Edinburgh Research Explorer

Contrastive Object-level Pre-training with Spatial Noise Curriculum Learning

Contrastive Object-level Pre-training with Spatial Noise Curriculum Learning

Chenhongyi Yang[1*]   Lichao Huang[2]   Elliot J. Crowley[1]

[1]School of Engineering, University of Edinburgh

[2]Horizon Robotics

Abstract

The goal of contrastive learning based pre-training is to leverage large quantities of unlabeled data to produce a model that can be readily adapted downstream. Current approaches revolve around solving an image discrimination task: given an anchor image, an augmented counterpart of that image, and some other images, the model must produce representations such that the distance between the anchor and its counterpart is small, and the distances between the anchor and the other images are large. There are two significant problems with this approach: (i) by contrasting representations at the image-level, it is hard to generate detailed object-sensitive features that are beneficial to downstream object-level tasks such as instance segmentation; (ii) the augmentation strategy of producing an augmented counterpart is fixed, making learning less effective at the later stages of pre-training. In this work, we introduce Curricular Contrastive Object-level Pre-training (CCOP) to tackle these problems: (i) we use selective search to find rough object regions and use them to build an inter-image object-level contrastive loss and an intra-image object-level discrimination loss into our pre-training objective; (ii) we present a curriculum learning mechanism that adaptively augments the generated regions, which allows the model to consistently acquire a useful learning signal, even in the later stages of pre-training. Our experiments show that our approach improves on the MoCo v2 baseline by a large margin on multiple object-level tasks when pre-training on multi-object scene image datasets. Code is available at https://github.com/ChenhongyiYang/CCOP.

1. Introduction

Imagine that you are given an *object-level task* such as object detection [16, 31, 25], or instance segmentation [19], but you do not have enough data or compute resources to warrant training a network from scratch. A popular so-

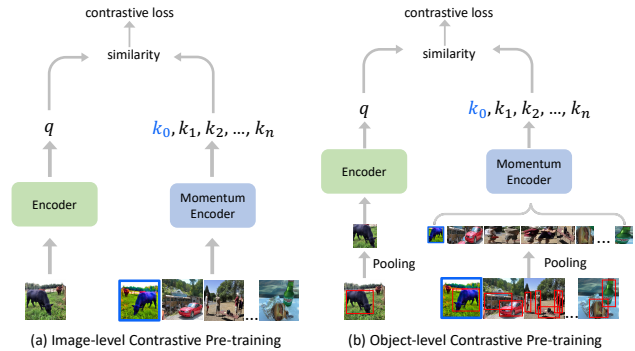*Corresponding Author. Email: chenhongyi.yang@ed.ac.uk

Figure 1: Image-level contrastive pre-training versus object-level contrastive pre-training. The former computes a single representation for each image and uses it in an instance discrimination task. This is not ideal for multi-object scene images. The latter addresses this problem by computing an embedding for each object in the input image and applies object-level instance discrimination.

lution is to take a pre-trained network trained on a existing large-scale dataset, and then use it for the downstream object-level task [16]. This allows the network to benefit from an existing store of data, helping it converges faster and generalise better than training from scratch. But which pre-trained network should you use?

For several years, the most popular pre-trained networks were those that had been trained with full supervision on ImageNet [33]; however, recent works [6, 18, 37, 35, 41, 21, 39, 42] have demonstrated that self-supervised *contrastive learning* can be used to produce comparable, or better networks without requiring any manual annotation. In contrastive learning, a deep model is trained to perform an instance discrimination task, in which two augmented views of an image are produced: the anchor, and its counterpart. Given the anchor, the model must find its counterpart from a huge number of noisy (negative) samples. This is achieved by mapping each image into a low-dimensional embedding space, and using a contrastive loss to maximise the similarity between the anchor image and its positive counterpart while minimising the similarity between the anchor and

the noisy samples. Intuitively, this enables the model to discriminate images based on high-level semantic features, rather than low-visual level cues.

However, there are two notable problems with this approach. Firstly, as illustrated in Figure 1, the contrastive loss is applied to distinguish whole images using a global representation, making it hard for the model to produce detailed object-level features; as a consequence, the resulting model may perform poorly when adapted downstream for object-level tasks such as detection and instance segmentation [31, 19]. The second problem lies in how the augmented view of the anchor is built [35]; current approaches use a fixed augmentation strategy throughout the whole pre-training process. However, in the later stages of pre-training, the model is strong enough to easily maximise the similarity between the anchor and its counterpart, saturating the loss gradient provided by the positive pair.

In this work, we present Curricular Contrastive Object-level Pre-training (CCOP) to address these problems. To learn regional spatially-sensitive features we introduce two additional loss terms utilising object-level features; we use selective search [13] to obtain rough object-level bounding boxes, and then extract features through the RoIAlign [19] operation. These features are fed into a MLP to produce object-level embeddings. We use a contrastive loss to allow the model to distinguish between object embeddings across different images, and an intra-image discrimination loss to allow the model to differentiate between objects from the same image. These two losses together with a standard image-level contrastive loss enable the model to learn not only global features but also local spatially-sensitive features, and excel when applied downstream to object-level tasks. To solve the loss saturation problem, we introduce a curriculum learning mechanism [2] to make the instance discrimination task harder as pre-training progresses. Specifically, in order to construct positive pairs of a suitable difficulty, we propose Spatial Noise Curriculum Learning (SNCL), which uses a Spatial Curriculum Selection (SCS) module that adaptively creates several augmented boxes and selects the most suitable proposal for the loss based on a dynamic difficulty-controlling mechanism. This allows the model to consistently receive a useful learning signal through the whole pre-training process.

In summary, our contributions are threefold:

- We present a simple and effective object-level self-supervised pre-training framework called Curricular Contrastive Object-level Pre-training (CCOP), in which rough object locations are obtained without supervision, and an object-level inter-image contrastive loss and an intra-image discrimination loss are combined to improve pre-training for downstream object-level tasks when training on multi-object scene images.

- We propose Spatial Noise Curriculum Learning (SNCL), which uses a Spatial Curriculum Selection (SCS) module to select the most informative positive pairs at an appropriate difficulty to aid learning.

