Spatially Consistent Representation Learning

Byungseok Roh* Wuhyun Shin* Ildoo Kim Sungwoong Kim
Kakao Brain
{peter.roh, aiden.hsin, ildoo.kim, swkim}@kakaobrain.com

Appendix

A. Implementation details

A.1 Image Augmentations for SCRL

We use the same set of image augmentations in SimCLR [3] and BYOL [6] except random cropping. We crop the patch of the image with an area uniformly sampled between 20% and 100% of that of the original image as described in Section **??**. We observe that this change is not detrimental to BYOL and results in increasing the intersection area between SCRL's v_1 and v_2. Table A1 shows the image augmentation parameters from BYOL [6].

augmentation parameter	$\mathbb{T}_1$	$\mathbb{T}_2$
random crop probability	1.0	1.0
flip probability	0.5	0.5
color jittering probability	0.8	0.8
brightness adjustment max intensity	0.4	0.4
contrast adjustment max intensity	0.4	0.4
saturation adjustment max intensity	0.2	0.2
hue adjustment max intensity	0.1	0.1
color dropping probability	0.2	0.2
Gaussian blurring probability	1.0	0.1
solarization probability	0.0	0.2

Table A1. Parameters used to generate image augmentations [6].

A.2 PASCAL VOC Object Detection

We use Faster R-CNN [10] with ResNet-50-FPN [7, 8]. The base learning rate is set to 0.02 and multiplied by 0.1 at 12000 and 16000 steps of training, respectively. We train a model over 18000 steps with 16 batches.

A.3 COCO Object Detection

We use Faster R-CNN [10] and RetinaNet [9] with ResNet-50-FPN [7, 8]. The base learning rates are set to 0.02 for Faster R-CNN and 0.01 for RetinaNet, and multiplied by 0.1 at 60000 and 80000 steps of training, respectively. We train a model over 90000 steps with 16 batches.

In the case of training on various downstream schedules, we multiply the training schedule to the default setting, *i.e.* milestone for 1/10 learning rate decaying step is [30000, 40000], and train a model over 45000 steps for $\times$0.5 LR schedule.

A.4 COCO Instance Segmentation

We use Mask R-CNN [10] with ResNet-50-FPN [7, 8]. The base learning rate is set to 0.02 and multiplied by 0.1 at 60000 and 80000 steps of training, respectively. We train a model over 90000 steps with 16 batches.

A.5 COCO Keypoints Detection

We use Mask R-CNN [10] (keypoint version) with ResNet-50-FPN [7, 8]. The base learning rate is set to 0.02 and multiplied by 0.1 at 60000 and 80000 steps of training, respectively. We train a model over 90000 steps with 16 batches.

A.6 Cityscapes Instance Segmentation

We use Mask R-CNN [10] with ResNet-50-FPN [7, 8]. The base learning rate is set to 0.01 and multiplied by 0.1 at 18000 steps of training. We train a model over 24000 steps with 8 batches.

B. Representation Quality of FPN Evaluated under RoI Linear Protocol

We conduct the RoI linear evaluation with the ResNet-50-FPN backbone, that is pretrained on ImageNet, and verify the correlation between the RoI evaluation accuracy and the object detection AP after being fine-tuned on the downstream task. For downstream object detection, we use the Faster R-CNN method with ResNet-50-FPN on COCO dataset. To make the representation for RoI evaluation compatible with FPN architecture, we concatenate the RoI-aligned features from every stage of the feature pyramid and feed it to a linear head as usual. Note that we use mini-batch statistics for batch normalization layers during the training of the linear head in order to adapt to the statistics of COCO dataset, while simultaneously tracking running

*Equal contribution

upstream	AP on COCO downstream	RoI evaluation acc. after downstream
random	29.77	68.52
supervised-IN	38.52	79.20
MoCo-v2	37.12	77.04
SimCLR-v2	38.14	77.67
SeLa-v2	37.75	80.55
DeepCluster-v2	37.97	80.61
SwAV	39.58	81.98
BYOL	39.98	81.34
SCRL	**40.94**	**82.39**

Table A2. The correlation between RoI evaluation accuracy and AP after being fine-tuned on the downstream task.

