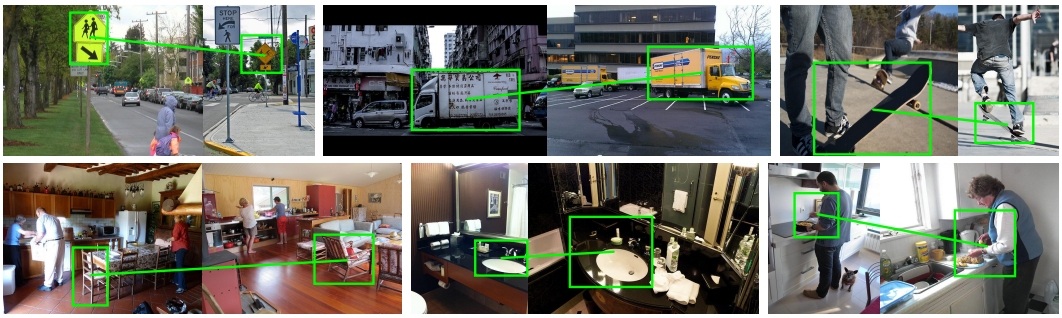


Figure 3: Top-ranked region correspondence discovered by ORL in Stage 2. We show a pair of discovered object-instance per image pair for clarity. More discovered correspondence pairs are provided in the supplementary material.

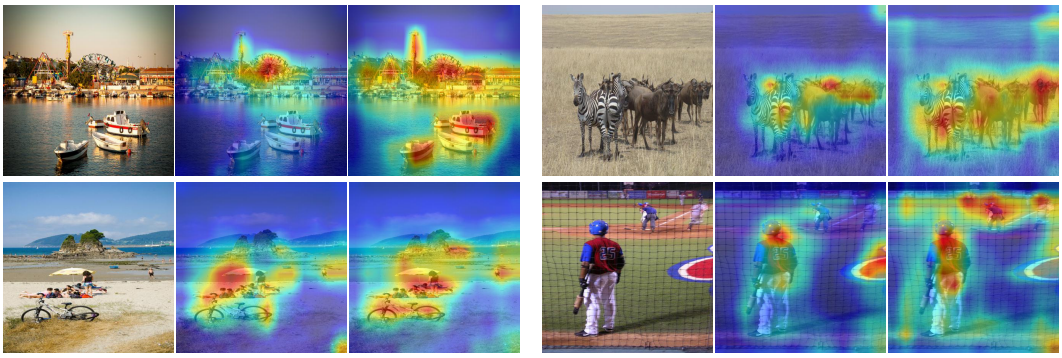


Image BYOL ORL Image BYOL ORL

Figure 4: Attention maps generated by BYOL and ORL. ORL can activate more object regions and produce more accurate object boundary in the heatmap than BYOL. We provide more attention maps in the supplementary material. Zoomed in for better visualized results.

largely correspond to the regions with similar object-level visual concepts (*i.e.*, objects or object parts) across images. These correspondence pairs can provide more natural and diverse object-level intra-class variances compared with the typical contrastive learning methods that perform intra-image augmentations to simulate such variances.

Attention maps. Figure 4 visualizes the attention maps generated by BYOL and ORL. We observe that both BYOL and ORL can produce relatively high-quality attention maps that focus on the foreground objects. This reflects from the side that current image-level contrastive learning methods have already induced a latent space with rich visual concepts. Nevertheless, ORL can activate more object regions and produce more accurate object boundary than BYOL in the generated attention maps. It is mainly due to introducing object-level similarity learning into ORL, which can minimize the inconsistent noise caused by image-level contrastive learning. In contrast, BYOL only uses the whole image to extract features, thus activating the most discriminative region.

5 Conclusion

In this work, we have presented a new self-supervised learning framework, ORL, for object-level representation learning from scene images. We leverage the latent prior of image-level self-supervised learning pre-training for discovering object-based region correspondence across images. The generated object-instance correspondence enables us to perform pairwise contrastive learning at the object level. ORL significantly improves the performance of self-supervised learning from scene images in a variety of downstream tasks. We expect that our method can be applied to larger-scale unlabeled data in the wild to fully realize its potential, and hope that our study can attract the community's attention to more general-purpose unsupervised representation learning from scene images.