- Through extensive experiments, we show our approach achieves state-of-the-art transfer learning performance on several object-level tasks when pre-training on MS-COCO [26].

2. Related Work

2.1. Self-Supervised Pre-training

Self-supervised methods [1, 10, 6, 18, 5, 17, 8] manipulate input data to extract a surrogate supervision signal in the form of a pretext task. There exist numerous pretext tasks including context prediction [30], grayscale image colourisation [44], jigsaw puzzle solving [3], rotation prediction [14], and image clustering [4].

Recently, instance discrimination as a pretext task has achieved promising results [6]. It considers each image in a dataset as its own class, and uses an InfoNCE loss [29] to match the same instance from among a large number of negative samples. However, this is extremely expensive. MoCo [18, 7] remedies this by using a memory bank that stores previously-computed representations as negative samples. They rely on noise contrastive estimation to compare instances, which is a special form of contrastive learning. [9] shows that the memory bank can be entirely replaced with elements from the same batch if the batch is large enough. More recently, BYOL [17], SimSiam [8], and SWAV [5] avoid comparison with negative samples. BYOL and SimSiam directly draw together features from the same instance, while SWAV maps image features to a set of trainable prototype vectors.

However, models produced through contrasting global image-level representations are unable to capture the inherent differences between image-level representations and object-level, and pixel-level representations. Several works have been proposed to remedy this. In PixPro [42], pixel-level contrastive learning is performed by exploiting the geometry between pixels in different views. DetCon [21] uses unsupervised segmentation algorithms to find pixels belonging to the same object and then contrastively learns pixel representations. DenseCL [37] adopts a grid matching mechanism between different images to compute similarity. Self-EMD [27] improves on this by computing similarity with the earth mover's distance. For object-level pre-training, Ins-Loc [43] combines images to produce pseudo-objects whose regional features are used for contrastive learning. In DetCo [41], a global image representation is composed of several local grid representations so that local features can explicitly contribute to the loss. In SCRL [32],
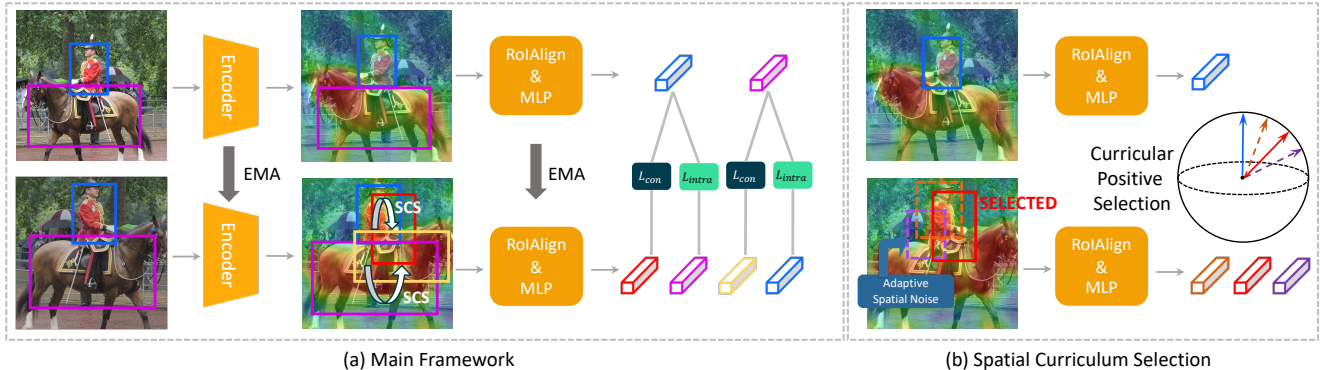
Figure 2: (a) Our main framework. We first compute a set of rough bounding boxes for objects in the input image using selective search. Two different views are fed to the query network (upper branch) and key network (lower branch) respectively. Each box in the key branch is then jittered by our Spatial Curriculum Selection module to yield augmented boxes of appropriate difficulty, and the hardest of these is selected to be matched with its counterpart in the query branch. (b) In more detail, the Spatial Curriculum Selection module takes a box in the key branch, and produces several jittered boxes where the extent of jittering is proportional to how far we are into pre-training. A feature is computed for each of these boxes, and the feature that is the least similar to the corresponding feature in the query branch is used for learning.

random regions are selected from images for use in contrastive learning. The closest work to our approach is SoCo [39]; this also utilises selective search to find rough objects and forces embeddings of regional representations to be close. However, it uses a fixed strategy for box augmentation, and does not provide a mechanism for the model to distinguish between different objects within an image.

Several works show that constructing good positive and negative samples is essential for contrastive pre-training. In [45], hard positive extrapolation is used to increase the distance between positive pairs and negative interpolation is used to improve the diversity of negative samples. A hard-negative mixing mechanism was proposed in [22] to create a hard negative set for each query in MoCo. InfoMin [35] uses an adversarial training strategy to reduce the mutual information between positive pairs.

2.2. Curriculum Learning

Curriculum learning was first proposed in [2] to mimic the easy-to-hard learning practices that are commonly used to help with human and animal learning. Recently [34] used a transferable curriculum learning framework to allow information transfer between source and target domains; DeepSDF [11] was proposed to help reconstruct shapes through gradually increasing difficulty; DCL [38] uses an adaptive sampling strategy with loss re-weighting to aid learning when given class-imbalanced data.

Following the core easy-to-hard philosophy of curriculum learning, our proposed SNCL gradually decrease the similarity between positive object pairs as our model gets stronger.

3. Method

In this section we describe our object-level self-supervising training framework: Curricular Contrastive Object-level Pre-training (CCOP). It is adapted from MoCo v2, which we review in Section 3.1. The key innovations of CCOP are (i) the incorporation of object-level objectives, and (ii) spatial noise curriculum learning, which we describe in Sections 3.2 and Section 3.3 respectively.

3.1. Description of the Baseline Model

Contrastive learning allows models to be trained so that different views of the same image are mapped to be close in some representation space, while different images are mapped to be far away from each other.

