Improved Baselines with Momentum Contrastive Learning

Xinlei Chen Haoqi Fan Ross Girshick Kaiming He
Facebook AI Research (FAIR)

Abstract

Contrastive unsupervised learning has recently shown encouraging progress, e.g.*, in Momentum Contrast (MoCo) and SimCLR. In this note, we verify the effectiveness of two of SimCLR's design improvements by implementing them in the MoCo framework. With simple modifications to MoCo—namely, using an MLP projection head and more data augmentation—we establish stronger baselines that outperform SimCLR and do not require large training batches. We hope this will make state-of-the-art unsupervised learning research more accessible. Code will be made public.*

1. Introduction

Recent studies on unsupervised representation learning from images [16, 13, 8, 17, 1, 9, 15, 6, 12, 2] are converging on a central concept known as *contrastive learning* [5]. The results are promising: *e.g.*, Momentum Contrast (MoCo) [6] shows that unsupervised pre-training can surpass its ImageNet-supervised counterpart in multiple detection and segmentation tasks, and SimCLR [2] further reduces the gap in linear classifier performance between unsupervised and supervised pre-training representations.

This note establishes stronger and more feasible baselines built in the MoCo framework. We report that two design improvements used in SimCLR, namely, an MLP projection head and stronger data augmentation, are orthogonal to the frameworks of MoCo and SimCLR, and when used with MoCo they lead to better image classification and object detection transfer learning results. Moreover, the MoCo framework can process a large set of negative samples without requiring large training batches (Fig. 1). In contrast to SimCLR's large 4k~8k batches, which require TPU support, our "**MoCo v2**" baselines can run on a typical 8-GPU machine and achieve better results than SimCLR. We hope these improved baselines will provide a reference for future research in unsupervised learning.

2. Background

Contrastive learning. Contrastive learning [5] is a framework that learns similar/dissimilar representations from data that are organized into similar/dissimilar pairs. This can be formulated as a dictionary look-up problem. An effective contrastive loss function, called InfoNCE [13], is:

$$\mathcal{L}_{q,k^+,\{k^-\}} = -\log \frac{\exp(q\cdot k^+/\tau)}{\exp(q\cdot k^+/\tau) + \sum_{k^-} \exp(q\cdot k^-/\tau)}. \quad (1)$$

Here q is a query representation, k^+ is a representation of the positive (similar) key sample, and $\{k^-\}$ are representations of the negative (dissimilar) key samples. τ is a temperature hyper-parameter. In the *instance discrimination* pretext task [16] (used by MoCo and SimCLR), a query and a key form a positive pair if they are data-augmented versions of the same image, and otherwise form a negative pair.

The contrastive loss (1) can be minimized by various mechanisms that differ in how the keys are maintained [6]. In an end-to-end mechanism (Fig. 1a) [13, 8, 17, 1, 9, 2], the negative keys are from the same batch and updated end-to-end by back-propagation. SimCLR [2] is based on this mechanism and requires a large batch to provide a large set of negatives. In the MoCo mechanism (Fig. 1b) [6], the negative keys are maintained in a queue, and only the queries and positive keys are encoded in each training batch. A momentum encoder is adopted to improve the representation consistency between the current and earlier keys. MoCo decouples the batch size from the number of negatives.

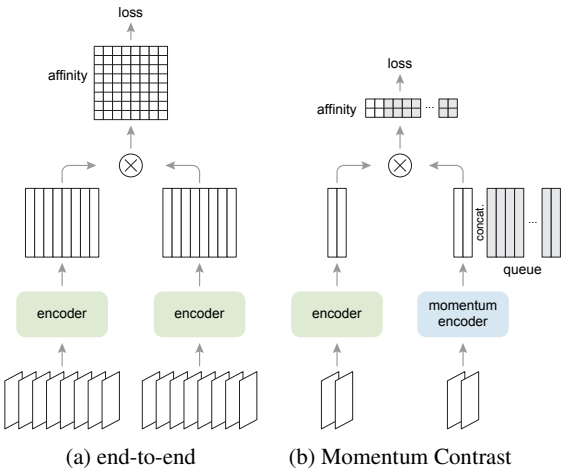

Figure 1. A **batching** perspective of two optimization mechanisms for contrastive learning. Images are encoded into a representation space, in which pairwise affinities are computed.

case	unsup. pre-train MLP	aug+	cos	epochs	ImageNet acc.	VOC detection AP_{50}	AP	AP_{75}
supervised					76.5	81.3	53.5	58.8
MoCo v1				200	60.6	81.5	55.9	62.6
(a)	✓			200	66.2	82.0	56.4	62.6
(b)		✓		200	63.4	82.2	56.8	63.2
(c)	✓	✓		200	67.3	**82.5**	57.2	63.9
(d)	✓	✓	✓	200	67.5	82.4	57.0	63.6
(e)	✓	✓	✓	**800**	**71.1**	**82.5**	**57.4**	**64.0**

Table 1. **Ablation of MoCo baselines**, evaluated by ResNet-50 for (i) ImageNet linear classification, and (ii) fine-tuning VOC object detection (mean of 5 trials). "**MLP**": with an MLP head; "**aug+**": with extra blur augmentation; "**cos**": cosine learning rate schedule.

Improved designs. SimCLR [2] improves the end-to-end variant of instance discrimination in three aspects: (i) a substantially larger batch (4k or 8k) that can provide more negative samples; (ii) replacing the output *fc* projection head [16] with an MLP head; (iii) stronger data augmentation.

