MULTI-LEVEL CONTRASTIVE LEARNING FOR DENSE PREDICTION TASK

Anonymous authors
Paper under double-blind review

ABSTRACT

In this work, we present Multi-Level Contrastive Learning for Dense Prediction Task (MCL), an efficient self-supervised method to learn region-level feature representation for dense prediction tasks. This approach is motivated by the three key factors in detection: localization, scale consistency and recognition. Considering the above factors, we design a novel pretext task, which explicitly encodes absolute position and scale information simultaneously by assembling multi-scale images in a montage manner to mimic multi-object scenario. Unlike the existing image-level self-supervised methods, our method constructs a multi-level contrastive loss by considering each sub-region of the montage image as a singleton to learn a regional semantic representation for translation and scale consistency, while reducing the pre-training epochs to the same as supervised pre-training. Extensive experiments show that MCL consistently outperforms the recent state-of-the-art methods on various datasets with significant margins. In particular, MCL obtains 42.5 $\mathrm{AP}^{\mathrm{bb}}$ and 38.3 $\mathrm{AP}^{\mathrm{mk}}$ on COCO with the 1x schedule and surpasses MoCo by 4.0 $\mathrm{AP}^{\mathrm{bb}}$ and 3.1 $\mathrm{AP}^{\mathrm{mk}}$, when using Mask R-CNN with an R50-FPN backbone pre-trained with 100 epochs. In addition, we further explore the alignment between pretext task and downstream tasks. We extend our pretext task to supervised pre-training, which achieves a similar performance with self-supervised learning, demonstrating the importance of the alignment between pretext task and downstream tasks.

1 INTRODUCTION

A generic large-scale supervised pre-training is a critical auxiliary task for computer vision community to progress, like ImageNet(Deng et al., 2009) pre-training, which has been confirmed by many works (Erhan et al., 2010; He et al., 2019; 2017; Lin et al., 2017; Qiao et al., 2021; Ren et al., 2015; Sohn et al., 2020). Downstream tasks benefit from initializing the model with pre-trained weights, for faster convergence and better generality. Recently, many advances are driven by instance discrimination tasks based on self-supervised learning (SSL), without relying on semantic annotations. They achieve the state-of-the-art results on the challenging ImageNet dataset under the k-NN and linear probing evaluation policy. Despite their advanced performance on classification tasks, some recent works (Pinheiro et al., 2020; Wang et al., 2021; Wei et al., 2021; Xie et al., 2021b; Yang et al., 2021) observe that these methods share a common fundamental weakness: The image-level representation learning doesn't transfer well to dense prediction tasks, such as object detection and instance segmentation. Furthermore, the success of state-of-the-art methods (Caron et al., 2020; Chen et al., 2020a; Grill et al., 2020; He et al., 2020; Hénaff et al., 2021) requires several times more training epochs than the supervised pre-training counterpart.

Different from ImageNet classification task, whose scale of objects varies in a small range, most object detection datasets have a large scale variation across object instances. Besides, the bounding boxes are required to be located precisely. Therefore, an ideal detector is supposed to be scale consistent to object instances and encode position information precisely. Pixel-level SSL methods (Liu et al., 2020; Wang et al., 2021) considers the spatial structure information as shown in Fig. 1(a). The pretext tasks treat each pixel in an image as a single instance and encourage the model to distinguish each pixel from others within the image. Unfortunately, the matching rule of positive pixel pair is based on the transportation cost of feature distance, which does not guarantee a precise and stable feature target assignment. Object-level SSL methods (Hénaff et al., 2021; Wei et al., 2021)

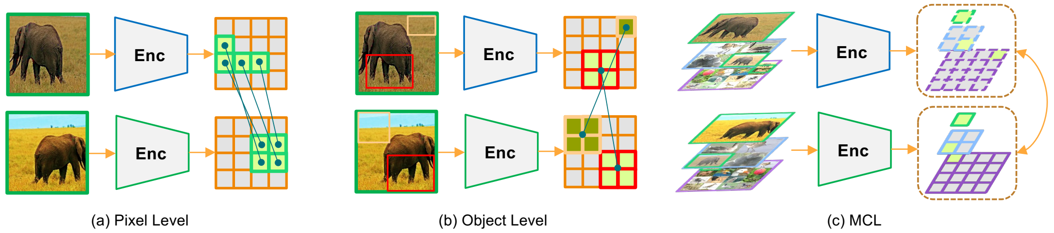

Figure 1: (a) Pixel-level methods match positive feature pair based on the transportation cost of feature distance, which does not guarantee precise assignment. (b) Object-level methods obtain localization by off-the-shelf algorithms, whose predictions are low-quality on non-object-centric dataset. (c) Our method learns regional representation for precise localization, scale consistency among multi-scale crops and semantic global representations. MCL aligns the feature map with the image region among multi-scale views.

focus on the proposals from some off-the-shelf algorithms, such as Selective Search (Uijlings et al., 2013) and Multiscale Combinatorial Grouping (Arbeláez et al., 2014), as illustrated in Fig. 1(b). However, the predicted bounding box and segmentation mask are not accurate enough when they're pre-trained on the non-object-centric dataset, such as COCO. The low-quality pseudo-labels yield an inferior result for dense prediction tasks due to the localization noise.

Motivated by the above observations, we propose a novel high-efficient self-supervised learning framework for dense prediction tasks, called Multi-Level Contrastive Learning (MCL). MCL learns regional representation for translation, scale and semantic consistency among mutli-scale regions and global representations. Besides, MCL achieves state-of-the-art transfer performance on the downstream tasks while reducing the training epochs significantly. We design a montage manner to assemble multi-scale images into non-overlapping grid for mimicking multi-object scenario. The montage assembly explicitly encodes the position and scale information of images. A single-level feature has limited capacity to represent objects with large scale variance. Therefore, we adopt a scale-aware positive target assignment strategy on different levels of the feature pyramid, which produces a multi-scale feature representation with strong semantic information. MCL treats each component image in the montage image as an independent instance and accurately extracts features from different pyramid levels for contrastive objective, according to the image coordinates, which bridges the gap between the pretext task and the downstream task. For a further investigation of the alignment between pre-trained model and finetuned model, we extend our pretext task to supervised pre-training, which achieves similar performance with self-supervised pre-training. This result breaks the empirical conclusion that SSL methods outperform their supervised counterparts in the downstream tasks (Caron et al., 2020; Chen et al., 2020a;b; Grill et al., 2020; He et al., 2021; 2020; Purushwalkam & Gupta, 2020; Tian et al., 2020; Yang et al., 2021) and demonstrates the importance of task alignment.

To evaluate the effect of MCL, we conduct extensive experiments on benchmarks for various dense prediction tasks. We demonstrate that MCL achieves state-of-the-art transfer performance from the representation learned on ImageNet and COCO dataset, while significantly reducing the training epochs, matching the supervised pre-training counterpart on ImageNet. MCL pre-trained on ImageNet with 100 epochs obtains 42.5 AP^{bb} and 38.3 AP^{mk} on COCO with the standard 1x schedule (Wu et al., 2019) and surpasses MoCo by 4.0 AP^{bb} and 3.1 AP^{mk}, using Mask R-CNN with an R50-FPN backbone. MCL pre-trained on the unlabeled COCO dataset achieves 41.8 AP^{bb} and 37.7 AP^{mk}, showing that MCL benefits from the multi-level pretext task design rather than the dataset bias (Purushwalkam & Gupta, 2020).

Our contributions are listed as follows: (1) An efficient self-supervised method, Multi-level Contrastive Learning, is designed to align the pretext task with the dense prediction task, improving scale invariance and localization precision. (2) Montage assembly is introduced in the self-supervised learning field for **the first time** to construct a montage image, mimicking multi-scale multi-object scenarios. (3) Our method achieves **state-of-the-art** transfer performance on the dense prediction downstream tasks, such as detection, segmentation, and pose estimation while reducing the pre-training cost to 100 ImageNet epochs.

2 Related Work

Instance contrastive learning. Instance-level contrastive learning considers each image as a singleton, only one sample in a class (Bojanowski & Joulin, 2017), which considers two augmented views of the same image as positive to be pulled closer, and all other images negative to be pushed further apart. MemoryBank (Wu et al., 2018) stores previously-computed representation in a memory bank to compare instances based on noise contrastive estimation. MoCo (He et al., 2020) uses a momentum encoder to store representation in a temporal manner, allowing the dictionary to be large. SimCLR (Chen et al., 2020a;b) shows that memory bank is not necessary when the mini-batch size is large enough. SwAV (Caron et al., 2020) clusters the data while enforcing consistency between cluster assignments. Besides, SwAV adopts multi-crop data augmentation, which uses a mix of views with different resolutions in place of two full-resolution views. BYOL (Grill et al., 2020) and SimSiam (Chen et al., 2020a) explore directly maximizing the similarity between two views of one image without negative pairs. Despite the success of instance-level contrastive learning on ImageNet linear probing, instance-wise contrastive learning does not encode position information explicitly, treating all regions equally. In contrast, MCL views each subimage in the montage image as a singleton to explicitly encode image localization with high fidelity.