References

[1] Mehdi Azabou, Mohammad Gheshlaghi Azar, Ran Liu, Chi-Heng Lin, Erik C Johnson, Kiran Bhaskaran-Nair, Max Dabagia, Keith B Hengen, William Gray-Roncal, Michal Valko, et al. Mine your own view: Self-supervised learning through across-sample prediction. *arXiv preprint arXiv:2102.10106*, 2021. 2

[2] Philip Bachman, R Devon Hjelm, and William Buchwalter. Learning representations by maximizing mutual information across views. In *NeurIPS*, pages 15509–15519, 2019. 3

[3] Mathilde Caron, Piotr Bojanowski, Julien Mairal, and Armand Joulin. Unsupervised pre-training of image features on non-curated data. In *ICCV*, pages 2959–2968, 2019. 3

[4] Mathilde Caron, Ishan Misra, Julien Mairal, Priya Goyal, Piotr Bojanowski, and Armand Joulin. Unsupervised learning of visual features by contrasting cluster assignments. *arXiv preprint arXiv:2006.09882*, 2020. 1, 3, 8

[5] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. *arXiv preprint arXiv:2002.05709*, 2020. 1, 2, 3, 6, 7

[6] Xinlei Chen, Haoqi Fan, Ross Girshick, and Kaiming He. Improved baselines with momentum contrastive learning. *arXiv preprint arXiv:2003.04297*, 2020. 3, 6, 7

[7] Xinlei Chen and Kaiming He. Exploring simple siamese representation learning. *arXiv preprint arXiv:2011.10566*, 2020. 1, 2, 3

[8] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *CVPR*, pages 248–255, 2009. 1, 6

[9] Carl Doersch, Abhinav Gupta, and Alexei A Efros. Unsupervised visual representation learning by context prediction. In *ICCV*, 2015. 3

[10] Debidatta Dwibedi, Yusuf Aytar, Jonathan Tompson, Pierre Sermanet, and Andrew Zisserman. With a little help from my friends: Nearest-neighbor contrastive learning of visual representations. *arXiv preprint arXiv:2104.14548*, 2021. 2

[11] Mark Everingham, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The pascal visual object classes (voc) challenge. *IJCV*, 88(2):303–338, 2010. 6

[12] Rong-En Fan, Kai-Wei Chang, Cho-Jui Hsieh, Xiang-Rui Wang, and Chih-Jen Lin. Liblinear: A library for large linear classification. *JMLR*, 9:1871–1874, 2008. 6

[13] Spyros Gidaris, Praveer Singh, and Nikos Komodakis. Unsupervised representation learning by predicting image rotations. In *ICLR*, 2018. 3

[14] Ross Girshick, Jeff Donahue, Trevor Darrell, and Jitendra Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *CVPR*, pages 580–587, 2014. 1

[15] Priya Goyal, Mathilde Caron, Benjamin Lefaudeux, Min Xu, Pengchao Wang, Vivek Pai, Mannat Singh, Vitaliy Liptchinsky, Ishan Misra, Armand Joulin, et al. Self-supervised pretraining of visual features in the wild. *arXiv preprint arXiv:2103.01988*, 2021. 3

[16] Priya Goyal, Piotr Dollár, Ross Girshick, Pieter Noordhuis, Lukasz Wesolowski, Aapo Kyrola, Andrew Tulloch, Yangqing Jia, and Kaiming He. Accurate, large minibatch sgd: Training imagenet in 1 hour. *arXiv preprint arXiv:1706.02677*, 2017. 5

[17] Priya Goyal, Dhruv Mahajan, Abhinav Gupta, and Ishan Misra. Scaling and benchmarking self-supervised visual representation learning. In *ICCV*, pages 6391–6400, 2019. 3, 6, 7

[18] Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre H Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Avila Pires, Zhaohan Daniel Guo, Mohammad Gheshlaghi Azar, et al. Bootstrap your own latent: A new approach to self-supervised learning. *arXiv preprint arXiv:2006.07733*, 2020. 1, 2, 3, 5, 6, 7

[19] Raia Hadsell, Sumit Chopra, and Yann LeCun. Dimensionality reduction by learning an invariant mapping. In *CVPR*, 2006. 3