MoCo v2 [7] is a successful contrastive pre-training framework and is adopted as our baseline approach. It comprises a query network $f_q^{\theta_q}$ and a key network $f_k^{\theta_k}$ that share the same architecture, where θ_q and θ_k are the model parameters. Each network consists of two parts: a backbone network e.g. ResNet-50 [20], and a MLP head that maps the backbone features to an embedding vector. For training, an image x is augmented to produce two views x_q and x_k that are fed into f_q and f_k respectively to get their embeddings z_q and z_k, and a memory queue Q is built to store the embeddings generated in previous iterations, which are used as negative samples. A contrastive loss is employed so that the model can match z_q and z_k against a huge number of noisy negative samples z_k^-:

$$\mathcal{L}_{con} = -\log \frac{\exp\left(z_q \cdot z_k/\tau\right)}{\exp\left(z_q \cdot z_k/\tau\right) + \sum_{z_k^- \in Q} \exp\left(z_q \cdot z_k^-/\tau\right)} \quad (1)$$

where τ is a temperature hyper-parameter. The query net-

work is updated using the gradient of this loss, and the key network is the exponential moving average (EMA) of the query network:

$$\theta_k \leftarrow m\theta_k' + (1-m)\theta_q. \qquad (2)$$

When pre-training is complete, the backbone of the query network is extracted for downstream use.

3.2. Object-level Objectives

Our goal is to pre-train a model that will excel when applied to downstream object-level tasks such as instance segmentation and object detection. This pre-training should be conducted without any supervision, so that we are able to leverage large amounts of unlabelled data wherever available in the wild. We would like such a model to produce object-sensitive features. Because of this we choose to use multi-object scene images for pre-training. However, identifying the object regions to produce these features is challenging; this typically requires supervision.

To circumvent this, we take inspiration from [15] and use selective search [13] to obtain rough bounding boxes for the objects without supervision. We use RoIAlign [19] to extract the regional features in a box from the feature map generated by the backbone network. These are fed into a 2-layer MLP to yield object embeddings. Please see Figure 2 for an illustration. We use a separate memory queue Q_{obj} to store object embeddings. Bounding boxes that correspond to the same object in the two augmented image views are treated as positive pairs, and the regional representations stored in Q_{obj} are used as negative samples in a constrative loss $\mathcal{L}_{con}^{obj}$, which assumes the same form as Equation 1.

However, contrasting box embeddings against the negative samples stored in Q_{obj} may allow the model to find a shortcut where it could leverage the global image-level information to avoid learning detailed object-level features. To prevent this from happening, we introduce an intra-image discrimination loss to force the model to compute diverse embeddings for objects in the same image. Given bounding boxes in an image $\{B_i\}_{i=1}^N$ with corresponding embeddings $\{z_i\}_{i=1}^N$, we use a hinge loss

$$\mathcal{L}_{intra} = \frac{1}{\|\mathbf{P}\|_1} \sum_{i=1}^N \sum_{j \neq i} \mathbf{P}_{ij} \max\{(z_i \cdot z_j - \alpha), 0\}. \quad (3)$$

Here $\mathbf{P}$ is an indicator matrix where $\mathbf{P}_{ij}$ is 1 only if the IoU between B_i and B_j is less than 0.05. Note that we compute $\mathcal{L}_{intra}$ using the original boxes instead of the augmented boxes. This loss reduces the similarity between box embeddings that do not overlap in an image, and can be seen as a form of regularisation which prevents the model from mapping distinct objects close together.

The final loss function used to train the model is combination of the original image contrastive loss in MoCo v2,

the object contrastive loss, and the intra-image discrimination loss:

$$\mathcal{L}_{total} = \mathcal{L}_{con}^{img} + \mathcal{L}_{con}^{obj} + \mathcal{L}_{intra} \qquad (4)$$

3.3. Spatial Noise Curriculum Learning

In contrastive learning, a fixed data augmentation strategy is often deployed for the whole training process. As a result, the learning signal that the model receives from the contrastive loss will decrease at the later stages of pre-training; as the model becomes stronger it can more easily match positive pairs with a high similarity from the noisy negative samples, which reduces the magnitude of the gradients provided by the positive pair. Let's consider the gradient of $\mathcal{L}_{con}$ with respect to z_q. For $\tau = 1$ it is given by

$$\nabla_{z_q}\mathcal{L}_{con} = \frac{\sum_{i=1}^M (z_i^- - z_k) \cdot \exp(z_q \cdot z_i^-)}{\exp(z_q \cdot z_k) + \sum_{i=1}^M \exp(z_q \cdot z_i^-)} \qquad (5)$$

where $\{z_i^-\}_{i=1}^M$ are noisy samples stored in the memory bank, and M is the size of the memory bank (65,335 in our case). In the later stages of training, the $z_q \cdot z_i^-$ terms tend to zero [45], and this gradient can be approximated as

$$\nabla_{z_q}\mathcal{L}_{con} \approx (z_q - z_k) + \frac{1}{M}\sum_i^M (z_i^- - z_q). \qquad (6)$$

We can see that when z_k is very close to z_q, the first term in Equation 6 disappears and only negative samples are contributing to the gradient this loss.

To address this problem, we employ curriculum learning to gradually raise the difficulty of the instance-discrimination task so that the model can consistently receive a useful learning signal. Here, we define the difficulty as the similarity between z_q and z_k. For image-level contrastive pre-training, one could generate several augmentations and adaptively select the most suitable augmented image for the loss. However, this would be expensive to compute. Fortunately, at the object-level we can augment a bounding box multiple times and quickly acquire their features through the effective RoIAlign operation [19]. This faciliates Spatial Noise Curriculum Learning (SNCL).

SNCL aims to gradually decrease the similarity of positive object pairs by jittering the bounding boxes as training progresses; we hypothesise that this will allow the model to attend to fine-grained visual cues in order to connect these bounding boxes. In our framework, we achieve this by introducing a Spatial Curriculum Selection (SCS) module, shown in Figure 2, that adaptively creates a set of jittered boxes of a dynamically-controlled difficulty, before selecting the hardest of these for learning. In detail, for a box $B^k = (x, y, w, h)$[1] in the key branch, the SCS module gen-

---

[1] (x, y) are the coordinates for the top-left corner of a box with width and height w and h respectively.