Dense Representation Learning. Dense representation learning predicts at the pixel level, compared with the instance contrastive learning. Recently, some self-supervised learning methods that learn at pixel level representation are proposed. ULDVR(Pinheiro et al., 2020) learns pixel-wise representation by forcing local features to remain constant over different view conditions. DCL(Wang et al., 2021) optimizes a pairwise contrastive similarity loss at the pixel level between two views of input images by the Hungarian matching strategy. Self-EMD(Liu et al., 2020) shares a similar basic idea, but updates the matching strategy to Earth Mover's distance(Rubner et al., 2000). PixPro (Xie et al., 2021b) matches the feature pixels by a hand-crafted decision rule. These matching strategies only implicitly map the localization in feature maps to the euclidean coordinate in the input image, but do not guarantee a precise feature target assignment. Different from the pixel-level label assignment, MCL assigns a positive sample by matching the image regions with different sizes in the montage image on multi-level feature maps. A scale-aware assignment strategy ensures the precision localization of each feature point.

Object-Level Representation Learning. Both single-stage detector and two-stage detector attend to a manageable number of candidate object regions and evaluate convolutional networks on each region. The regions of interest have a rectangular shape and come in different sizes. RoIAlign (He et al., 2017) is proposed to extract the features of particular regions on the convolutional feature maps. Following Fast-RCNN (Girshick, 2015), SoCo (Wei et al., 2021) selects the proposal bounding boxes generated from Selective Search (Uijlings et al., 2013) and applies RoIAlign to extract object features by constructing multiple augmented views, which is used for contrastive loss. DetCon (Hénaff et al., 2021) identifies object-based regions with the off-the-shelf approximate segmentation algorithms (Arbeláez et al., 2014; Felzenszwalb & Huttenlocher, 2004) to produce a semantic mask. The contrastive detection objective then pulls together pooled feature vectors from the same mask and pushes apart features from different masks and different images. Whereas, the segmentation mask and bounding box predicted by the off-the-shelf methods are not accurate enough, incurring pseudo-label noise, which leads to an inferior result on the non-object-centric dataset. In contrast, MCL constructs montage images and precisely annotates the localization of each component image. As a result, MCL maintains a high transfer performance when pre-trained on the COCO dataset.

3 METHOD

Our goal is to learn regional semantic representation and scale consistency without supervision while keeping a reasonable training epoch. Typically, instance-level SSL methods (He et al., 2020; Misra & Maaten, 2020) learn occlusion-invariant representations (Purushwalkam & Gupta, 2020). In this section, we show that the idea of instance discrimination can be applied at the region level for learning visual representations that generalize well on dense prediction tasks. As illustrated in Fig. 2, MCL constructs multiple augmented views in different sizes and produces a multi-scale feature representation in which a contrastive loss is applied across the levels. The montage assembly guarantees the precision of pseudo bounding box label, which explicitly encodes the absolute position information. To enhance localization representation learning, we further introduce positional embedding, injected into the stem feature. To align the pretext task and downstream tasks, all the network modules in the downstream model are pre-trained to get a well-initialized representation ability. We also extend our pretext task to supervised pre-training, whose details can be found in Appendix. A.

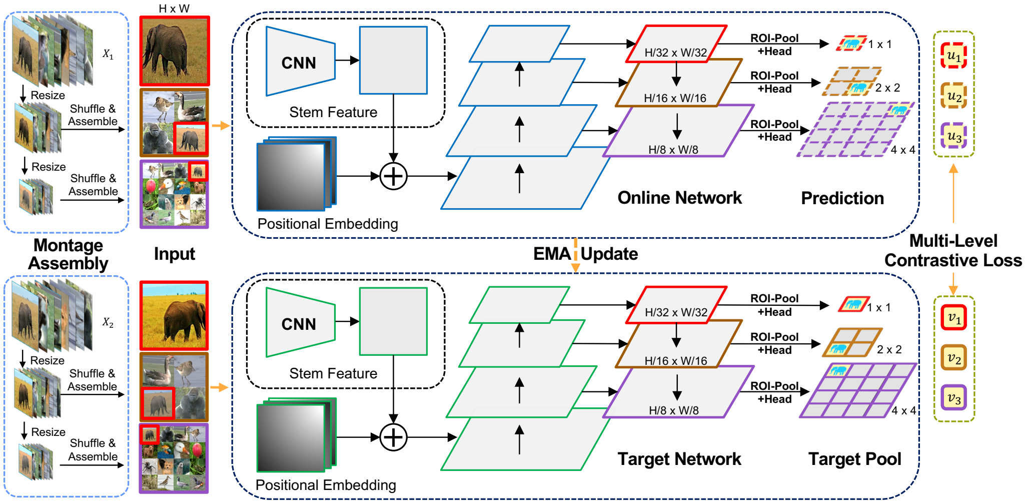

Figure 2: Overview of our method. This figure illustrates MCL with a model, whose FPN contains 3 levels. The image batch X is processed via the same augmentation pipeline with different random seeds. The images are downsampled by a factor of 2 and shuffled to construct montage input. The subfigures are multi-scale and precisely localized. The Positional Embedding is injected into the stem feature by addition operation. Stem feature is the feature map whose stride is 4. For ResNet-50, the stem feature is the feature map after the first max-pooling module. The feature pyramid is further ROI-pooled according to the subfigure location. Contrastive learning is performed on multi-level features to learn semantic regional representations via scale consistency regularization. More details of Multi-Level Contrastive Loss can be found in Fig. 3. The target network is not optimized by gradient and updated by the online network in EMA manner.

3.1 Montage Assembly

Algorithm 1 Montage Pseudo Code

```
# s: the level of downsampling ratio
# x: the input images batch with shape of (B, C, H, W)
ratio = pow(2, s)
x_aug = aug(x) # data augmentation
x_aug_ds = interpolate(x_aug, scale_factor = 1. / ratio)
x_aug_ds = shuffle(x_aug_ds)
B_ds = B / ratio / ratio
H_ds, W_ds = H / ratio, W / ratio
x_aug_ds = x_aug_ds.reshape(B_ds, ratio, ratio, C, H_ds, W_ds)
x_aug_ds = x_aug_ds.permute(0, 3, 1, 4, 2, 5)
x_aug_ds = x_aug_ds.reshape(B, C, H, W)
```

The photomontage is the process and the result of making a composite photograph by cutting, gluing, rearranging and overlapping two or more photographs into a new image. An interesting observation is that montage assembles images in different scales at the specific locations, therefore, montage assembly explicitly encodes the position and scale information. The image batch X is processed via the same augmentation pipeline with different random seeds. Towards handling the large scale variation, the resized images are downsampled to $\frac{1}{2^s}$ original size. The s ranges from $\{0, 1, 2, ..., S-1\}$, which also matches the level of feature maps in FPN. For encoding position and scale information, all the downsampled images with the same downsampling ratio are randomly combined to construct the montage image and all the new montage images have aligned shape with the original augmented images. The pseudo code is provided in Alg. 1 for clarity.

3.2 Multi-Level Contrastive Learning

Detectors with FPN assign anchor boxes of scale within a range to a specific pyramid level. Following this basic idea, we propose to extract features from FPN according to the downsampling ratio. Concretely, we assign the images with downsampling ratio of 2^s to P_{5-s} for a 3-level FPN architecture, where we denote the final feature set of FPN as $\{P_3, P_4, P_5\}$ from the finest resolution map to the coarsest resolution map. Similar to the RoI pooling operator, we map the rectangular window of the component images onto the FPN features. The dense prediction head is attached to further process the feature. As for the non-FPN framework, whose final features are single-level, we construct a feature pyramid by interpolating the final feature to the specific sizes.

The augmented views are encoded by two encoders, online network f_θ and target network $f_{\theta'}$, where the target network is implemented as an exponential moving average (EMA) of the online network.