[20] Xufeng Han, Thomas Leung, Yangqing Jia, Rahul Sukthankar, and Alexander C Berg. Matchnet: Unifying feature and metric learning for patch-based matching. In *CVPR*, 2015. 3

[21] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *CVPR*, pages 9729–9738, 2020. 1, 2, 3

[22] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *ICCV*, pages 2961–2969, 2017. 1, 7

[23] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016. 5

[24] Olivier J Hénaff, Aravind Srinivas, Jeffrey De Fauw, 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. 3, 7

[25] 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. In *ICLR*, 2019. 3

[26] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *ICML*, 2015. 5

[27] Angjoo Kanazawa, David W Jacobs, and Manmohan Chandraker. Warpnet: Weakly supervised matching for single-view reconstruction. In *CVPR*, 2016. 3

[28] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *NeurIPS*, 2012. 1

[29] Gustav Larsson, Michael Maire, and Gregory Shakhnarovich. Learning representations for automatic colorization. In *ECCV*, pages 577–593, 2016. 3

[30] Junnan Li, Pan Zhou, Caiming Xiong, Richard Socher, and Steven CH Hoi. Prototypical contrastive learning of unsupervised representations. *arXiv preprint arXiv:2005.04966*, 2020. 2

[31] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection. In *CVPR*, pages 2117–2125, 2017. 7

[32] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *ECCV*, 2014. 1

[33] Songtao Liu, Zeming Li, and Jian Sun. Self-emd: Self-supervised object detection without imagenet. *arXiv preprint arXiv:2011.13677*, 2020. 1, 3, 7

[34] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *CVPR*, 2015. 1

[35] Ilya Loshchilov and Frank Hutter. Sgdr: Stochastic gradient descent with warm restarts. *arXiv preprint arXiv:1608.03983*, 2016. 5

[36] Ishan Misra and Laurens van der Maaten. Self-supervised learning of pretext-invariant representations. In *CVPR*, pages 6707–6717, 2020. 1, 2, 3, 6, 7

[37] Vinod Nair and Geoffrey E Hinton. Rectified linear units improve restricted boltzmann machines. In *ICML*, 2010. 5

[38] Mehdi Noroozi and Paolo Favaro. Unsupervised learning of visual representations by solving jigsaw puzzles. In *ECCV*, 2016. 3

[39] Mehdi Noroozi, Hamed Pirsiavash, and Paolo Favaro. Representation learning by learning to count. In *ICCV*, pages 5898–5906, 2017. 3

[40] Yuki Ono, Eduard Trulls, Pascal Fua, and Kwang Moo Yi. Lf-net: Learning local features from images. *arXiv preprint arXiv:1805.09662*, 2018. 3

[41] Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018. 3

[42] Deepak Pathak, Philipp Krahenbuhl, Jeff Donahue, Trevor Darrell, and Alexei A Efros. Context encoders: Feature learning by inpainting. In *CVPR*, 2016. 3

[43] Chao Peng, Tete Xiao, Zeming Li, Yuning Jiang, Xiangyu Zhang, Kai Jia, Gang Yu, and Jian Sun. Megdet: A large mini-batch object detector. In *CVPR*, pages 6181–6189, 2018. 7

[44] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *NeurIPS*, pages 91–99, 2015. 1

[45] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *IJCV*, 115(3):211–252, 2015. 1

[46] Ramprasaath R Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *ICCV*, pages 618–626, 2017. 3

[47] Ramprasaath R Selvaraju, Karan Desai, Justin Johnson, and Nikhil Naik. Casting your model: Learning to localize improves self-supervised representations. *arXiv preprint arXiv:2012.04630*, 2020. 1, 3

[48] James Thewlis, Samuel Albanie, Hakan Bilen, and Andrea Vedaldi. Unsupervised learning of landmarks by descriptor vector exchange. In *ICCV*, 2019. 3

[49] James Thewlis, Andrea Vedaldi, and Hakan Bilen. Unsupervised object learning from dense equivariant image labelling. In *NeurIPS*, 2017. 3

[50] Yonglong Tian, Dilip Krishnan, and Phillip Isola. Contrastive multiview coding. *arXiv preprint arXiv:1906.05849*, 2019. 3

[51] Yonglong Tian, Chen Sun, Ben Poole, Dilip Krishnan, Cordelia Schmid, and Phillip Isola. What makes for good views for contrastive learning. *NeurIPS*, 2020. 7