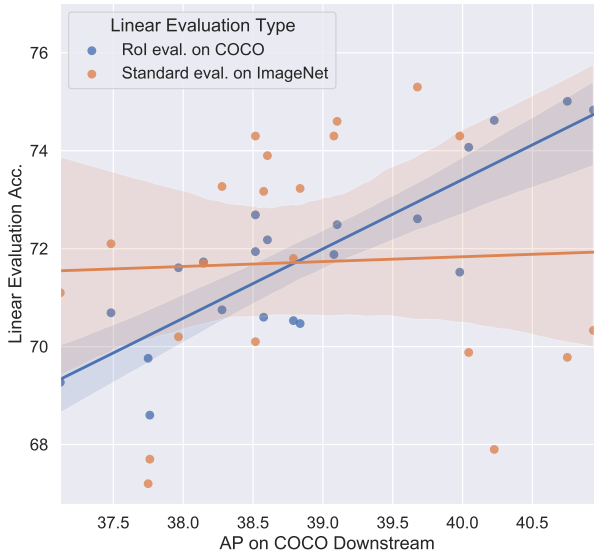

Figure A1. The correlation between two types of linear evaluation after upstream and the actual downstream performance using the initial representation. The proposed RoI evaluation (blue) shows higher positive correlation than the standard linear evaluation protocol (orange). Each point corresponds to different upstream methods with various upstream schedules. The straight line depicts linear regression result and the shaded area around it represents 95% confidence interval.

statistics that is to be used during the test. As shown in Table A2, a strong positive correlation (Pearson's coefficient is 0.97) between the two columns is observed. This justifies our assumption on our protocol with which we can measure the quality of representation without direct access to object detection downstream task.

C. The Correlation between Linear Evaluation Protocols and the Downstream Performance

In this section, we further discuss how the proposed and the standard linear evaluation protocols are correlated to the actual downstream performance with a wider range of examples. Specifically, we use SeLa-v2 [2], DeepCluster-v2 [2], SimCLR-v2 [4], MoCo-v2 [5], SwAV [2], BYOL [6], and our method, SCRL, with varied upstream epochs and ablated optional techniques such as multicrop in SwAV or box generation details in SCRL. In the case of other baselines, publicly available checkpoints provided by the authors are used. For upstream, ResNet-50 backbone is pretrained on ImageNet with different methods and, for downstream, Faster R-CNN with additional FPN is fine-tuned on COCO detection task. We apply the same treatment as described in Section B for batch normalization layers during RoI evaluation.

As shown in Figure A1, the proposed protocol, RoI evaluation shows a significantly higher correlation to the downstream performance and, in addition, Pearson correlation of our protocol (0.85) is 20× higher than the one of the standard image classification protocol on ImageNet (0.04). Based on this observation, we suggest that one can use our protocol to measure transferability to the object detection during upstream under self-supervision, in an online manner, without access to the actual downstream validation.

upstream dataset	pretrain	AP	AP_{50}	AP_{75}
COCO	BYOL	35.4	55.6	38.0
	SCRL	**39.0**	**60.1**	**42.4**

Table A3. COCO detection using Faster R-CNN, ResNet-50-FPN. Upstreams are trained with the unlabeled COCO dataset with 2000 epochs.

D. Upstream training with COCO dataset

We train the model on upstream task using unlabeled COCO `train2017` for the number of steps that correspond to 200 epochs on ImageNet. Then, we fine-tune it on COCO detection task with 1× training schedule and obtain 39.0 AP, which is 3.6 points higher than BYOL as shown in Table A3.

