A. Appendix

A.1. Experimentation Details

Pre-training. The initial learning rate is set as 0.03 and follows a cosine decaying schedule [36]. The weight decay is 0.0001, SGD momentum is 0.9, and the augmentations are the same as [6]. We use 1.0 for loss balancing term λ. We use a momentum queue of 65,536 for IN-1K experiments, and of 16,384 for IN-100 experiments. Table 7 details the exact pre-training parameters used for ReSim, where we followed the default training parameters from [6, 7]. IN-100 experiments followed the exact same set of parameters except training occurred for 500 epochs with moco-k of 16,384 on IN-100, instead of 200 epochs and moco-k of 65,536 as was done on IN-1K (we used a smaller queue as IN-100 has fewer images).

Object detection. The R50-C4 and R50-FPN backbones used for object detection are similar to those available in `Detectron2` [51], and followed the parameters settings and adjustments from [25]. Specifically, for *R50-C4*, the object detection backbone uses the output of the C4 stage, and the box prediction head uses the C5 stage with a batch norm layer following its output.

Selecting negative samples for ReSim. We use stride one average pooling of corresponding downsampling rates on feature maps of key views as negative samples. On $C4/P4$, since we use sliding window of size 48×48, where the feature map is downsampled 16x, the kernel size of average pooling is 3×3. The same rationale applies to $P3$ feature map.

While MoCo uses a queue to maintain a large number of negative image-level samples, we find that such a queue is unnecessary for region-level samples as there are a large number of negative region samples within each batch. For similarity learning on $C4/P4$, we synchronize the pooled features across GPUs, leading to $12\times12\times256 = 36,864$ negative regions; for $P3$, we use the pooled features on each individual GPU, leading to $26\times26\times256/8 = 21,632$ negative samples. Note that we do not change the temperature hyperparameter in the contrastive learning objective, as the number of negatives for region-level similarity is roughly the same as the number of negatives for image-level similarity with momentum-based queue.

A.2. Extended Experimental Results

ImageNet linear probe performance . While the object detection transfer performance leads to a substantial improvement compared to MoCo-v2, the ReSim-C4 linear probe classification accuracy from [6] drops from 67.5% to 66.1% at 200 epochs of pre-training. This drop performance decrease indicates that ImageNet classification does not necessarily indicate an improved performance for region-level transfer tasks such as object detection. This observation was similarly reported by Chen et al. [6] where the authors observed that "linear classification accuracy is not monotonically related to transfer performance in detection."

Parameter	MoCo-v2	SimSiam
batch size	256	256
num gpus	8	8
lr	0.03	0.10
schedule	cosine	cosine
opt	SGD	SGD
opt momentum	0.9	0.9
weight decay	1e-4	1e-4
epochs	200	200
projection-mlp-dims	C5: $2048 \rightarrow 128$ C4: $1024 \rightarrow 128$ C3: $512 \rightarrow 128$	C5: $2048 \rightarrow 2048 \rightarrow 2048$ C4: $1024 \rightarrow 1024 \rightarrow 1024$ C3: $512 \rightarrow 512 \rightarrow 512$
moco-k	65536	-
moco-m	0.999	-
moco-t	0.2	-
prediction-mlp-dims	-	C5: $2048 \rightarrow 512 \rightarrow 2048$ C4: $1024 \rightarrow 256 \rightarrow 1024$ C3: $512 \rightarrow 128 \rightarrow 512$

Table 7. This table provides the parameters that were used for pre-training ReSim-C4 and ReSim-FPN for the MoCo-v2 [6] and SimSiam [7] implementations carried out in this paper (unless otherwise noted). The values in the brackets indicate a change in a parameter value when pretraining the SimSiam implementation compared to the MoCo-v2 version. IN-100 experiments followed the exact same set of parameters except training occurred for 500 epochs with moco-k of 16384 on IN-100, instead of 200 epochs and moco-k of 65536 as was done on IN-1K. The SimSiam projection and prediction dimensions indicate the dimensions of the MLP as specified in [7].