[52] Nikolai Ufer and Bjorn Ommer. Deep semantic feature matching. In *CVPR*, 2017. 3

[53] Jasper RR Uijlings, Koen EA Van De Sande, Theo Gevers, and Arnold WM Smeulders. Selective search for object recognition. *IJCV*, 104(2):154–171, 2013. 2, 4

[54] Grant Van Horn, Oisin Mac Aodha, Yang Song, Yin Cui, Chen Sun, Alex Shepard, Hartwig Adam, Pietro Perona, and Serge Belongie. The inaturalist species classification and detection dataset. In *CVPR*, pages 8769–8778, 2018. 6

[55] Xinlong Wang, Rufeng Zhang, Chunhua Shen, Tao Kong, and Lei Li. Dense contrastive learning for self-supervised visual pre-training. *arXiv preprint arXiv:2011.09157*, 2020. 1, 3, 7

[56] Yuxin Wu, Alexander Kirillov, Francisco Massa, Wan-Yen Lo, and Ross Girshick. Detectron2, 2019. 7

[57] Zhirong Wu, Yuanjun Xiong, Stella X Yu, and Dahua Lin. Unsupervised feature learning via non-parametric instance discrimination. In *CVPR*, pages 3733–3742, 2018. 1, 2, 3, 7

[58] Jiahao Xie, Xiaohang Zhan, Ziwei Liu, Yew Soon Ong, and Chen Change Loy. Delving into inter-image invariance for unsupervised visual representations. *arXiv preprint arXiv:2008.11702*, 2020. 2

[59] Mang Ye, Xu Zhang, Pong C Yuen, and Shih-Fu Chang. Unsupervised embedding learning via invariant and spreading instance feature. In *CVPR*, pages 6210–6219, 2019. 3

[60] Ramin Zabih and John Woodfill. Non-parametric local transforms for computing visual correspondence. In *ECCV*, 1994. 3

[61] Sergey Zagoruyko and Nikos Komodakis. Learning to compare image patches via convolutional neural networks. In *CVPR*, 2015. 3

[62] Xiaohua Zhai, Avital Oliver, Alexander Kolesnikov, and Lucas Beyer. S4l: Self-supervised semi-supervised learning. In *ICCV*, pages 1476–1485, 2019. 6

[63] Richard Zhang, Phillip Isola, and Alexei A Efros. Colorful image colorization. In *ECCV*, 2016. 3

[64] Richard Zhang, Phillip Isola, and Alexei A Efros. Split-brain autoencoders: Unsupervised learning by cross-channel prediction. In *CVPR*, 2017. 3, 6

[65] Bolei Zhou, Agata Lapedriza, Aditya Khosla, Aude Oliva, and Antonio Torralba. Places: A 10 million image database for scene recognition. *TPAMI*, 40(6):1452–1464, 2017. 1

[66] Bolei Zhou, Agata Lapedriza, Jianxiong Xiao, Antonio Torralba, and Aude Oliva. Learning deep features for scene recognition using places database. In *NeurIPS*, pages 487–495, 2014. 6

[67] Chengxu Zhuang, Alex Lin Zhai, and Daniel Yamins. Local aggregation for unsupervised learning of visual embeddings. In *ICCV*, pages 6002–6012, 2019. 2, 3

Checklist

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes]
   (b) Did you describe the limitations of your work? [Yes] See the supplementary material.
   (c) Did you discuss any potential negative societal impacts of your work? [Yes] See the supplementary material.
   (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes]
2. If you are including theoretical results...
   (a) Did you state the full set of assumptions of all theoretical results? [N/A]
   (b) Did you include complete proofs of all theoretical results? [N/A]
3. If you ran experiments...
   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [Yes]
   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [Yes]
   (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [Yes]
   (d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [Yes]
4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...
   (a) If your work uses existing assets, did you cite the creators? [Yes]
   (b) Did you mention the license of the assets? [No]
   (c) Did you include any new assets either in the supplemental material or as a URL? [No]
   (d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [N/A]
   (e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [N/A]
5. If you used crowdsourcing or conducted research with human subjects...
   (a) Did you include the full text of instructions given to participants and screenshots, if applicable? [N/A]

(b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A]

(c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A]