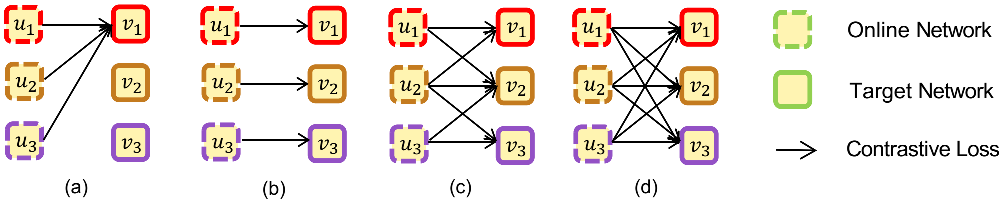

Figure 3: Details of Multi-Level Contrastive Loss. u_i and v_i are extracted from the i-th level feature pyramid. The arrow means to set the corresponding target feature as the positive sample and the other target features at the same level as negative samples.

The online network is attached with a projector g_θ and a predictor h_θ, while the target network is only appended with a projector $g_{\theta'}$. In summary, we represent each view pair as normalized latent features $\mathbf{u}_{s_i}$ and $\mathbf{v}_{s_i}$, where

$$\mathbf{u}_{s_i} = h_\theta \circ g_\theta \circ f_\theta(\mathbf{I}_{s_i}), \qquad \mathbf{v}_{s_i} = g_{\theta'} \circ f_{\theta'}(\mathbf{I}'_{s_i}). \tag{1}$$

We adopt the contrastive loss function in the form of InfoNCE (Van den Oord et al., 2018):

$$\mathcal{L}_{\mathbf{u}} = -\log \frac{\exp(\mathbf{u} \cdot \mathbf{v}^+/\tau))}{\exp(\mathbf{u} \cdot \mathbf{v}^+/\tau) + \sum_{\mathbf{v}^-} \exp(\mathbf{u} \cdot \mathbf{v}^-/\tau)}, \tag{2}$$

where the subscript of latents are omitted for simplicity, $\mathbf{v}^+$ is the target network's output on the same subimage as $\mathbf{u}$ and the set $\{\mathbf{v}^-\}$ is composed of target network's outputs from other subimages. τ is a temperature hyper-parameter (Touvron et al., 2021) for l_2-normalized latent features. As the number of latent features is sufficiently large, we use the negative samples co-existing in the same batch, following (Bachman et al., 2019; Chen et al., 2020a; Hjelm et al., 2018; Ye et al., 2019). Besides, we adopt a symmetric loss (Caron et al., 2020; Chen & He, 2021; Grill et al., 2020): $\mathcal{L} = \mathcal{L}_{\mathbf{u}} + \mathcal{L}_{\mathbf{v}}$.

3.3 Multi-Level Contrastive Loss

Multi-scale samples are generated in the montage stage, so we propose a series of modes to construct the final loss. Specifically, we design four matching strategies for assigning both the positive and the negative samples to the online features. As shown in Fig. 3, (a) All the images in different sizes target the view in the largest shape, (b) Each image level aims to pull close features from the counterpart level, (c) Latent features match the features from the adjacent levels, and (d) A dense connection is applied to all levels, treating all image resolution equally. The empirical study and comparison are provided in Sec.4.3.

3.4 Positional Embedding

Zero padding allows CNNs to encode absolute position information implicitly. However, dense prediction tasks require the precise localization of targets. Therefore, we introduce three positional embeddings: Learnable Positional Embedding (LPE), Cartesian Spatial Grid (CSG) and Sinusoidal Positional Embedding (SPE), to enhance localization representation learning. The details can be found in Appendix. A.

4 Experiments

In this section, we perform a series of experiments to evaluate our pre-training mechanism on dense prediction tasks, *e.g.*, COCO (Lin et al., 2014) detection, instance segmentation, pose estimation, Cityscapes segmentation (Cordts et al., 2016) and LVIS (Gupta et al., 2019) long tail object detection and segmentation.

4.1 Pre-training Setup

We pre-train MCL on ImageNet-1K (Deng et al., 2009) and COCO (Lin et al., 2014) dataset with LARS (You et al., 2017) optimizer and a batch size of 4096. All the models are pre-trained by default for 100 epochs on the ImageNet training set (about 1.28 million images). The training cost

Table 1: Comparison with state-of-the-art methods on COCO *val* set. All the models are pre-trained on COCO dataset and finetuned with Mask-RCNN following 1x schedule (Wu et al., 2019).

Methods	AP^{bb}	AP^{bb}_{50}	AP^{bb}_{75}
Supervised	38.9	59.6	42.7
BYOL Grill et al. (2020)	39.3(+0.4)	59.0(-0.6)	42.8(+0.1)
DenseCL Wang et al. (2021)	39.8(+0.9)	59.7(+0.1)	43.3(+0.6)
Self-EMD Liu et al. (2020)	40.4(+1.3)	61.1(+1.5)	43.7(+1.0)
SoCo Wei et al. (2021)	40.6(+1.5)	61.1(+1.5)	44.4(+1.7)
MCL	**41.8**(+2.9)	**62.1**(+2.5)	**45.8**(+3.1)
Methods	AP^{mk}	AP^{mk}_{50}	AP^{mk}_{75}
Supervised	35.4	56.5	38.1
BYOL Grill et al. (2020)	-	-	-
DenseCL Wang et al. (2021)	35.8(+0.4)	56.6(+0.1)	38.6(+0.5)
Self-EMD Liu et al. (2020)	-	-	-
SoCo Wei et al. (2021)	36.4(+1.0)	58.1(+1.6)	38.1(+0.0)
MCL	**37.7**(+2.3)	**59.3**(+2.8)	**40.5**(+2.4)

Table 2: Results on COCO for RetinaNet. All the models are pre-trained on ImageNet and finetuned on COCO with 1x schedule. MCL outperforms all the other state-of-the-art methods.

Methods	Epoch	AP^{bb}	AP^{bb}_{50}	AP^{bb}_{75}
Rand Init	-	24.5	39.0	25.7
Supervised	90	37.4	56.5	39.7
InsDis Wu et al. (2018)	200	35.5	54.1	38.2
PIRL Misra & Maaten (2020)	200	35.7	54.2	38.4
MoCo He et al. (2020)	200	36.3	55.0	39.0
MoCo v2 He et al. (2020)	200	37.2	56.2	39.6
InfoMin Tian et al. (2020)	200	38.1	57.3	40.9
SwAV Caron et al. (2020)	400	36.5	56.4	38.8
PixPro Xie et al. (2021b)	100	37.9	56.7	40.5
SoCo Wei et al. (2021)	100	38.2	57.4	40.9
InsLoc Yang et al. (2021)	200	36.4	55.3	39.0
DenseCL Wang et al. (2021)	200	37.6	56.3	40.3
MCL	100	**39.1**	**58.5**	**41.8**

Table 3: Comparison with SOTA methods on COCO by using Mask R-CNN. All the detectors are evaluated on COCO *val* 2017 set. “-” means that the results are missing in the source paper. MCL outperforms all the other SOTA SSL methods while significantly reducing the training epochs.

Methods	Epoch	1× Schedule						2× Schedule					
		AP^{bb}	AP^{bb}_{50}	AP^{bb}_{75}	AP^{mk}	AP^{mk}_{50}	AP^{mk}_{75}	AP^{bb}	AP^{bb}_{50}	AP^{bb}_{75}	AP^{mk}	AP^{mk}_{50}	AP^{mk}_{75}
Rand Init	-	31.0	49.5	33.2	28.5	46.8	30.4	38.4	57.5	42.0	34.7	54.8	37.2
Supervised	90	38.9	59.6	42.7	35.4	56.5	38.1	41.3	61.3	45.0	37.3	58.3	40.3
MoCo He et al. (2020)	200	38.5	58.9	42.0	35.1	55.9	37.7	40.8	61.6	44.7	36.9	58.4	39.7
MoCo v2 Chen et al. (2020c)	200	40.4	60.2	44.2	36.4	57.2	38.9	41.7	61.6	45.6	37.6	58.7	40.5
InfoMin Tian et al. (2020)	200	40.6	60.6	44.6	36.7	57.7	39.4	42.5	62.7	46.8	38.4	59.7	41.4
BYOL Grill et al. (2020)	300	40.4	61.6	44.1	37.2	58.8	39.8	42.3	62.6	46.2	38.3	59.6	41.1
SwAV Caron et al. (2020)	400	-	-	-	-	-	-	42.3	62.8	46.3	38.2	60.0	41.0
SoCo Wei et al. (2021)	100	42.3	62.5	46.5	37.6	59.1	40.5	43.2	63.3	47.3	38.8	60.6	41.9
DCL Wang et al. (2021)	200	40.3	59.9	44.3	36.4	57.0	39.2	41.2	61.9	45.1	37.3	58.9	40.1
ReSim Xiao et al. (2021)	200	39.8	60.2	43.5	36.0	57.1	38.6	41.4	61.9	45.4	37.5	59.1	40.3
DetCon Hénaff et al. (2021)	300	42.0	-	-	37.8	-	-	-	-	-	-	-	-
PixPro Xie et al. (2021b)	400	41.4	61.6	45.4	-	-	-	-	-	-	-	-	-
MCL	**100**	**42.5**	**62.8**	**46.9**	**38.2**	**59.8**	**41.2**	**43.4**	**63.6**	**47.5**	**39.1**	**60.8**	**41.9**