Figure 3: Our Spatial Curriculum Selection module augments object bounding boxes so that they are an appropriate difficulty for the model throughout training (note that colour distortions are not shown here).

erates a set of augmented boxes $\{\hat{B}_i^k\}$ through:

$$x' = \sigma_x w + x, \; y' = \sigma_y h + y,$$
$$w' = w \cdot \exp(\sigma_w), \; h' = h \cdot \exp(\sigma_h), \qquad (7)$$

where $\sigma_x, \sigma_y, \sigma_w, \sigma_h$ are noise coefficient drawn from a uniform distribution $(-\zeta, \zeta)$. In order to increase the difficulty, we linearly scale ζ from 0.3 to 1.0 as training proceeds. We ensure that the IoU between the jittered boxes and the original box is larger than a threshold β_t, which is dynamically lowered from 0.8 to 0.3 in our SNCLwith respect to the training iteration t. Then, the SCS module select the hardest box $\hat{B}_j^k$ to compute z_k for contrastive learning where $j = argmin_i\{z_q \cdot \hat{z}_i\}$. Here $\hat{z}_i$ is the object embedding corresponding to $\hat{B}_i^k$. This enlarges the first term in Equation 6, allowing z_k to contribute to the loss gradient. We visualise the boxes produced by this process for different epochs in Figure 3.

4. Experiments

4.1. Experimental Setup

Datasets. CCOP utilises multi-object scene images as pre-training data to produce a model that excels at object-level downstream tasks. For our experiments we use MS-COCO [26] as the pre-training dataset; specifically, the *train2017* subset for pre-training, which contains 118k images. Note that we do not use any annotations.

Model settings. We use MoCo v2 [7] as our baseline. We use ResNet-50 [20] as the default backbone network. We follow [39] to add a feature pyramid network (FPN) [24] to the backbone network, whose generated feature maps are from $\frac{1}{4}$ to $\frac{1}{32}$ of the input image size. Following [18] we use the C_5 feature map for image embeddings, and the features

Method	AP^{bb}	AP^{bb}_{50}	AP^{bb}_{75}	AP^{mk}	AP^{mk}_{50}	AP^{mk}_{75}
Rand Init	31.0	49.6	33.3	28.6	46.7	30.5
IN Sup	38.7	59.4	42.2	35.2	56.2	37.6
SimCLR [6]	38.0	57.3	41.5	34.3	54.5	36.6
BYOL [17]	38.0	57.4	41.5	34.5	54.6	36.9
SimSiam [8]	38.3	58.0	41.8	34.8	55.2	37.2
MoCo v2 [7]	38.5	58.0	41.9	34.9	55.2	37.3
Self-EMD [27]	39.3	**60.1**	42.8	-	-	-
DenseCL [37]	39.6	59.3	43.3	35.7	**56.8**	38.4
CCOP (Ours)	**39.9**	59.8	**43.9**	**36.2**	56.8	**38.8**

Table 1: COCO Object detection and instance segmentation using Mask R-CNN FPN with $1\times$ schedule. 'Rand Init' refers to training from scratch and 'IN Sup' refers to ImageNet supervised pre-training. All unsupervised models are pre-trained using COCO *train2017* for 800 epochs.

Method	AP^{bb}	AP^{bb}_{50}	AP^{bb}_{75}	AP^{mk}	AP^{mk}_{50}	AP^{mk}_{75}
Rand Init	17.7	31.0	17.7	16.5	28.8	16.6
IN Sup	25.5	42.4	27.0	23.3	39.4	24.3
SimCLR	24.1	40.0	25.3	22.0	37.2	22.8
BYOL	24.2	41.6	24.9	22.3	38.5	22.9
SimSiam	24.5	40.5	26.0	22.4	37.8	23.4
MoCo v2	24.2	40.1	25.5	22.2	37.6	22.9
DenseCL	26.1	42.7	27.2	23.8	40.1	24.9
CCOP (Ours)	**26.4**	**43.3**	**27.9**	**24.0**	**40.4**	**25.1** †

Table 2: Semi-supervised object detection and instance segmentation on COCO using Mask R-CNN FPN. All models are fine-tuned for 27k iterations using only 10% of the COCO *train2017* annotated data.

for object-level embeddings are pooled by a RoIAlign operation from the FPN feature maps according to their size. We use two separate MLPs with intermediate dimensions of 2048 to get 128-D embeddings. Both Q_{img} and Q_{obj} are of size 65,335.

Selective search and data augmentation. Selective search [13] is performed offline before pre-training to produce bounding box proposals for each image. Boxes whose IoU are larger than 0.5 are merged. We remove bounding boxes whose areas are smaller than 12^2 and whose aspect ratios are outside of the range $(0.33, 3)$. During training, images are augmented as in MoCo v2: a random patch of the image is cropped and resized to 224×224, followed by colour distortion, random Gaussian blur, random grayscale and random horizontal flip operations. If after augmentation, a bounding box is clipped by more than 40% of its original area, it is removed.

Optimisation. We closely follow the pre-training settings in [37]. We train the model for 800 epochs with an initial learning rate of 0.3, which is cosine-annealed [28] to zero. We use an SGD optimiser with weight decay and momentum of 0.0001 and 0.9 respectively. The margin parameter α in Equation 3 is set to 0.4. We use 8 Nvidia 2080Ti GPUs for pre-training with a mini-batch size of 256.

