Leverage Your Local and Global Representations: A New Self-Supervised Learning Strategy Supplementary Material

Tong Zhang[1] Congpei Qiu[2] Wei Ke[2] Sabine Süsstrunk[1] Mathieu Salzmann[1]
[1] School of Computer and Communication Sciences, EPFL, Switzerland
[2] Xi'an Jiaotong University, China

1. Datasets and hyper-parameters

CIFAR10 contains 50000 training images and 10000 test images from 10 different classes, with an image size of 32×32. Due to the image size and dataset scale, we use a ResNet-18 [10] as backbone, and we set r_l to 0.6 and r_g to 0.4, which means the local and global views will be randomly cropped from $[0.08, 0.6]$ and $[0.4, 1]$, respectively, followed by a resizing to 32.

STL10 is a subset of ImageNet-1k (IN-1k) [18], that contains 10 classes depicted in images resized to 96×96. It has 100000 unlabeled images, 5000 training images (500 per class) and 8000 test images (800 per class). Following [20], we use the small AlexNet [13] as backbone and use 105000 images as our self-supervised training set. r_l and r_g are set to 0.4 and 0.4.

IN-100 is also a subset of IN-1k, but retaining the image size of IN-1k. It contains 100 classes, naturally corresponding to $1/10$ of the IN-1K images (124k) [19]. We use a ResNet-34 to extract the feature representation. We set the same r_l and r_g as for STL10, and resize our local crops to 96×96 to reduce the computation and GPU memory.

We summarize all the datasets information used in our image recognition under transfer learning setting experiments in Table 1.

2. Ablation for multi-crop strategy

In order to justify the superiority of our strategy on utilizing the multi-crop, we compare our results with the multi-crop learning strategy provided by SwAV [3]:

$$\mathcal{L} = -\frac{1}{N}\frac{1}{M-1}\sum_{i=1}^{N}\sum_{\mathbf{v}^+\in\{\mathbf{v}_i^+\}}\ell(\mathbf{z}_i,\mathbf{v}^+)$$

$$\ell(\mathbf{z}_i,\mathbf{v}^+) = \log\frac{\exp(\mathbf{z}_i^T\mathbf{v}^+/\tau)}{\exp(\mathbf{z}_i^T\mathbf{v}^+/\tau)+\sum_{\mathbf{v}^-\in\{\mathbf{v}_i^-\}}\exp(\mathbf{z}_i^T\mathbf{v}^-/\tau)} \tag{1}$$

For each crop representation $\mathbf{z_i}$, the set $\{\mathbf{v}^+\}$ is the corresponding set of positive crops, and M is the number of crops per instance. Since they need to compute the InfoNCE loss for each positive pair, their multiple crops strategy needs more computation than ours. Note that we follow the ablation study in supplementary material [4] to pick the crop ratio for local and global, where they provide the best ratios are $[0.05, 0.3]$ and $[0.3, 1.0]$.

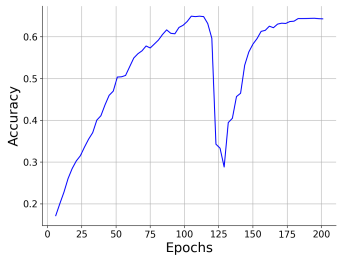

Figure 1. The KNN monitor of applying SwAV-style multi-crop to SimSiam.

As shown as Table 2 and Table 3, applying SwAV-style multi-crop to MoCo, MoCo (4-crop), has slight improvement, however, SimSiam (4-crop) degenerates a lot due to pulling the global to local undermines the optimization, where the KNN results during the optimization can be seen as Figure 1. The same observation of degenerating also can be found in [4], where they apply multi-crop to BYOL [8]. As BYOL and SimSiam do not have negative pairs and memory bank, the local crop will easily mislead the distribution of global crops. In contrast, our strategy without the local-to-local dissimilarity on both MoCo and SimSiam, namely only using multi-crop part of our strategy, achieves significantly better results than the SwAV-sytle multi-crop. It manifests that more computation within the positive pairs do not necessarily benefits the performance.