E. Downstream training with Sparse R-CNN

We perform COCO detection task with Sparse R-CNN [11] that does not use predefined anchors and non-maximum suppression (NMS) through the bipartite matching, similar to DETR [1]. As with other experiments we compared in the paper, SCRL outperforms the supervised ImageNet pre-trained counterpart on Sparse R-CNN. This experiment shows that our SCRL can be applicable to any other detection frameworks to boost the performance without additional training cost and efforts.

F. Additional Qualitative Analysis

Figure A2 illustrates the detected boxes with correct class prediction, where the triplet-pair of each image represents the results from the model having been initialized

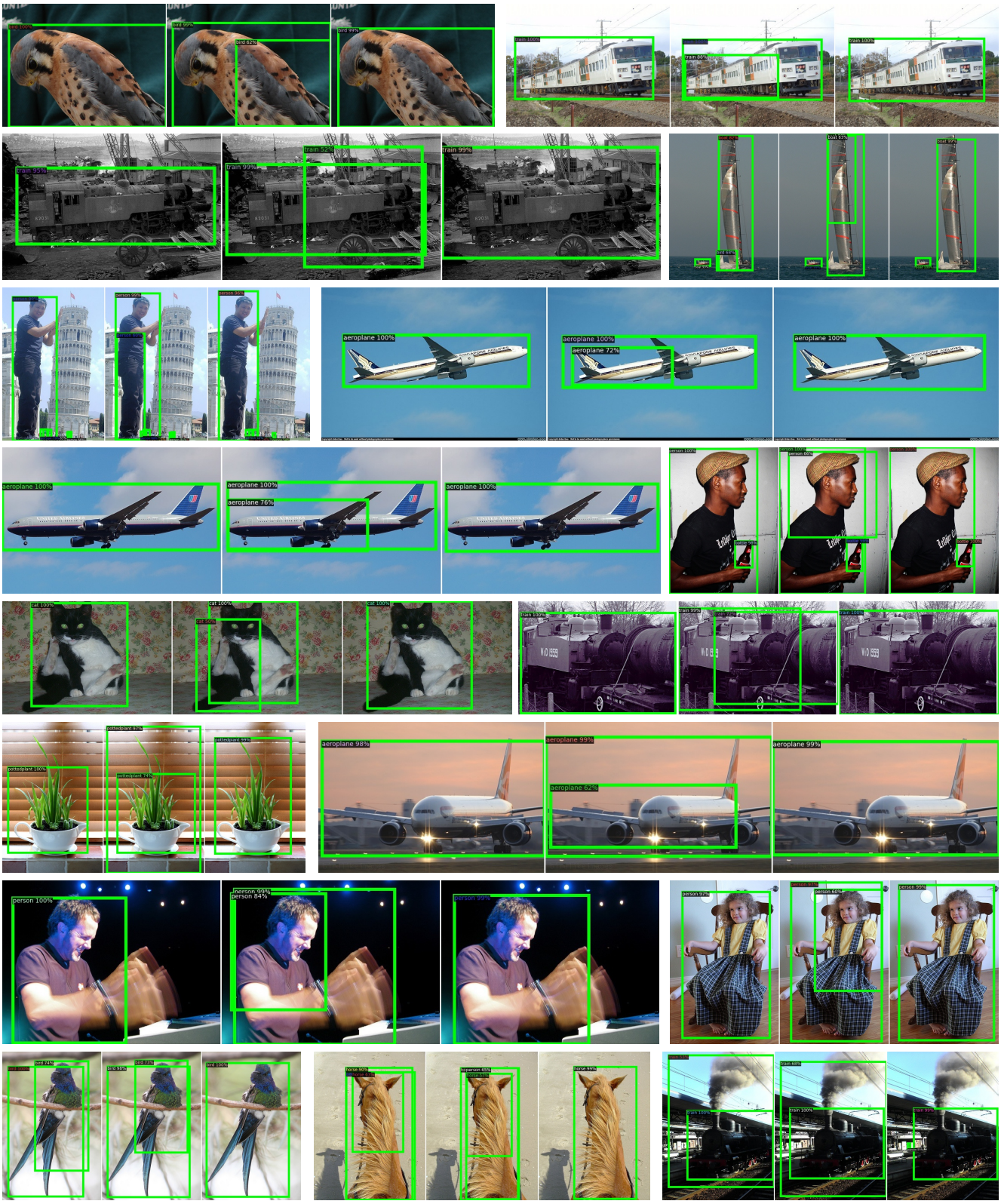

Figure A2. Qualitative comparison among ImageNet (left), BYOL (middle) and SCRL (right) on PASCAL VOC detection w/ Faster R-CNN, ResNet-50-FPN.

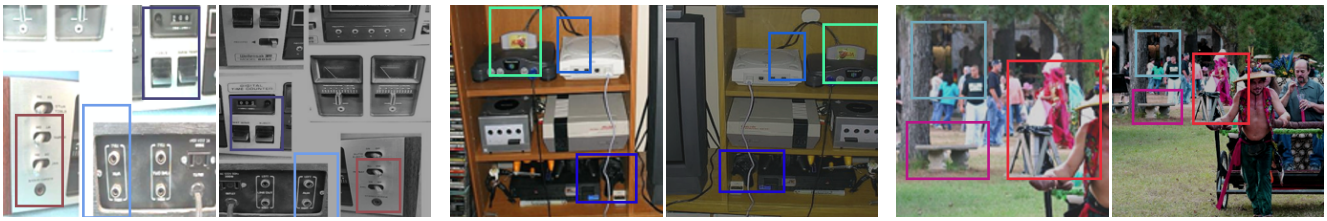

Figure A3. Randomly generated boxes from two augmented views. In two augmented views, two rectangular regions of the same color are spatially matched.

method	pretrain	AP	AP_{50}	AP_{75}
Sparse R-CNN	supervised-IN	42.3	61.2	45.7
	SCRL	**44.3**	**63.0**	**48.0**
Sparse R-CNN*	supervised-IN	44.5	63.4	48.2
	SCRL	**46.7**	**65.7**	**51.1**

Table A4. COCO detection using Sparse R-CNN [11], ResNet-50-FPN. The training schedule is 36 epochs and all downstream tasks are trained with the default hyper-parameters as in [11]. Here * indicates that the model is trained with 300 learnable proposal boxes and random crop training augmentation, similar to Deformable DETR [12].

learning rate	pretrain	AP	AP_{50}	AP_{75}
0.45 (default)	SCRL	40.9	**62.5**	44.5
0.3	SCRL	**41.2**	62.4	**45.1**

Table A5. Performance improvement in COCO detection task with ResNet-50 trained for 1000 epochs, when using coarsely-tuned learning rate of 0.3.