is comparable with supervised pre-training. For non-object-centric datasets, models are optimized for 530 epochs on the COCO training set and unlabeled set (about 241 thousand images) to match the training iteration on ImageNet. We employ the same data augmentation pipeline of BYOL (Grill et al., 2020), which is composed of random crop augmentation, random horizontal flip, color distortion, Gaussian blur, grayscaling and the solarization operation. All the component images are augmented separately with different random seeds but share an augmentation pipeline. The learning rate is linearly warmed up at the first 10 epochs and cosine annealed during the remaining epochs. The learning rate is set based on the batch size: $lr = 1.0 \times BatchSize/256$ and the weight decay is set to $1e^{-5}$. The weights of the target network are updated with a momentum coefficient m, starting from 0.99 and increased to 1 in the cosine scheduler same as (Chen et al., 2021; Grill et al., 2020).

4.2 Downstream Tasks

Pre-training on Non-object-centric Dataset. ImageNet is an object-centric dataset, which introduces dataset biases into pre-training and costs more efforts to collect than non-iconic images. As indicated in Tab. 1 and Tab. 3, most of the SOTA methods yield an inferior result when pre-trained on the COCO dataset, compared with the results on ImageNet dataset. MCL still obtains a large improvement, 1.4 AP^{bb}/1.3 AP^{mk} over the previous SOTA method, SoCo (Wei et al., 2021). This result demonstrates that MCL is robust to dataset and benefits mainly from the scale-invariance and precise localization representation rather than the dataset bias. Besides, the results manifest that pixel-level SSL methods and object-level methods fail in the non-iconic scenario.

COCO Object Detection and Instance Segmentation. Object detection and instance segmentation require simultaneous object location and classification while handling large variance of object size. We adopt Mask-RCNN (He et al., 2017) and RetinaNet (Lin et al., 2017) with ResNet-50 FPN backbone as detectors to evaluate the models pre-trained on ImageNet and COCO dataset.

Table 4: Semi-Supervised one-stage detection fine-tuned on COCO 1%, 5% and 10% data. All methods **except** MCL are pre-trained 200 epochs on ImageNet. MCL is per-trained for 100 epochs.

Methods	1% Data			5% Data			10% Data		
	AP	AP_{50}	AP_{75}	AP	AP_{50}	AP_{75}	AP	AP_{50}	AP_{75}
Rand Init	1.4	3.5	1.0	3.6	7.4	3.0	3.7	7.5	3.2
Supervised	8.2	16.2	7.2	16.5	30.3	15.9	19.6	34.5	19.7
MoCoHe et al. (2020)	7.0(-1.2)	13.5(-2.7)	6.5(-0.7)	15.0(-1.5)	27.0(-3.3)	14.9(-1.0)	18.2(-1.4)	31.6(-2.9)	18.4(-1.3)
MoCo v2Chen et al. (2020c)	8.4(+0.2)	15.8(-0.4)	8.0(+0.8)	16.8(+0.3)	29.6(-0.7)	16.8(+0.9)	20.0(+0.4)	34.3(-0.2)	20.2(+0.5)
DetCoXie et al. (2021a)	9.9(+1.7)	19.3(+3.1)	9.1(+1.9)	18.7(+2.2)	32.9(+2.6)	18.7(+2.8)	21.9(+2.3)	37.6(+3.1)	22.3(+2.6)
MCL	**12.1**(+3.9)	**22.6**(+6.4)	**11.6**(+4.4)	**20.7**(+4.2)	**35.6**(+5.3)	**21.2**(+5.3)	**23.8**(+4.2)	**39.6**(+5.1)	**24.2**(+4.5)

Table 5: Transfer Learning on LVIS dataset using Mask R-CNN with R50-FPN trained for 180k iterations. MCL significantly improves the performance on **rare** categories by 4.3 AP^{bb}/4.3 AP^{mk}. AP_r, AP_c and AP_f are the average precision of rare, common and frequent categories, respectively.

Methods	AP^{bb}	AP^{bb}_{50}	AP^{bb}_{75}	AP^{bb}_{r}	AP^{bb}_{c}	AP^{bb}_{f}	AP^{mk}	AP^{mk}_{50}	AP^{mk}_{75}	AP^{mk}_{r}	AP^{mk}_{c}	AP^{mk}_{f}
Supervised	23.9	37.2	25.5	10.2	21.8	32.2	23.1	35.4	24.3	11.1	21.6	30.1
MCL	**26.2**	**40.7**	**28.4**	**14.5**	**23.4**	**34.0**	**25.5**	**38.5**	**27.1**	**15.4**	**23.9**	**31.6**

As shown in Tab. 3, MCL outperforms the state-of-the-art (SOTA) unsupervised pre-training methods on the COCO 1x and 2x schedules with only 100 training epochs, achieving 42.5 AP^{bb}/38.2 AP^{mk} and 43.4 AP^{bb}/39.1 AP^{mk} on the 1x and 2x schedule, respectively. Our method surpasses the supervised counterpart by 3.6 AP^{bb} and 2.8 AP^{mk} on 1x schedule, showing that MCL accelerates the model converging on the downstream tasks. To verify the extendability of MCL, we conduct experiments on RetinaNet (Lin et al., 2017), which is a representative single-stage detector. We follow the standard COCO 1x schedule and include SyncBN in the backbone and FPN for a fair comparison. Tab. 2 shows that MCL exceeds the supervised baseline by 1.7 AP^{bb}.

Finetune in Low Data Regime. One of the purposes of pre-training is to improve the target task performance in a low data regime. Therefore, we conduct experiments on a mini version of the COCO dataset. Specifically, we randomly sample 1, 5 and 10% of COCO training data as the labeled dataset. To avoid overfitting, we finetune the detectors with 12k iterations. Other settings are the same as COCO 1x schedule. Tab. 4 indicates that MCL has strong generalization than other methods and outperform the supervised counterparts by about 4 AP. This result shows that MCL can be extended to semi-supervised learning for object detection as a consistency regularization.

Transfer Learning on LVIS Dataset. Compared with COCO, LVIS v1 dataset (Gupta et al., 2019) is more challenging due to the long tail distribution, which contains 1203 categories. To demonstrate the effectiveness and generality of our method, we finetune a Mask R-CNN model and follow the standard LVIS v1 1x training schedule, which is twice COCO detection training iterations. Tab. 5 shows that MCL significantly improves the performance on rare categories by 4.3 AP^{bb}/4.3 AP^{mk}, which is much larger than the improvement of common and frequent categories.

Cityscapes and COCO KeyPoint Dataset. To evaluate our method on other downstream tasks, we choose Cityscapes and COCO Keypoint dataset. Cityscapes is a dataset for autonomous driving in urban streets. We follow MoCo (He et al., 2020) to evaluate on instance segmentation with Mask R-CNN and to evaluate on semantic segmentation with Semantic FPN. For the COCO Keypoint dataset, we attach the standard keypoint head on Mask R-CNN. As shown in Tab. 6, MCL achieves 35.7 AP^{mk} on Cityscapes instance segmentation task, 76.1 mIoU on semantic segmentation task and 66.5 AP^{kp} on COCO Keypoint task. The superior results show that MCL is also suitable for other dense prediction tasks besides the detection task.

Supervised Pre-training on Transformer and CNN with MCL pretext task. It seems like a foregone conclusion that self-supervised pre-training surpasses the supervised counterpart on downstream tasks. However, we find that MCL pretext task facilitates the finetuning of supervised pre-training. For a fair comparison, we pre-train models with the same epochs as the normal supervised learning. Concretely, ResNet-50 (He et al., 2016) is trained with 100 epochs and Swin-T (Liu et al., 2021) is trained with 300 epochs. The other hyperparameters keep unchanged. The evaluation is still based on COCO 1x training schedule. Mask R-CNN with Swin-T is finetuned with MMDetection (Chen et al., 2019). Tab. 7 shows that MCL outperforms 3.2 AP^{bb}/2.4 AP^{mk} over supervised counterpart for ResNet-50. The results also demonstrates that MCL pretext task is effective for the state-of-the-art Swin-Transformer architecture, surpassing the baseline by 0.8 AP^{bb}/0.7 AP^{mk}.