Meanwhile, we also provide the results of running MoCo [9] and SimSiam [5] for 400 epochs, which is twice our counterparts. Although MoCo and SimSiam employ the same amount of image crops as our model when training for 400 epochs, their parameters are updated twice as often as ours. Despite the unfair comparison, our models outperform

Dataset	No. of training	No. of validation	No. of test	No. of class	Metric
Food [2]	68175	7575	25250	101	Top-1 Acc
MIT67 [17]	4690	670	1340	67	Top-1 Acc
Pets [16]	2940	740	3669	37	MPC Acc
Flowers [15]	1020	1020	6149	102	MPC Acc
Caltech101 [7]	2525	505	5647	101	MPC Acc
Cars [12]	6494	1650	8041	196	Top-1 Acc
Aircraft [14]	3334	3333	3333	100	MPC Acc
DTD(split 1) [6]	1880	1880	1880	47	Top-1 Acc

Table 1. The information of all datasets used in image recognition on transfer learning. The MPC denotes the Mean Per-Class.

	KNN (acc %)	Linear (acc %)
MoCo	64.18	68.48
MoCo (400 epoch)	71.84	75.44
MoCo (4-crop)	65.4	69.78
MoCo-LoGo w/o L2L	69	73.24
MoCo-LoGo	**76.82**	**79.32**

Table 2. Test results of MoCo and MoCo based frameworks trained on ImageNet-100 with a ResNet-34 backbone. We show the top-1 accuracy for a KNN and a linear classifier. Default training epoch is 200

	KNN (acc %)	Linear (acc %)
SimSiam	71.72	75.48
SimSiam (400 epoch)	73.16	76.78
SimSiam (4-crop)	64.32	68.46
SimSiam-LoGo w/o L2L	76.64	79.52
SimSiam-LoGo	**78.48**	**80.94**

Table 3. Test results of SimSiam and SimSiam based frameworks trained on ImageNet-100 with a ResNet-34 backbone. We show the top-1 accuracy for a KNN and a linear classifier. Default training epoch is 200.

	KNN (acc %)	Linear (acc %)
MoCo	66.02	74.84
MoCo(400 epoch)	78.08	82.06
SwAV(4-crop)	69.4	80.6
MoCo+LoGo	**81.32**	**85.14**

Table 4. Training on ImageNet-100 with a ResNet-50 backbone. We show the top-1 accuracy for a KNN and a linear classifier.

them by a wide margin.

3. Transfer Learning

Since the benefits of applying SwAV-style multi-crop to MoCo and SimSiam are quite minor, we did not add them into the comparison in transfer learning. We add SwAV with 4 crops and MoCo trained for 400 epoch into the ablation baselines. As shown in Table 4, MoCo increases around 12 percent after training 200 epochs more, however, it is still lower than ours with 200 epochs training. At the same time, the accuracy of SwAV with the same number of crops is lower than ours (both with and without local-to-local dissimilarity) by a large margin.

We use the L-BFGS to minimize the cross-entropy loss with ℓ_2 regularization [5, 8] for the linear classification, where 5000 iterations are applied. Table 5 summarizes the results of linear classification when transferring to other datasets where the backbone is resnet50 trained on ImageNet-100, our strategy improves both MoCo and SimSiam by a large margin compared to the existing strategy. At the same time we also include the results of pre-training on IN-100 and testing on IN-1k in Table 6.

Besides, as shown in Table 7, we provide the results of our method used to pre-train a backbone on MSCOCO followed by applying a linear SVM on the resulting VOC representations. Note that our full model (last column) still has the best performance.