Method	COCO RetinaNet			PascalVOC Faster R-CNN			COCO Keypoint R-CNN		
	AP^{bb}	AP_{50}^{bb}	AP_{70}^{bb}	AP^{bb}	AP_{50}^{bb}	AP_{75}^{bb}	AP^{kp}	AP_{50}^{kp}	AP_{kp}^{bb}
Rand Init	25.5	40.9	26.7	39.8	65.7	41.4	63.0	85.2	69.0
IN Sup	37.9	57.9	40.4	54.4	80.4	59.7	65.5	87.1	**71.8**
SimCLR	36.4	55.8	38.4	53.1	78.9	58.4	65.2	86.8	70.7
BYOL	36.7	56.1	39.3	52.4	79.6	57.1	65.1	86.6	70.8
SimSiam	36.9	56.6	39.1	53.2	79.3	58.3	65.6	86.8	71.5
MoCo v2	36.6	56.3	38.8	53.6	79.9	58.9	65.3	86.8	71.3
DenseCL	38.3	58.1	41.0	55.3	80.5	60.8	65.5	87.2	71.5
CCOP (Ours)	**38.4**	**58.4**	**41.2**	**55.8**	**80.7**	**61.9**	**65.7**	**87.4**	71.6

Table 3: Transfer learning performance of COCO object detection using RetinaNet, Pascal VOC object detection using Faster R-CNN FPN, and COCO Keypoint detection using Keypoint R-CNN FPN. All models are pre-trained on COCO *train2017* for 800 epochs.

Implementation. Our code is based on OpenSelf-Sup[2] and is available at `https://github.com/ChenhongyiYang/CCOP`. We performed pre-training for competing methods locally[3] across 8 Nvidia 2080Ti GPUs, using the training schemes given in their respective papers, and the config files in the OpenSelfSup toolkit where available. Downstream models in Section 4.2 were trained locally using Detectron2 [40].

4.2. Main Results

Here, we apply our pre-trained model to a suite of downstream object-level tasks and compare against existing pre-training approaches.

COCO object detection and segmentation. In Table 1, we compare the performance of our approach to existing pre-training schemes by fine-tuning a Mask R-CNN FPN model. We followed the training settings in [37]: the model is trained on the COCO *train2017* subset and tested on the *mini-val* subset. Synchronised batch normalisation (SyncBN) is used in the backbone, FPN, and detection heads. CCOP achieves a 1.4 and 1.3 absolute improvement on AP^{bb} and AP^{mk} compared to the MoCo v2 baseline, and surpasses other recent pre-training approaches including those designed for object-level and pixel-level tasks such as DenseCL [37] and Self-EMD [27].

Semi-supervised detection and segmentation. In Table 2, we report the semi-supervised object detection and instance-segmentation performance on COCO where only 10% of images are used for fine-tuning[4]. This experiment allows us to gauge how well CCOP works in a regime where we have limited downstream annotation. Our approach achieves a 2.4 AP^{bb} and 1.8 AP^{mk} improvement over the baseline MoCo v2 model and is state-of-the-art.

COCO single-stage object detection. Here, we show that the model pre-trained using CCOP can be used in a single stage object detector, demonstrating its flexibility. We transfer the pre-trained model to a RetinaNet [25] detector using $1\times$ training schedule. SyncBN is used in the backbone and FPN, and group normalisation is used in the detection heads. Results are given in Table 3. Our model attains 1.8 absolute AP^{bb} improvement compared to the baseline of MoCo v2, which is larger than the improvement for the two-stage Mask R-CNN. This could be because our method is able to pre-train not only the backbone network, but also the FPN neck that plays an more important role in single stage detectors [23].

Pascal VOC object detection. To verify whether our model can generalise to datasets other than COCO, we transfer the pre-trained model to a Faster R-CNN FPN detector and fine-tune it using the Pascal VOC [12] dataset. Following [18], we train the model for 24k iterations using the *trainval2007+2012* sets and evaluate it on the *test2007* set. The results are provided in Table 3; our methods achieves a 2.2 improvement in AP over the baseline and surpasses other state-of-the-art approaches, indicating that our model is able to generalise to new data.

COCO human keypoint detection. Our pre-trained model also works well when used downstream for human keypoint detection. In Table 3, we provide the results obtained when adapting our pre-trained model to a Keypoint R-CNN FPN [19] model. Our approach achieves 65.7 AP^{kp}, broadly improving on alternative pre-training methods.

4.3. Ablation Studies

In this section, we examine how the different aspects of CCOP contribute to the overall performance. We use

---

Obj	Intra	SNCL	AP^{bb}	AP^{bb}_{50}	AP^{bb}_{75}	AP^{mk}	AP^{mk}_{50}	AP^{mk}_{75}
			38.5	58.0	41.9	34.9	55.2	37.3
✓			39.3	59.3	42.9	35.6	56.5	38.2
✓	✓		39.6	59.5	43.2	35.8	56.2	38.5
✓		✓	39.7	**59.9**	43.4	36.0	**56.9**	38.5
✓	✓	✓	**39.9**	59.8	**43.9**	**36.2**	56.8	**38.8**

Table 4: Ablation Studies of our contributions. 'Obj' stands for object-level contrastive learning; 'Intra' stands for intra-image discrimination loss; SNCL is our Spatial Noise Curriculum Learning approach.

Method	AP^{bb}	AP^{bb}_{50}	AP^{bb}_{75}	AP^{mk}	AP^{mk}_{50}	AP^{mk}_{75}
MoCo v2	38.5	58.0	41.9	34.9	55.2	37.3
Random	39.1	58.9	42.8	35.1	55.8	37.8
GT	39.4	59.3	43.2	35.5	56.4	38.2
SS	**39.9**	**59.8**	**43.9**	**36.2**	**56.8**	**38.8**

Table 5: Comparison of different object generation approaches. 'Random' stands for random generated boxes; 'GT' stands for human annotated boxes; 'SS' stands for selective search.

β	AP^{bb}	AP^{bb}_{50}	AP^{bb}_{75}	AP^{mk}	AP^{mk}_{50}	AP^{mk}_{75}
1.0	39.6	59.5	43.2	35.8	56.2	38.5
0.6	39.5	56.5	42.9	35.7	56.5	38.0
0.3	39.2	58.9	42.7	35.5	56.1	38.2
SNCL	**39.9**	**59.8**	**43.9**	**36.2**	**56.8**	**38.8**

Table 6: Comparison of box augment strategies with different IoU lower-bound β. Note that 1.0 stands for no augmentation.