Table 6: Results of Mask R-CNN and Semantic FPN on COCO Keypoint and Cityscapes dataset. The results demonstrate that MCL is available for other dense prediction tasks besides detection task.

Methods	Keypoint						Cityscapes		
	AP^{bb}	AP^{bb}_{50}	AP^{bb}_{75}	AP^{kp}	AP^{kp}_{50}	AP^{kp}_{75}	mIoU	AP^{mk}	AP^{mk}_{50}
Supervised	57.5	84.0	63.0	65.6	87.0	71.3	72.9	31.8	58.5
SoCo Wei et al. (2021)	58.0 (+0.5)	84.3(+0.3)	64.2(+1.2)	65.9(+0.3)	87.0(+0.0)	71.8(+0.5)	74.5(+1.6)	34.7(+2.9)	63.0(+4.5)
MCL	**58.4**(+0.9)	**84.7**(+0.7)	**64.2**(+1.2)	**66.5**(+0.9)	**87.3**(+0.3)	**72.8**(+1.5)	**76.1**(+3.2)	**35.7**(+3.9)	**63.9**(+5.4)

Table 7: Results on COCO dataset using Mask R-CNN with supervised pre-training. The results show that MCL can be extended to Swin-Transformer backbone. All the backbones are pre-trained for the same epochs. Vanilla means to pre-train model in a standard supervised manner.

Methods	ResNet-50						Swin-T					
	AP^{bb}	AP^{bb}_{50}	AP^{bb}_{75}	AP^{mk}	AP^{mk}_{50}	AP^{mk}_{75}	AP^{bb}	AP^{bb}_{50}	AP^{bb}_{75}	AP^{mk}	AP^{mk}_{50}	AP^{mk}_{75}
Vanilla	38.9	59.6	42.7	35.4	56.6	38.1	43.9	65.3	48.5	39.6	62.3	42.5
MCL	**42.1**	**62.6**	**45.9**	**37.8**	**59.5**	**40.6**	**44.7**	**66.0**	**49.0**	**40.3**	**62.8**	**43.4**

4.3 Ablation Study

Montage Downsampling Level. Feature pyramid network is a default component in the most mainstream detectors and our method is based on montage assembly over the multi-level downsampled images and feature pyramid. So we investigate the effect of the object scale variance. The first line in Tab. 8a is the result of MoCo v3 pre-trained by 100 epochs on COCO 2017 *val* set. The performance is improved as more downsampled rates are included, which means that object detectors benefit from scale-invariant representation. By the comparison between the penultimate and the last line in Tab. 8a, we find that the representation of fine-grained objects is important for the COCO object detection dataset, in which about 41% of objects are small.

Multi-Level Contrastive Loss. As shown in Fig. 3, we propose a series of meaningful positive pair matching strategies. The loss indicators in Tab. 8b are same as those in Fig. 3. We find that setting the images with the largest resolution as positive pair samples leads to the best result. This result is reasonable because a higher resolution typically yields a better representation. The reason why *b* loss mode is inferior can be that the supervision from the counterpart level lacks semantic information for the small component images. The result of *c* loss mode is slightly better than *b* mode due to the feature matching across levels. *d* loss mode yields a lower result than *c*. We conjecture that the representation from the low-resolution image is inferior to the high-resolution image.

Positional Embedding. Tab. 8c examines the importance of positional embedding and evaluates the quality of each type. We pre-train the model with positional embedding injected but finetune detectors without positional embedding for a fair comparison. SPE achieves the best result, surpassing the baseline by 0.5 AP^{bb}/0.5 AP^{mk}. The performance of CSG is almost equivalent to SPE, meaning that the hand-crafted positional embeddings contain sufficient absolute position information to boost the localization ability. LPE introduces additional learnable parameters but yields a lower performance than baseline. The learnable parameters are crucial for the features learned by the backbone.

Weight Decay. Weight norm is an important factor in the alignment between pre-training and finetuning. We empirically demonstrate this conclusion by modifying the weight decay hyperparameter, which influences the weight norm of the converged model. Typically, the weight decay is set to $1e^{-6}$ for LARS optimizer, which is widely adopted in unsupervised pre-training works (Caron et al., 2020; Chen et al., 2020a; 2021; Grill et al., 2020). We set the weight decay from $1.5e^{-6}$ to $1e^{-5}$ to evaluate the effect. The results in Tab. 8d show that a large weight decay leads to a superior result. To explicitly verify the conclusion, we simply divide the non-normalization layer parameters by a fixed number to downscale the weight norm. The results show that reducing the weight norm of a model pre-trained with a small weight decay leads to a non-negligible improvement. Normalization techniques exist in many mainstream models (Dosovitskiy et al., 2020; He et al., 2016; Huang et al., 2017; Liu et al., 2021; Tolstikhin et al., 2021; Sandler et al., 2018; Wu & He, 2018), which makes output resilient to the parameter scale, we take Batch Normalization (Ioffe & Szegedy, 2015) for example: $\mathrm{BN}(Wx) = \mathrm{BN}((\alpha W)x)$, and we can show that: $\frac{\partial \mathrm{BN}((\alpha W)x)}{\partial \alpha W} = \frac{1}{\alpha} \cdot \frac{\partial \mathrm{BN}(Wx)}{\partial W}$, where α is a positive scalar. In the case that $\alpha < 1$, the gradient of parameter W is magnified. Following the SGD update rule, the model weight of $t + 1$ step is $W_{t+1} = W_t - \eta \frac{1}{\alpha} \frac{\partial \mathrm{BN}(Wx)}{\partial W}$, where η is the learning rate. Suppose that the learning rate is suitable and the weight is well-initialized, the

Table 8: Ablation studies on COCO for the proposed MCL method. All the models are trained on ImageNet dataset for 100 epochs. The loss indicators in (b) are same as those in Fig. 3. Div in (d) means all the non-normalization layer parameters are divided by a fixed number. In (e), B means that only the backbone is pre-trained, F indicates FPN neck and H is the detection head. The results are reported with Mask R-CNN in all tables except (f), in which the results of RetinaNet are provided.

(a) Study on Downsampling Level.

Level	$\mathrm{AP}^{\mathrm{bb}}$	$\mathrm{AP}^{\mathrm{bb}}_{50}$	$\mathrm{AP}^{\mathrm{bb}}_{75}$	$\mathrm{AP}^{\mathrm{mk}}$	$\mathrm{AP}^{\mathrm{mk}}_{50}$	$\mathrm{AP}^{\mathrm{mk}}_{75}$
1	40.7	60.9	44.6	36.8	58.0	39.8
2	41.4	61.6	45.4	37.3	58.6	39.9
3	41.8	61.7	45.5	37.6	58.8	40.2
4	42.5	62.8	46.9	38.2	59.8	41.2

(b) Study on Multi-Level Loss.

Loss	$\mathrm{AP}^{\mathrm{bb}}$	$\mathrm{AP}^{\mathrm{bb}}_{50}$	$\mathrm{AP}^{\mathrm{bb}}_{75}$	$\mathrm{AP}^{\mathrm{mk}}$	$\mathrm{AP}^{\mathrm{mk}}_{50}$	$\mathrm{AP}^{\mathrm{mk}}_{75}$
a	42.5	62.8	46.9	38.2	59.8	41.2
b	41.8	61.8	45.7	37.4	58.8	40.2
c	42.0	62.2	46.0	37.8	59.2	40.8
d	41.0	61.5	44.6	37.1	58.6	40.0

(c) Study on Position Embedding.

PE	$\mathrm{AP}^{\mathrm{bb}}$	$\mathrm{AP}^{\mathrm{bb}}_{50}$	$\mathrm{AP}^{\mathrm{bb}}_{75}$	$\mathrm{AP}^{\mathrm{mk}}$	$\mathrm{AP}^{\mathrm{mk}}_{50}$	$\mathrm{AP}^{\mathrm{mk}}_{75}$
None	42.0	62.3	46.3	37.7	59.5	40.5
LPE	39.4	59.5	42.4	34.4	55.9	36.7
CSG	42.2	62.3	46.3	37.8	59.3	40.8
SPE	42.5	62.8	46.9	38.2	59.8	41.2

(d) Study on Weight Decay.