4. Crop Ratio

We provide results of using different crop ratios for local and global crops. Due to the computation limitation, we conduct experiments on STL datasets to show how size affects our model's performance. We firstly study our robustness when local crop size changes. Table 8 shows that both SimSiam-LoGo and MoCo-LoGo are robust to the local size while keeping global size constant as $[0.4, 1]$. When the local crop size is higher than 0.4, the results are quite stable. By contrast, we fix the range of local crop ratio as $[0.08, 1]$ and change the global size τ_g from 0.3 to 0.7 as Table 8 shown, which is equivalent to swapping the order of local and global crops in our strategy. It clearly illustrates that the lower the global ratio is, the worse the results we can have, especially for SimSiam-LoGo. The experiments support our claim that pulling the global crops to local crop will confuse the optimization. Note that both Table 8 and

	CIFAR10	CIFAR100	Food	MIT67	Pets	Flowers	Caltech	Cars	Aircraft	DTD
MoCo(200 epoch)	83.71	60.59	58.21	57.54	64.30	85.56	74.12	32.63	46.23	60.64
MoCo(400 epoch)	84.60	61.60	59.37	61.64	70.08	82.43	77.25	33.86	41.21	64.47
MoCo-LoGo	**86.09**	**63.43**	**65.67**	**67.54**	**76.17**	**92.13**	**82.09**	**40.77**	50.07	**67.87**
SwAV(multi-crop)	83.4	59.31	59.3	63.36	66.45	88.73	78.33	36.26	52.36	66.06
SimSiam	**86.8**	**65.61**	61.40	63.36	72.12	91.43	83.44	44.83	54.92	65.11
SimSiam-LoGo	86.64	65.55	**65.31**	**68.28**	**75.78**	**93.02**	**84.01**	**47.79**	**58.44**	**68.94**

Table 5. Image recognition under transfer learning setting for different self-training methods. We highlight the best results in **bold**.

	MoCo	MoCo+LoGo w/o L2L	MoCo+LoGo
Linear	47.49	54.49	**58.03**

Table 6. Results of linear evaluation of ResNet-50 pretrained on IN-100 and test on IN-1000

	MoCo	MoCo+LoGo w/o L2L	MoCo+LoGo
MAP	76.12	77.97	**79.59**

Table 7. Results of linear evaluation of ResNet-50 pretrained on MSCOCO

Table 9 show that the MoCo-LoGo has better robustness to the crop size due to its memory bank mechanism, which provide a large number of negative pairs.

5. Connection with MINE

Mutual Information Neural Estimator (MINE) [1] estimates the MI of an image $\mathbf{x}$ and its latent vector $\mathbf{z}$ by optimizing a function f_{θ_d} parametrized by θ:

$$I_\Theta(X,Z) = \sup_\theta \mathbb{E}_{\mathbb{P}_{XZ}}[f_{\theta_d}(\mathbf{x},\mathbf{z})] - \log\left(\mathbb{E}_{\mathbb{P}_X\otimes\mathbb{P}_Z}\left[e^{f_{\theta_d}((\mathbf{x},\mathbf{z}))}\right]\right). \quad (2)$$

Where function $f_{\theta_d} : \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}^+$. However, estimating the mutual information between two local crops is not our purpose. We are looking for estimating the dataset dependent similarity of two crops. Thus, we relax the cost function by using the property of concave function, $\log\left(\mathbb{E}_{\mathbb{P}_X\otimes\mathbb{P}_Z}\left[e^{f_{\theta_d}}\right]\right) < \mathbb{E}_{\mathbb{P}_X\otimes\mathbb{P}_Z}[f_{\theta_d}]$, to have:

$$I_\Theta(X,Z) \geq \sup_\theta \mathbb{E}_{\mathbb{P}_{XZ}}[f_{\theta_d}((\mathbf{x},\mathbf{z}))] - \mathbb{E}_{\mathbb{P}_X\otimes\mathbb{P}_Z}[f_{\theta_d}((\mathbf{x},\mathbf{z}))]. \quad (3)$$