α	AP^{bb}	AP^{bb}_{50}	AP^{bb}_{75}	AP^{mk}	AP^{mk}_{50}	AP^{mk}_{75}
0.0	38.8	58.1	41.9	35.2	55.0	37.4
0.2	39.5	59.5	42.9	35.6	56.5	38.0
0.4	**39.9**	**59.8**	**43.9**	**36.2**	56.8	**38.8**
0.6	**39.9**	**59.8**	43.4	36.0	**56.8**	38.5

Table 7: Ablation study of the margin parameter α in the intra-image discrimination loss.

Backbone	Method	AP^{bb}	AP^{bb}_{50}	AP^{bb}_{75}	AP^{mk}	AP^{mk}_{50}	AP^{mk}_{75}
ResNet-18	MoCo v2	35.0	53.9	38.2	31.8	51.2	34.0
	CCOP	35.6	54.7	38.6	32.3	52.0	34.4
ResNet-50	MoCo v2	38.5	58.0	41.9	34.9	55.2	37.3
	CCOP	39.9	59.8	43.9	36.2	56.8	38.8
ResNet-101	MoCo v2	41.0	60.7	44.5	36.7	57.6	39.5
	CCOP	42.1	61.9	45.9	37.6	58.7	40.3

Table 8: Performance comparison between our CCOP and the baseline MoCo v2 on different backbone sizes.

COCO object detection and instance segmentation to evaluate different models, which are fine-tuned using a Mask R-CNN FPN model with $1\times$ training schedule.

Role of each module. In Table 4 we examine how object-level contrastive learning, the intra-image discrimination loss, and Spatial Noise Curriculum Learning contribute to the overall performance. Our baseline MoCo v2 model achieves 38.5 AP^{bb} and 34.9 AP^{mk}. When we include the object-level contrastive learning loss, the performance increases significantly to 39.3 AP^{bb} and 35.6 AP^{mk}. This shows that incorporating object-level information in the pre-training stage can help with downstream object-level tasks. When we add the intra-image discrimination loss, the performance is further enhanced to 39.6 AP^{bb} and 35.8 AP^{mk}. Separately adding SNCL without the intra-image loss also improves performance to 39.7 AP^{bb} and 36.0 AP^{mk}. Finally, when all three modules are adopted, we achieve 39.9 AP^{bb} and 36.2 AP^{mk}.

Object Generation. We compare proposing regions using selective search to two alternative strategies: randomly generating proposals, and using ground-truth bounding boxes. For random generation, we produce 12 bounding boxes per view (this is on average how many boxes selective search produces) and we remove boxes whose areas are smaller than 12^2 and whose aspect ratios are out of range $(0.33, 3)$. For ground-truth, we use the COCO annotations. The performance for these strategies is given in Table 5. Using randomly generated boxes gives us a marginal improvement over the baseline but is worse than using ground-truth boxes or those obtained through selective search. Our intuition is that random boxes allow the model to learn spatially-sensitive features, but do not give any notion of what an object is. We found it intriguing that selective search boxes outperform ground-truth. A possible explanation is that selective search introduces more diversity into training.

Curriculum Learning. We conduct experiments to study the extent to which curriculum learning benefits our approach. We compare SNCL to box generations with fixed magnitude noise and with a fixed IoU lower-bound β. The results are presented in Table 6. Note that when $\beta = 1$ there is no augmentation applied. We observe that even if no noise is added, the pre-trained model is quite good achieving a 39.6 AP^{bb} and 35.8 AP^{mk}. Lowering the IoU lower-bound to 0.6 barely changes the performance. However when we lower it to 0.3 this harms performance. We observe that Spatial Noise Curriculum Learning can avoid such problems by gradually enlarging the extent of augmentation. This stabilises training for the early stages of learning and raise the difficulty in later stages, leading to a superior performance of 39.9 AP^{bb} and 36.2 AP^{mk}.

Intra-image Margin In Table 7 we explore the influence of the margin α parameter in the intra-image discrimination loss. We can see that the performance is robust for large margins, but low for small values. A possible reason is that a small margin makes it difficult for the model to learn good embeddings as it introduce too much conflict in the image.

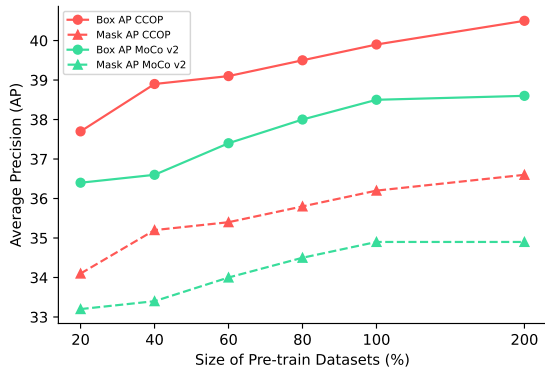**Model sizes.** In Table 8, we compare the performance

Figure 4: Performance comparison between the baseline MoCo v2 and our CCOP on Mask R-CNN FPN with 1× training schedule under different sizes of pre-training dataset. Note that 200% means that we add the extra COCO *unlabeled* set for pre-training. As we increase the dataset size, CCOP performance continues to increase.

of CCOP and the baseline MoCo v2 using ResNets of different sizes. There are performance boosts when using our approach in all cases. However, these are more significant as the models get larger.

Pre-train dataset size. In Figure 4, we show the performance of our approach and the baseline model for different sizes of the pre-training dataset. We take subsets with different percentages of the COCO training set and use them to pre-train the model for 800 epochs. We also test the performance when we add the COCO *unlabeled* set into pre-training, which contains an extra 123k images. We observe that the performance of the baseline starts to saturate earlier than our method. This suggests that our method has the potential for large-scale pre-training.

Pre-train dataset size. In Figure 4, we show the performance of our approach and the baseline model for different sizes of the pre-training dataset. We take subsets with different percentages of the COCO training set and use them to pre-train the model for 800 epochs. We also test the performance when we add the COCO *unlabeled* set into pre-training, which contains an extra 123k images. We observe that the performance of the baseline starts to saturate earlier than our method. This suggests that our method has the potential for large-scale pre-training.