Weight Decay	$\mathrm{AP}^{\mathrm{bb}}$	$\mathrm{AP}^{\mathrm{bb}}_{50}$	$\mathrm{AP}^{\mathrm{bb}}_{75}$	$\mathrm{AP}^{\mathrm{mk}}$	$\mathrm{AP}^{\mathrm{mk}}_{50}$	$\mathrm{AP}^{\mathrm{mk}}_{75}$
1.5e-6	41.7	62.2	45.9	37.8	59.3	40.6
1.5e-6 Div 1.5	41.8	62.3	45.9	37.8	59.3	40.6
1.5e-6 Div 2	42.2	62.4	46.4	37.8	59.4	40.6
5e-6	42.3	62.6	46.7	38.0	59.7	40.8
1e-5	42.5	62.8	46.9	38.2	59.8	41.2

(e) Study on Architecture Alignment.

Arch.	$\mathrm{AP}^{\mathrm{bb}}$	$\mathrm{AP}^{\mathrm{bb}}_{50}$	$\mathrm{AP}^{\mathrm{bb}}_{75}$	$\mathrm{AP}^{\mathrm{mk}}$	$\mathrm{AP}^{\mathrm{mk}}_{50}$	$\mathrm{AP}^{\mathrm{mk}}_{75}$
B	41.1	61.3	45.4	37.2	58.6	39.9
B+F	41.5	61.6	45.5	37.4	58.6	40.1
B+F+H	42.5	62.8	46.9	38.2	59.8	41.2

(f) Study on Training Epoch.

Epoch	$\mathrm{AP}^{\mathrm{bb}}$	$\mathrm{AP}^{\mathrm{bb}}_{50}$	$\mathrm{AP}^{\mathrm{bb}}_{75}$	$\mathrm{AP}^{\mathrm{bb}}_{s}$	$\mathrm{AP}^{\mathrm{bb}}_{m}$	$\mathrm{AP}^{\mathrm{bb}}_{l}$
100	39.1	58.5	41.8	26.5	43.7	47.3
200	39.5	59.2	42.7	25.8	44.2	47.9
400	39.9	59.8	42.7	26.7	44.4	48.3

relatively small weight norm leads to a faster convergence, compared with the large model weight. This observation also supports the conclusion of (Erhan et al., 2010) that unsupervised pre-training, serving as a strong regularization, guides the learning towards basins of attraction of minima that support better generalization from the training data set.

Architecture Alignment. We ablate each architecture component step by step to verify the importance of alignment of the downstream and pre-train model architecture. Tab. 8e reports the studies, in which the baseline achieves 41.1 $\mathrm{AP}^{\mathrm{bb}}$ / 37.2 $\mathrm{AP}^{\mathrm{mk}}$. FPN neck further improves the performance to 41.5 $\mathrm{AP}^{\mathrm{bb}}$ / 37.4 $\mathrm{AP}^{\mathrm{mk}}$ and Detection Head finally improves the result to 42.5 $\mathrm{AP}^{\mathrm{bb}}$ / 38.2 $\mathrm{AP}^{\mathrm{mk}}$. Pre-training detection head leads to additional gain for Mask R-CNN, while MCL also outperforms other state-of-the-art methods on RetinaNet, which has a different detection head and FPN architecture from Mask R-CNN. This phenomenon demonstrates that MCL is robust to model architecture.

Training Epochs. Self-supervised learning typically benefits from long training epochs. Following this empirical conclusion, we extend the training epochs to 200 epochs and 400 epochs on the ImageNet dataset. We finetune the pre-trained model using RetinaNet with the standard 1x COCO schedule. Pre-trained for 200 epochs, MCL improves the detection result to 39.5 AP. Another 200 training epochs increase the performance by 0.4 AP, which means that a long training schedule further improves the performance.

5 Conclusion

In this work, we introduce a novel self-supervised framework based on multi-level contrastive learning. Our method learns regional representation for precise localization, scale consistency among multi-scale crops and semantic global representations. The montage assembly explicitly encodes absolute position and scale information. Multi-level contrastive learning aligns the feature map with the image region and regularizes the scale consistency. Positional embedding further enhances the localization capability without introducing additional parameters and computational cost during finetuning. Besides, we empirically explore the alignment between pre-training and finetuning by investigating the interactions of weight norm and pretext task with transfer performance. Our experiment results demonstrate the state-of-the-art transfer performance on various dense prediction tasks. The success of applying our pretext task in a supervised learning scenario proves the importance of task alignment. A further fine-grain representation learning under our framework may lead to a promising result.

References

Pablo Arbeláez, Jordi Pont-Tuset, Jonathan T Barron, Ferran Marques, and Jitendra Malik. Multiscale combinatorial grouping. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 328–335, 2014.

Philip Bachman, R Devon Hjelm, and William Buchwalter. Learning representations by maximizing mutual information across views. *Advances in neural information processing systems*, 32, 2019.

Piotr Bojanowski and Armand Joulin. Unsupervised learning by predicting noise. *arXiv preprint arXiv:1704.05310*, 2017.

Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *European conference on computer vision*, pp. 213–229. Springer, 2020.

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.

Kai Chen, Jiaqi Wang, Jiangmiao Pang, Yuhang Cao, Yu Xiong, Xiaoxiao Li, Shuyang Sun, Wansen Feng, Ziwei Liu, Jiarui Xu, et al. Mmdetection: Open mmlab detection toolbox and benchmark. *arXiv preprint arXiv:1906.07155*, 2019.

Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pp. 1597–1607. PMLR, 2020a.

Ting Chen, Simon Kornblith, Kevin Swersky, Mohammad Norouzi, and Geoffrey E Hinton. Big self-supervised models are strong semi-supervised learners. *Advances in neural information processing systems*, 33:22243–22255, 2020b.

Xinlei Chen and Kaiming He. Exploring simple siamese representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 15750–15758, 2021.

Xinlei Chen, Haoqi Fan, Ross Girshick, and Kaiming He. Improved baselines with momentum contrastive learning. *arXiv preprint arXiv:2003.04297*, 2020c.

Xinlei Chen, Saining Xie, and Kaiming He. An empirical study of training self-supervised vision transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 9640–9649, 2021.

Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. The cityscapes dataset for semantic urban scene understanding. In *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pp. 248–255. Ieee, 2009.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

Dumitru Erhan, Aaron Courville, Yoshua Bengio, and Pascal Vincent. Why does unsupervised pre-training help deep learning? In *Proceedings of the thirteenth international conference on artificial intelligence and statistics*, pp. 201–208. JMLR Workshop and Conference Proceedings, 2010.

Pedro F Felzenszwalb and Daniel P Huttenlocher. Efficient graph-based image segmentation. *International journal of computer vision*, 59(2):167–181, 2004.

Ross Girshick. Fast r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pp. 1440–1448, 2015.

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.

Agrim Gupta, Piotr Dollar, and Ross Girshick. Lvis: A dataset for large vocabulary instance segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 5356–5364, 2019.

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

Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pp. 2961–2969, 2017.

Kaiming He, Ross Girshick, and Piotr Dollár. Rethinking imagenet pre-training. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 4918–4927, 2019.

Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9729–9738, 2020.

Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. *arXiv preprint arXiv:2111.06377*, 2021.

Olivier J Hénaff, Skanda Koppula, Jean-Baptiste Alayrac, Aaron van den Oord, Oriol Vinyals, and João Carreira. Efficient visual pretraining with contrastive detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 10086–10096, 2021.

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

Gao Huang, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q Weinberger. Densely connected convolutional networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 4700–4708, 2017.

Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *International conference on machine learning*, pp. 448–456. PMLR, 2015.

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 *European conference on computer vision*, pp. 740–755. Springer, 2014.

Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. In *Proceedings of the IEEE international conference on computer vision*, pp. 2980–2988, 2017.

Songtao Liu, Zeming Li, and Jian Sun. Self-emd: Self-supervised object detection without imagenet. *arXiv preprint arXiv:2011.13677*, 2020.

Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 10012–10022, 2021.

Ishan Misra and Laurens van der Maaten. Self-supervised learning of pretext-invariant representations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 6707–6717, 2020.

Pedro O Pinheiro, Amjad Almahairi, Ryan Y Benmalek, Florian Golemo, and Aaron Courville. Unsupervised learning of dense visual representations. *arXiv preprint arXiv:2011.05499*, 2020.

Senthil Purushwalkam and Abhinav Gupta. Demystifying contrastive self-supervised learning: Invariances, augmentations and dataset biases. *Advances in Neural Information Processing Systems*, 33:3407–3418, 2020.

Limeng Qiao, Yuxuan Zhao, Zhiyuan Li, Xi Qiu, Jianan Wu, and Chi Zhang. Defrcn: Decoupled faster r-cnn for few-shot object detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 8681–8690, 2021.

Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. *Advances in neural information processing systems*, 28: 91–99, 2015.

Yossi Rubner, Carlo Tomasi, and Leonidas J Guibas. The earth mover's distance as a metric for image retrieval. *International journal of computer vision*, 40(2):99–121, 2000.