with ImageNet pretraining, BYOL, and SCRL, respectively. We found BYOL tends to detect a part of the object simultaneously as well as the entire object. As we described in the paper, we conjecture that these unintended consequences of BYOL are caused by semantically inconsistent matching between randomly cropped views by aggressive augmentation. Though BYOL outperforms ImageNet pretrained representation on the entire test set, the shortcoming observed in this qualitative analysis implies that there still exists room for further improvement, which is exactly where SCRL tries to fill by introducing the spatial consistency. Thereby, SCRL detects the entire object solidly since it produces position and scale-invariant features. Interestingly, the bottom row in Figure A2 shows that SCRL is robust to such phenomena even though when ImageNet pretraining generate multiple boxes in a single object.

G. Random Boxes from Two Augmented Views

Figure A3 shows randomly generated boxes from two augmented views during the upstream training. We use $K = 3$ which is the total number of generated boxes in an image for simplicity while the main experiment generates 10 boxes (*i.e.* $K = 10$) as a default training setting.

H. Performance Improvement by Hyperparameter Search

In all experiments in the paper, we naively transfer the sharable hyperparameters of BYOL to SCRL with which one can reproduce the performance reported in [6]. Although SCRL already outperforms BYOL under this condition, we observe an additional gain in downstream performance by tuning the learning rate alone with simple grid search, *i.e.* +0.3 AP increase with the learning rate of 0.3 on COCO detection task, compared to the default learning rate, 0.45, as shown in Table A5.