4.4. Qualitative Studies

We qualitatively study how CCOP learns useful features from bounding boxes generated in an unsupervised manner on multi-object input images and how this helps with downstream object-level tasks.

In Figure 5, we visualise the learnt object embeddings
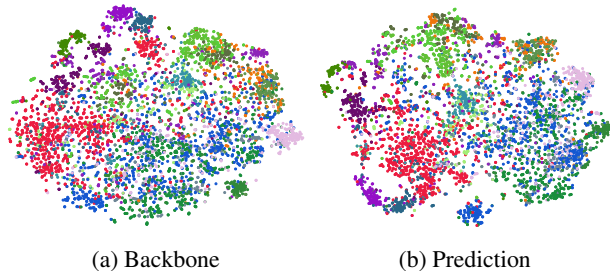

(a) Backbone   (b) Prediction

Figure 5: Visualisation of regional backbone features and predicted box embeddings on COCO *mini-val* set using T-SNE for dimension reduction. The objects are acquired by using oracle annotations. We only draw the common 20 classes, shown in different colors, that appears in both COCO and Pascal VOC.

Topk	1	5	10	20	50	100
Prediction	50.1	44.4	40.8	37.5	33.6	30.5
Backbone	48.1	43.2	39.4	35.9	31.7	28.3

Table 9: Comparison of kNN recall between the predicted object embeddings and the backbone regional features on COCO *mini-val* set using oracle boxes and class annotations.

using T-SNE [36]. In detail, we forward the COCO *mini-val* images through the pre-trained model and compute the object embeddings of the ground-truth boxes, then pick 20 common classes (denoted with different colours) that appear in both COCO and Pascal VOC for visualisation. We can see that the box embeddings in the same class form a series of clear clusters in the embedding space even though the model is trained without supervision, demonstrating the model's ability to capture high-level semantic information. As only the pre-trained backbone is used for fine-tuning, we also visualise the regional features that are directed pooled and averaged from the backbone network. The clustering structure can also be clearly observed in these features even before they are transformed using an MLP head. Further, we query the k nearest neighbours of each object in the representation space and check how many of them are in the same class, which we denote as kNN recall. These values are given in Table 9. The performance using backbone features is only slightly worse than the performance using the actual embeddings, indicating that the pre-trained model has strong potential for transfer learning.

Finally, we observe how good our approach is at learning spatially-sensitive and scale-sensitive features that are important to object-level tasks. First, two augmented views of an image are fed into the query and key branch respectively. Then we pool and average an object's regional features from the query network and use this it as a convolution
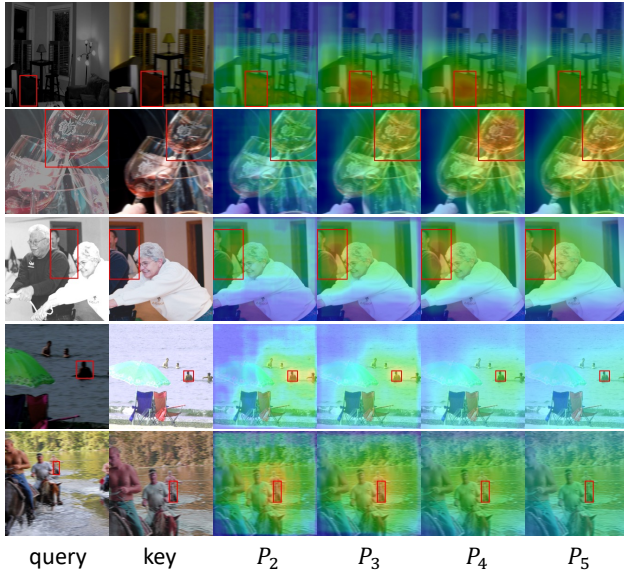
Figure 6: Visualisation of box query results. We pool the box features directly from the query network's backbone features and used it as convolution kernel to compute the cross-correlation results with key image's features in different FPN layers.

kernel to compute the cross-correlation with the backbone features computed in the key branch. The results are in Figure 6. We can see that the same object in the key image is highlighted; this suggests that the pre-trained model is able to extract features that are suitable for object-level tasks.

5. Conclusion

In this work, we propose an object-level contrastive pre-training framework: Curricular Contrastive Object-level Pre-training (CCOP) to improve transfer learning performance for downstream object-level tasks. Our approach is designed for pre-training on multi-object scene images that take both inter-image and intra-image instance discrimination into consideration. To further improve the performance, we designed Spatial Noise Curriculum Learning that adaptively selects the positive pairs of an appropriate difficulty for the current stage of training. Extensive experimental results verify that a model pre-trained using our method is able to transfer well to different tasks, different downstream model architectures, and different datasets.

References

[1] Pulkit Agrawal, João Carreira, and Jitendra Malik. Learning to See by Moving. In *ICCV*, 2015. 2

[2] Yoshua Bengio, Jérôme Louradour, Ronan Collobert, and Jason Weston. Curriculum Learning. In *ICML*, 2009. 2, 3

[3] Fabio M. Carlucci, Antonio D'Innocente, Silvia Bucci, Barbara Caputo, and Tatiana Tommasi. Domain Generalization by Solving Jigsaw Puzzles. In *CVPR*, 2019. 2

[4] Mathilde Caron, Piotr Bojanowski, Armand Joulin, and Matthijs Douze. Deep Clustering for Unsupervised Learning of Visual Features. In *ECCV*, 2018. 2

[5] Mathilde Caron, Ishan Misra, Julien Mairal, Priya Goyal, Piotr Bojanowski, and Armand Joulin. Unsupervised Learning of Visual Features by Contrasting Cluster Assignments. In *NeurIPS*, 2020. 2

[6] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A Simple Framework for Contrastive Learning of Visual Representations. In *ICML*, 2020. 1, 2, 5

[7] Xinlei Chen, Haoqi Fan, Ross Girshick, and Kaiming He. Improved baselines with momentum contrastive learning. *arXiv preprint arXiv:2003.04297*, 2020. 2, 3, 5