Mark Sandler, Andrew Howard, Menglong Zhu, Andrey Zhmoginov, and Liang-Chieh Chen. Mobilenetv2: Inverted residuals and linear bottlenecks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 4510–4520, 2018.

Kihyuk Sohn, Zizhao Zhang, Chun-Liang Li, Han Zhang, Chen-Yu Lee, and Tomas Pfister. A simple semi-supervised learning framework for object detection. *arXiv preprint arXiv:2005.04757*, 2020.

Yonglong Tian, Chen Sun, Ben Poole, Dilip Krishnan, Cordelia Schmid, and Phillip Isola. What makes for good views for contrastive learning? *Advances in Neural Information Processing Systems*, 33:6827–6839, 2020.

Ilya O Tolstikhin, Neil Houlsby, Alexander Kolesnikov, Lucas Beyer, Xiaohua Zhai, Thomas Unterthiner, Jessica Yung, Andreas Steiner, Daniel Keysers, Jakob Uszkoreit, et al. Mlp-mixer: An all-mlp architecture for vision. *Advances in Neural Information Processing Systems*, 34, 2021.

Hugo Touvron, Matthieu Cord, Matthijs Douze, Francisco Massa, Alexandre Sablayrolles, and Hervé Jégou. Training data-efficient image transformers & distillation through attention. In *International Conference on Machine Learning*, pp. 10347–10357. PMLR, 2021.

Jasper RR Uijlings, Koen EA Van De Sande, Theo Gevers, and Arnold WM Smeulders. Selective search for object recognition. *International journal of computer vision*, 104(2):154–171, 2013.

Aaron Van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv e-prints*, pp. arXiv–1807, 2018.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

Xinlong Wang, Rufeng Zhang, Chunhua Shen, Tao Kong, and Lei Li. Dense contrastive learning for self-supervised visual pre-training. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 3024–3033, 2021.

Fangyun Wei, Yue Gao, Zhirong Wu, Han Hu, and Stephen Lin. Aligning pretraining for detection via object-level contrastive learning. *Advances in Neural Information Processing Systems*, 34, 2021.

Yuxin Wu and Kaiming He. Group normalization. In *Proceedings of the European conference on computer vision (ECCV)*, pp. 3–19, 2018.

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

Zhirong Wu, Yuanjun Xiong, Stella X Yu, and Dahua Lin. Unsupervised feature learning via nonparametric instance discrimination. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3733–3742, 2018.

Tete Xiao, Colorado J Reed, Xiaolong Wang, Kurt Keutzer, and Trevor Darrell. Region similarity representation learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 10539–10548, 2021.

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 *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 8392–8401, 2021a.

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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 16684–16693, 2021b.

Ceyuan Yang, Zhirong Wu, Bolei Zhou, and Stephen Lin. Instance localization for self-supervised detection pretraining. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 3987–3996, 2021.

Mang Ye, Xu Zhang, Pong C Yuen, and Shih-Fu Chang. Unsupervised embedding learning via invariant and spreading instance feature. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 6210–6219, 2019.

Yang You, Igor Gitman, and Boris Ginsburg. Large batch training of convolutional networks. *arXiv preprint arXiv:1708.03888*, 2017.

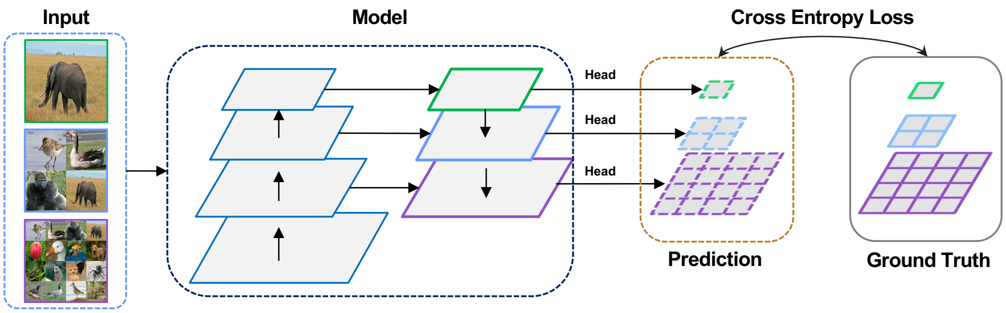

Figure 4: Overview of MCL applied in the supervised scenario. All the input images are augmented via the same pipeline with different seeds. The input images are downsampled and assembled at different levels. The ground truths are assembled in the same order as the montage images. The multi-level cross-entropy loss is applied to optimize the model.

A Appendix

A.1 Additional Implementation Details

Multi-Level Contrastive Loss. As described in Sec. 3.1, the images are downsampled according to the level index and assembled in a montage manner. The numbers of montage image in level i is $\frac{B}{2^i}$, where i starts from 0 to $S-1$ and B is the batch size. Therefore, the montage assembly increases the computational cost marginally and the upper bounder is twice the baseline batch size. Since the highest resolution images typically yield the best semantic representation and the empirical result in Sec. 4.3, the first level contrastive loss (on the highest resolution images) is important to the representation learning. As a consequence, we assign different loss weight to each level, $\frac{1}{2^{(i+1)}}$ for the i-th level. The detection head is a shared 4 *CONV* head without *fc* layer across levels, which are not loaded in the RetinaNet detector. We attach a global average pooling layer on the detection head to aggregate the features because averaging implicitly encourages a high response region. Both the projection head and prediction head are 2-layer MLPs whose hidden layer dimension is 2048. The final linear layer has a 256-dimension output and a final BN layer is attached to the projection head to accelerate the convergence.

Multi-Level Supervised Learning. We extend MCL to the supervised learning scenario to demonstrate the importance of the alignment between the pretext task and downstream tasks. As illustrated in Fig. 4, we generate S augmented views for the model, which are downsampled to $\frac{1}{2^s}$ original size. We set s as the level index, which starts from 0 to $S-1$. Different from self-supervised learning, we simply adopt the same optimizer as the normal setting, SGD optimizer for ResNet and AdamW optimizer for Swin-Transformer.

Positional Embedding. *Learnable Positional Embedding.* The stem block outputs feature maps with a shape of (C, H, W). We set up an x-axis and y-axis Learnable Positional Embedding (LPE) table, which contains H or W learnable $\frac{C}{2}$-dimension vectors separately. The embeddings in the tables are selected according to cartesian coordinates and concatenated together. *Cartesian Spatial Grid.* To remove the additional parameters, Cartesian Spatial Grid (CSG) is introduced as a coordinate indicator. The absolute coordinates are applied with linear scaling of both coordinate values to make them fall in the range $[-1, 1]$. The transformation between locations is $[2\delta_x/H, 2\delta_y/W]$, where δ_x and δ_y are the offsets in the unnormalized space. As the feature maps are 2-dimension, we repeat the tensor along the channel dimension and fuse the positional embedding by addition operation rather than concatenation, which requires extra channels instantiated and mismatches the downstream model. *Sinusoidal Positional Embedding.* Motivated by (Carion et al., 2020; Vaswani et al., 2017), Sinusoidal Positional Embedding (SPE) contains the relative and absolute position information. Similar to LPE, we concatenate the embeddings in two dimensions and each position is encoded by sine and cosine function of different frequencies:

$$PE_{(pos,2i)} = \sin(\omega_i \cdot pos), \qquad PE_{(pos,2i+1)} = \cos(\omega_i \cdot pos), \tag{3}$$

Table 9: Results of the long training schedule for RetinaNet finetuned on COCO with 90k, 180k, and 540k. MCL not only accelerates the convergence but also improves the final performance.

Methods	Epoch	1x schedule			2x schedule			6x schedule		
		AP	AP_{50}	AP_{75}	AP	AP_{50}	AP_{75}	AP	AP_{50}	AP_{75}
Supervised	90	37.4	56.6	39.7	38.8	58.7	41.2	39.2	58.6	42.1
MoCo v2	800	37.9(+0.5)	57.1(+0.5)	40.4(+0.7)	39.8(+1.0)	59.3(+0.6)	42.8(+1.6)	40.2(+1.0)	59.9(+1.3)	43.1(+1.0)
MCL	400	**39.9**(+2.5)	**59.8**(+3.2)	**42.7**(+3.0)	**41.2**(+2.4)	**61.1**(+2.4)	**44.0**(+2.8)	**41.4**(+2.2)	**61.1**(+2.5)	**44.5**(+2.4)

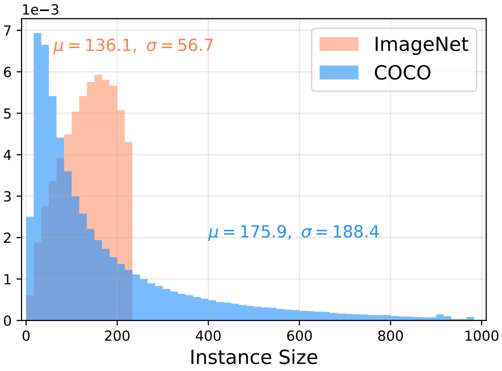

Figure 5: Instance Size Distribution. For the COCO dataset, all the images are resized to (1333, 800) shape. For the ImageNet dataset, all the images are resized to 224 × 224 to calculate the statistics.