I. Using Negative Pairs for Upstream Training

We exploit the negative pairs based on the SimCLR framework and obtain somewhat better results (+0.47 AP on COCO detection). However, this performance improvement requires an increased batch size and a sophisticated composition of the negative pairs, therefore we leave it for future works.

J. Scale-invariant Representation Learning

Before coming up with the feature-level matching, we had started from the baselines enforcing the input-level consistency. The best model share same details with BYOL but with the modification in the augmentation: spatially consistent cropping(or just use the entire image), and random aspect resizing followed by mean-padding to ensure the same spatial dimension for the sake of parallelized computation. We observe the performance degradation on the localization downstream task, in comparison to SCRL even with a single RoI pair. We hypothesize that this is due to the limitation in obtaining consistent local representation against an internal geometric translation of an object.

References

[1] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In Andrea Vedaldi, Horst Bischof, Thomas Brox, and Jan-Michael Frahm, editors, *Computer Vision - ECCV 2020 - 16th European Conference, Glasgow, UK, August 23-28, 2020, Proceedings, Part I*, volume 12346 of *Lecture Notes in Computer Science*, pages 213–229. Springer, 2020. 2

[2] Mathilde Caron, Ishan Misra, Julien Mairal, Priya Goyal, Piotr Bojanowski, and Armand Joulin. Unsupervised learning of visual features by contrasting cluster assignments. *CoRR*, abs/2006.09882, 2020. 2

[3] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey E. Hinton. A simple framework for contrastive learning of visual representations. *CoRR*, abs/2002.05709, 2020. 1

[4] Ting Chen, Simon Kornblith, Kevin Swersky, Mohammad Norouzi, and Geoffrey E. Hinton. Big self-supervised models are strong semi-supervised learners. *CoRR*, abs/2006.10029, 2020. 2

[5] Xinlei Chen, Haoqi Fan, Ross B. Girshick, and Kaiming He. Improved baselines with momentum contrastive learning. *CoRR*, abs/2003.04297, 2020. 2

[6] Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre H. Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Ávila Pires, Zhaohan Daniel Guo, Mohammad Gheshlaghi Azar, Bilal Piot, Koray Kavukcuoglu, Rémi Munos, and Michal Valko. Bootstrap your own latent: A new approach to self-supervised learning. *CoRR*, abs/2006.07733, 2020. 1, 2, 4

[7] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, pages 770–778. IEEE Computer Society, 2016. 1

[8] Tsung-Yi Lin, Piotr Dollár, Ross B. Girshick, Kaiming He, Bharath Hariharan, and Serge J. Belongie. Feature pyramid networks for object detection. In *CVPR*, pages 936–944. IEEE Computer Society, 2017. 1

[9] Tsung-Yi Lin, Priya Goyal, Ross B. Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. In *ICCV*, pages 2999–3007. IEEE Computer Society, 2017. 1

[10] Shaoqing Ren, Kaiming He, Ross B. Girshick, and Jian Sun. Faster R-CNN: towards real-time object detection with region proposal networks. In *NIPS*, pages 91–99, 2015. 1

[11] Peize Sun, Rufeng Zhang, Yi Jiang, Tao Kong, Chenfeng Xu, Wei Zhan, Masayoshi Tomizuka, Lei Li, Zehuan Yuan, Changhu Wang, and Ping Luo. SparseR-CNN: End-to-end object detection with learnable proposals. *arXiv preprint arXiv:2011.12450*, 2020. 2, 4

[12] Xizhou Zhu, Weijie Su, Lewei Lu, Bin Li, Xiaogang Wang, and Jifeng Dai. Deformable DETR: deformable transformers for end-to-end object detection. *CoRR*, abs/2010.04159, 2020. 4