[8] Xinlei Chen and Kaiming He. Exploring Simple Siamese Representation Learning. *CVPR*, 2021. 2, 5

[9] Xinlei Chen, Saining Xie, and Kaiming He. An Empirical Study of Training Self-Supervised Vision Transformers. In *ICCV*, 2021. 2

[10] Carl Doersch, Abhinav Gupta, and Alexei A Efros. Unsupervised visual representation learning by context prediction. In *ICCV*, 2015. 2

[11] Yueqi Duan, Haidong Zhu, He Wang, Li Yi, Ram Nevatia, and Leonidas J Guibas. Curriculum DeepSDF. In *ECCV*, 2020. 3

[12] Mark Everingham, SM Ali Eslami, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The Pascal Visual Object Classes Challenge: A Retrospective. *IJCV*, 2015. 6

[13] Pedro F Felzenszwalb and Daniel P Huttenlocher. Efficient graph-based image segmentation. *IJCV*, 2004. 2, 4, 5

[14] Zeyu Feng, Chang Xu, and Dacheng Tao. Self-Supervised Representation Learning by Rotation Feature Decouplin. In *CVPR*, 2019. 2

[15] Ross Girshick. Fast r-cnn. In *ICCV*, 2015. 4

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

[17] 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. In *NeurIPS*, 2020. 2, 5

[18] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum Contrast for Unsupervised Visual Representation Learning. In *CVPR*, 2020. 1, 2, 5, 6

[19] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *ICCV*, 2017. 1, 2, 4, 6

[20] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016. 3, 5

[21] Olivier J. Hénaff, Skanda Koppula, Jean-Baptiste Alayrac, Aaron van den Oord, Oriol Vinyals, and João Carreira. Efficient Visual Pretraining With Contrastive Detection. In *ICCV*, 2021. 1, 2

[22] Yannis Kalantidis, Mert Bulent Sariyildiz, Noe Pion, Philippe Weinzaepfel, and Diane Larlus. Hard negative mixing for contrastive learning. *NeurIPS*, 2020. 3

[23] Yanghao Li, Yuntao Chen, Naiyan Wang, and Zhaoxiang Zhang. Scale-aware trident networks for object detection. In *ICCV*, 2019. 6

[24] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection. In *CVPR*, 2017. 5

[25] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. In *ICCV*, 2017. 1, 6

[26] 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. 2, 5

[27] Songtao Liu, Zeming Li, and Jian Sun. Self-emd: Self-supervised object detection without imagenet. *arXiv preprint arXiv:2011.13677*, 2020. 2, 5, 6

[28] Ilya Loshchilov and Frank Hutter. Sgdr: Stochastic gradient descent with warm restarts. In *ICLR*, 2017. 5

[29] Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation Learning with Contrastive Predictive Coding. *arXiv preprint arXiv:1807.03748*, 2018. 2

[30] Deepak Pathak, Philipp Krahenbuhl, Jeff Donahue, Trevor Darrell, and Alexei A Efros. Context Encoders: Feature Learning by Inpainting. In *CVPR*, 2016. 2

[31] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *NeurIPS*, 2015. 1, 2

[32] Byungseok Roh, Wuhyun Shin, Ildoo Kim, and Sungwoong Kim. Spatially Consistent Representation Learning. In *CVPR*, 2021. 2

[33] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. ImageNet Large Scale Visual Recognition Challenge. *IJCV*, 2015. 1

[34] Yang Shu, Zhangjie Cao, Mingsheng Long, and Jianmin Wang. Transferable Curriculum for Weakly-supervised Domain Adaptation. In *AAAI*, 2019. 3

[35] Yonglong Tian, Chen Sun, Ben Poole, Dilip Krishnan, Cordelia Schmid, and Phillip Isola. What Makes for Good Views for Contrastive Learning? In *NeurIPS*, 2020. 1, 2, 3

[36] Laurens Van der Maaten and Geoffrey Hinton. Visualizing Data Using T-SNE. *JMLR*, 2008. 8

[37] Xinlong Wang, Rufeng Zhang, Chunhua Shen, Tao Kong, and Lei Li. Dense contrastive learning for self-supervised visual pre-training. In *CVPR*, 2021. 1, 2, 5, 6

[38] Yiru Wang, Weihao Gan, Jie Yang, Wei Wu, and Junjie Yan. Dynamic Curriculum Learning for Imbalanced Data Classification. In *ICCV*, 2019. 3

[39] Fangyun Wei, Yue Gao, Zhirong Wu, Han Hu, and Stephen Lin. Aligning Pretraining for Detection via Object-Level Contrastive Learning. *arXiv preprint arXiv:2106.02637*, 2021. 1, 3, 5

[40] Yuxin Wu, Alexander Kirillov, Francisco Massa, Wan-Yen Lo, and Ross Girshick. Detectron2. https://github.com/facebookresearch/detectron2, 2019. 6

[41] Enze Xie, Jian Ding, Wenhai Wang, Xiaohang Zhan, Hang Xu, Peize Sun, Zhenguo Li, and Ping Luo. DetCo: Unsupervised Contrastive Learning for Object Detection. In *ICCV*, 2021. 1, 2

[42] Zhenda Xie, Yutong Lin, Zheng Zhang, Yue Cao, Stephen Lin, and Han Hu. Propagate Yourself: Exploring Pixel-Level Consistency for Unsupervised Visual Representation Learning. In *CVPR*, 2021. 1, 2

[43] Ceyuan Yang, Zhirong Wu, Bolei Zhou, and Stephen Lin. Instance localization for self-supervised detection pretraining. In *CVPR*, 2021. 2

[44] Richard Zhang, Phillip Isola, and Alexei A Efros. Colorful Image Colorization. In *ECCV*, 2016. 2

[45] Rui Zhu, Bingchen Zhao, Jingen Liu, Zhenglong Sun, and Chang Wen Chen. Improving Contrastive Learning by Visualizing Feature Transformation. In *ICCV*, 2021. 3, 4