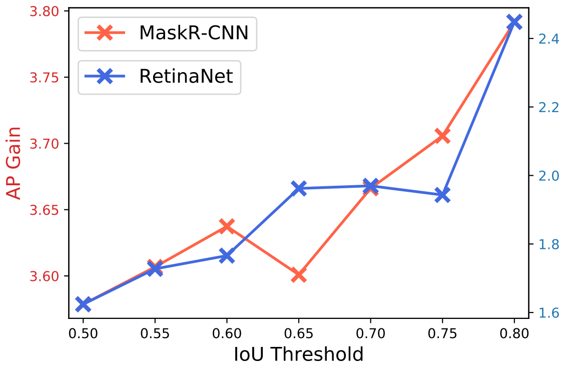

Figure 6: Score Distance Distribution. The AP^{bb} gain of MCL over the supervised counterpart. MCL performs better than the baseline at a higher IoU threshold, indicating that the MCL features have better localization capability.

where $\omega_i = 1/10000^{2i/d}$ and d is half of the feature channel dimension. The transformation between locations is only related to the position offsets:

$$\begin{bmatrix} \sin(\omega_i p) \\ \cos(\omega_i p) \end{bmatrix} = \begin{bmatrix} \cos(\omega_i \delta) & \sin(\omega_i \delta) \\ -\sin(\omega_i \delta) & \cos(\omega_i \delta) \end{bmatrix} \cdot \begin{bmatrix} \sin(\omega_i q) \\ \cos(\omega_i q) \end{bmatrix}, \tag{4}$$

where $\delta = p - q$ indicates the position offset. Different from CSG, SPE keeps a consistent transformation distance when extending to large scale feature maps.

A.2 Discussion

As discussed in Sec. 1, the scale of objects varies in a small range for ImageNet classification model, whereas the scale variation of MS-COCO dataset (Lin et al., 2014) is large across object instances for detectors. As shown in Fig. 5, the standard variance of the scale of instances in MS-COCO is 188.4, while that of ImageNet is 56.7. Typically, a high IoU means a high precision of prediction. Fig. 6 shows that MCL performs better than baseline at a high IoU threshold, which demonstrates that localization capability benefits from explicit position information.

A.3 Additional Results

Long Finetuning Schedule. The experiments in Sec. 4 mainly follow the 1x and 2x schedule, which are not long enough for detectors to be fully converged. We extend the training schedule to 6x schedule, *i.e.* 540k iterations. Tab. 9 shows that MCL pre-trained with 400 epochs achieves 41.2 AP^{bb} as 2x schedule is applied. MCL with 6x schedule still surpasses the supervised counterpart and MoCo v2 pre-trained with 800 epochs. These results prove that pre-training not only accelerates the convergence but also improves the final performance.

Mask R-CNN with C4 on COCO. As described in Sec. 3.1, MCL is compatible with the non-FPN framework. We construct a feature pyramid by interpolating the single-level feature to the specific sizes. The results in Tab. 10 show that MCL achieves SOTA results while significantly reducing the training epochs. Our method achieves a superior result with a 1x schedule and benefits from a long finetune schedule, *i.e.* 2x COCO schedule. We believe that the reason that MCL yields an inferior result on Mask R-CNN C4, compared with Mask R-CNN FPN, is that Mask R-CNN C4 has a lower performance on small object detection.

Table 10: Comparison with SOTA methods on COCO by using Mask R-CNN with R50-C4. All the detectors are evaluated on COCO *val* 2017 set. "-" means that the results are missing in the source paper. MCL achieves SOTA results while significantly reducing the training epochs.

Methods	Epoch	**1**× Schedule						**2**× Schedule					
		AP^{bb}	AP^{bb}_{50}	AP^{bb}_{75}	AP^{mk}	AP^{mk}_{50}	AP^{mk}_{75}	AP^{bb}	AP^{bb}_{50}	AP^{bb}_{75}	AP^{mk}	AP^{mk}_{50}	AP^{mk}_{75}
Rand Init	-	26.4	44.0	27.8	29.3	46.9	30.8	35.6	54.6	38.2	31.4	51.5	33.5
Supervised	90	38.2	58.2	41.2	33.3	54.7	35.2	40.0	59.9	43.1	34.7	56.5	36.9
MoCo He et al. (2020)	200	38.5	58.3	41.6	33.6	54.8	35.6	40.7	60.5	44.1	35.4	57.3	37.6
SimCLR Chen et al. (2020a)	200	-	-	-	-	-	-	39.6	59.1	42.9	34.6	55.9	37.1
MoCo v2 Chen et al. (2020c)	800	39.3	58.9	42.5	34.3	55.7	36.5	41.2	60.9	44.6	35.8	57.7	38.2
InfoMin Tian et al. (2020)	200	39.0	58.5	42.0	34.1	55.2	36.3	41.3	61.2	45.0	36.0	57.9	38.3
BYOL Grill et al. (2020)	300	-	-	-	-	-	-	40.3	60.5	43.9	35.1	56.8	37.3
SwAV Caron et al. (2020)	400	-	-	-	-	-	-	39.6	60.1	42.9	34.7	56.6	36.6
SimSiam Chen & He (2021)	200	39.2	59.3	42.1	34.4	56.0	36.7	-	-	-	-	-	-
PixPro Xie et al. (2021b)	400	40.5	59.8	44.0	-	-	-	-	-	-	-	-	-
SoCo Wei et al. (2021)	100	40.4	60.4	43.7	34.9	56.8	37.0	41.1	61.0	44.4	35.6	57.5	38.0
MCL	100	40.0	60.3	43.2	34.7	56.7	36.7	**41.7**	**61.7**	**45.4**	**36.1**	**58.1**	**38.5**

Table 11: Comparison with state-of-the-art self-supervised learning methods on ImageNet-1K **linear evaluation** with the ResNet-50 backbone. Table (a) demonstrates that MCL outperforms SoCo, a state-of-the-art self-supervised pre-training for object detection. Table (b) shows that it is a trade-off between the performance of the upstream task and downstream tasks.

(a) Self-supervised learning for dense prediction.

Methods	Epoch	Top-1	Top-5
SoCo (C4)	100	59.7	82.8
SoCo (C4)	400	62.6	84.6
SoCo (FPN)	100	53.0	77.5
SoCo (FPN)	400	54.2	79.5
SoCo* (FPN)	400	53.9	79.2
MCL	100	69.9	88.9
MCL	400	**71.5**	**89.9**

(b) Self-supervised learning for classification.

Methods	Epoch	Top-1	Top-5
Supervised	90	76.5	-
MoCo He et al. (2020)	200	60.6	-
SimCLR Chen et al. (2020a)	1000	69.3	89.0
MoCo v2 Chen et al. (2020c)	800	71.1	-
InfoMin Tian et al. (2020)	800	73.0	91.1
BYOL Grill et al. (2020)	1000	74.3	91.6
SwAV Caron et al. (2020)	800	75.3	-
SimSiam Chen & He (2021)	800	71.3	-

Linear Evaluation on ImageNet-1K. MCL learns global semantic representation besides scale consistency and regional localization. We present the ImageNet-1K linear evaluation results for reference. Following the common setting (Caron et al., 2020; Grill et al., 2020; He et al., 2020), data augmentation contains random crop with resize of 224 × 224 pixels and random flip. Only the backbone network parameters are loaded and frozen. The classification head is trained for 100 epochs, using an SGD optimizer with a momentum of 0.9 and a batch size of 256. The learning rate starts with 10 and the weight decay is 0. In the test phase, the data augmentation is a center crop from a resized 256 × 256 image. Tab. 11 shows that MCL surpasses SoCo on ImageNet linear evaluation, learning a semantic global representation. Compared with the self-supervised learning methods for image classification, MCL outperforms them on dense prediction tasks, while underperforms some of them on the linear evaluation. This phenomenon shows that the improvement on upstream task does not guarantee a better transfer performance on downstream tasks, due to the task misalignment.