At the same time, the second term in MINE has a complexity of $O(n2)$, which is highly computationally and memory intensive. As a result, we use random sampling to substitute the expectation estimator, and our similarity estimator f_{θ_d} is derived by maximizing the following cost function:

$$\theta_d^* = \underset{\theta_d}{\operatorname{argmax}}\, \mathbb{E}_{\mathbb{P}_{\mathbf{z}^l}}[f_{\theta_d}(\mathbf{z}_1^l, \mathbf{z}_2^l) - f_{\theta_d}(\mathbf{z}_1^l, \mathbf{z}^{l-})]. \quad (4)$$

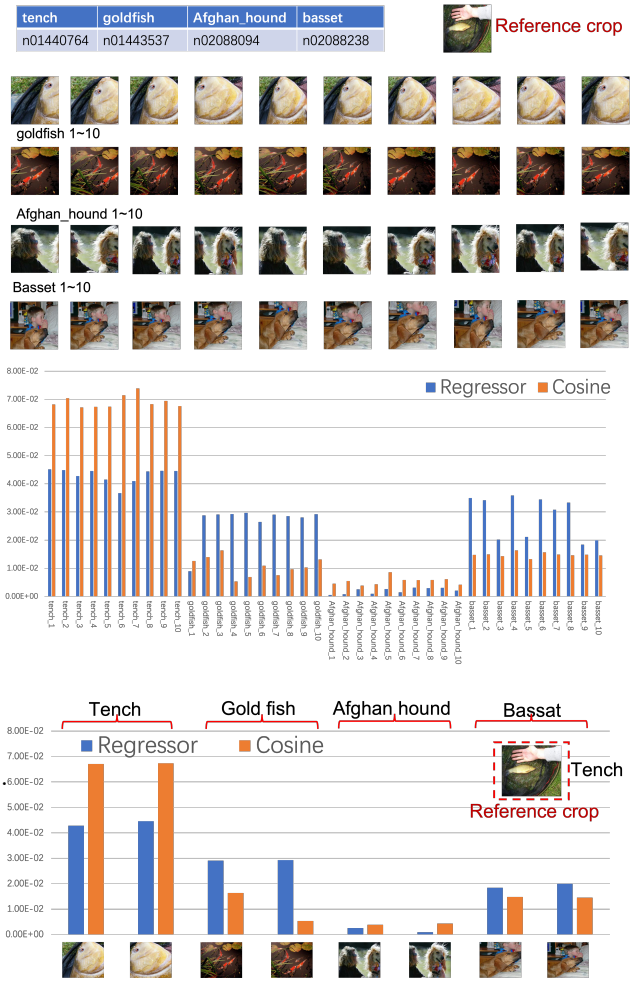

Figure 2. The top figure shows the ImageNet ID of 4 different classes and their random crops. The middle figure displays the normalized similarity between reference crop and other 40 different crops pair-wisely. The bottom image is the selected two crops for each class different classes

τ_l	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1.0
MoCo-LoGo	76.33	76.79	76.02	77.15	76.91	77.36	77.28	77.76
SimSiam-LoGo	75.68	76.96	76.53	76.1	76.43	76.88	76.16	76.54

Table 8. We keep the global crop ratio τ_g as 0.4, which means our global crop is randomly pick between $[0.4, 1]$, and change the ratio of local crop from $[0.08, \tau_l]$. The number are top-1 KNN accuracy on STL10 datasets.

τ_g	0.3	0.4	0.5	0.6	0.7
MoCo-LoGo	75.03	76.38	76.67	77.36	77.21
SimSiam-LoGo	72.43	73.38	75.56	75.75	75.84

Table 9. The results of changing global crop ratio range as $[0.08, \tau_g]$, while keeping the local crop ratio to be the same, namely $[0.08, 1]$. The number are top-1 KNN accuracy STL10 datasets.

where $\mathbf{z}^{l-}$ is a local crop representation from a different image and it can be randomly sampled in the same batch. Interestingly and intuitively, the loose and fast version of MINE meets our assumption that the similarity of positive crops pairs is greater than the similarity of negative crops pairs. Note that, we use the optimal $f_{\theta_d^*}$ as our similarity regressor instead of the supremum of equation 4, which is used in MINE. Through training with the encoder f_{θ_e}, the estimator f_{θ_d} will also adjust its similarity value based on the distribution of the feature space.

More results of our regressor can be seen in Figure 2, where we give a specific example of how we generate the regressor output in the main paper. Since the reference crop has a human hand, our regressor gives higher similarity to the Basset than Cosine distance, where a human (hand) is next to the dog. At the same time, the similarity with the Bassat is lower than with the Tench (correct class) and Gold fish (Semantically closer). The observation is consistent with the main paper. Due to the cost function we designed, the regressor avoids the high-confidence issue [11].

References

[1] Mohamed Ishmael Belghazi, Aristide Baratin, Sai Rajeswar, Sherjil Ozair, Yoshua Bengio, Aaron Courville, and R Devon Hjelm. Mine: mutual information neural estimation. *arXiv preprint arXiv:1801.04062*, 2018. 3

[2] Lukas Bossard, Matthieu Guillaumin, and Luc Van Gool. Food-101–mining discriminative components with random forests. In *European conference on computer vision*, pages 446–461. Springer, 2014. 2

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

[4] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. *arXiv preprint arXiv:2104.14294*, 2021. 1

[5] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pages 1597–1607. PMLR, 2020. 1, 2

[6] Mircea Cimpoi, Subhransu Maji, Iasonas Kokkinos, Sammy Mohamed, and Andrea Vedaldi. Describing textures in the wild. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3606–3613, 2014. 2

[7] Li Fei-Fei, Rob Fergus, and Pietro Perona. Learning generative visual models from few training examples: An incremental bayesian approach tested on 101 object categories. In *2004 conference on computer vision and pattern recognition workshop*, pages 178–178. IEEE, 2004. 2

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

[9] 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*, pages 9729–9738, 2020. 1

[10] 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*, pages 770–778, 2016. 1

[11] Matthias Hein, Maksym Andriushchenko, and Julian Bitterwolf. Why relu networks yield high-confidence predictions far away from the training data and how to mitigate the problem. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 41–50, 2019. 4

[12] Jonathan Krause, Michael Stark, Jia Deng, and Li Fei-Fei. 3d object representations for fine-grained categorization. In *Proceedings of the IEEE international conference on computer vision workshops*, pages 554–561, 2013. 2

[13] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. *Advances in neural information processing systems*, 25:1097–1105, 2012. 1

[14] Subhransu Maji, Esa Rahtu, Juho Kannala, Matthew Blaschko, and Andrea Vedaldi. Fine-grained visual classification of aircraft. *arXiv preprint arXiv:1306.5151*, 2013. 2

[15] Maria-Elena Nilsback and Andrew Zisserman. Automated flower classification over a large number of classes. In *2008 Sixth Indian Conference on Computer Vision, Graphics & Image Processing*, pages 722–729. IEEE, 2008. 2

[16] Omkar M Parkhi, Andrea Vedaldi, Andrew Zisserman, and CV Jawahar. Cats and dogs. In *2012 IEEE conference on computer vision and pattern recognition*, pages 3498–3505. IEEE, 2012. 2

[17] Ariadna Quattoni and Antonio Torralba. Recognizing indoor scenes. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 413–420. IEEE, 2009. 2

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

[19] Yonglong Tian, Dilip Krishnan, and Phillip Isola. Contrastive multiview coding. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XI 16*, pages 776–794. Springer, 2020. 1

[20] Tongzhou Wang and Phillip Isola. Understanding contrastive representation learning through alignment and uniformity on the hypersphere. In *International Conference on Machine Learning*, pages 9929–9939. PMLR, 2020. 1