In the MoCo framework, a large number of negative samples are readily available; the MLP head and data augmentation are orthogonal to how contrastive learning is instantiated. Next we study these improvements in MoCo.

3. Experiments

Settings. Unsupervised learning is conducted on the 1.28M ImageNet [3] training set. We follow two common protocols for evaluation. (i) *ImageNet linear classification*: features are frozen and a supervised linear classifier is trained; we report 1-crop (224×224), top-1 validation accuracy. (ii) *Transferring to VOC object detection* [4]: a Faster R-CNN detector [14] (C4-backbone) is fine-tuned end-to-end on the VOC 07+12 `trainval` set[1] and evaluated on the VOC 07 `test` set using the COCO suite of metrics [10]. We use the same hyper-parameters (except when noted) and codebase as MoCo [6]. All results use a standard-size ResNet-50 [7].

MLP head. Following [2], we replace the *fc* head in MoCo with a 2-layer MLP head (hidden layer 2048-d, with ReLU). Note this only influences the unsupervised training stage; the *linear* classification or transferring stage does not use this MLP head. Also, following [2], we search for an optimal τ w.r.t. ImageNet linear classification accuracy:

τ	0.07	0.1	0.2	0.3	0.4	0.5
w/o MLP	60.6	**60.7**	59.0	58.2	57.2	56.4
w/ MLP	62.9	64.9	**66.2**	65.7	65.0	64.3

Using the default $\tau = 0.07$ [16, 6], pre-training with the MLP head improves from 60.6% to 62.9%; switching to the optimal value for MLP (0.2), the accuracy increases to 66.2%. Table 1(a) shows its detection results: in contrast to the big leap on ImageNet, the detection gains are smaller.

Augmentation. We extend the original augmentation in [6] by including the blur augmentation in [2] (we find the stronger color distortion in [2] has diminishing gains in our higher baselines). The extra augmentation alone (*i.e.*, no MLP) improves the MoCo baseline on ImageNet by 2.8% to 63.4%, Table 1(b). Interestingly, its detection accuracy is higher than that of using the MLP alone, Table 1(b) *vs.* (a), despite much lower linear classification accuracy (63.4% *vs.* 66.2%). This indicates that *linear classification accuracy is not monotonically related to transfer performance in detection*. With the MLP, the extra augmentation boosts ImageNet accuracy to 67.3%, Table 1(c).

[1]For all entries (including the supervised and MoCo v1 baselines), we fine-tune for 24k iterations on VOC, up from 18k in [6].

case	unsup. pre-train MLP	aug+	cos	epochs	batch	ImageNet acc.
MoCo v1 [6]				200	256	60.6
SimCLR [2]	✓	✓	✓	200	256	61.9
SimCLR [2]	✓	✓	✓	200	8192	66.6
MoCo v2	✓	✓	✓	200	256	**67.5**
*results of **longer** unsupervised training follow:*						
SimCLR [2]	✓	✓	✓	1000	4096	69.3
MoCo v2	✓	✓	✓	800	256	**71.1**

Table 2. **MoCo *vs.* SimCLR**: ImageNet linear classifier accuracy (**ResNet-50, 1-crop 224×224**), trained on features from unsupervised pre-training. "aug+" in SimCLR includes blur and stronger color distortion. SimCLR ablations are from Fig. 9 in [2] (we thank the authors for providing the numerical results).

mechanism	batch	memory / GPU	time / 200-ep.
MoCo	256	**5.0G**	**53 hrs**
end-to-end	256	7.4G	65 hrs
end-to-end	4096	93.0G†	n/a

Table 3. **Memory and time cost** in 8 V100 16G GPUs, implemented in PyTorch. †: based on our estimation.

Comparison with SimCLR. Table 2 compares SimCLR [2] with our results, referred to as MoCo v2. For fair comparisons, we also study a cosine (half-period) learning rate schedule [11] which SimCLR adopts. See Table 1(d, e). Using pre-training with 200 epochs and a batch size of 256, MoCo v2 achieves 67.5% accuracy on ImageNet: this is 5.6% higher than SimCLR *under the same epochs and batch size*, and better than SimCLR's large-batch result 66.6%. With 800-epoch pre-training, MoCo v2 achieves 71.1%, outperforming SimCLR's 69.3% with 1000 epochs.

Computational cost. In Table 3 we report the memory and time cost of our implementation. The end-to-end case reflects the SimCLR cost in GPUs (instead of TPUs in [2]). The 4k batch size is intractable even in a high-end 8-GPU machine. Also, under the same batch size of 256, the end-to-end variant is still more costly in memory and time, because it back-propagates to both q and k encoders, while MoCo back-propagates to the q encoder only.

Table 2 and 3 suggest that large batches are not necessary for good accuracy, and state-of-the-art results can be made more accessible. The improvements we investigate require only a few lines of code changes to MoCo v1, and we will make the code public to facilitate future research.

References

[1] Philip Bachman, R Devon Hjelm, and William Buchwalter. Learning representations by maximizing mutual information across views. *arXiv:1906.00910*, 2019.

[2] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. *arXiv:2002.05709*, 2020.

[3] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. ImageNet: A large-scale hierarchical image database. In *CVPR*, 2009.

[4] Mark Everingham, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The PASCAL Visual Object Classes (VOC) Challenge. *IJCV*, 2010.

[5] Raia Hadsell, Sumit Chopra, and Yann LeCun. Dimensionality reduction by learning an invariant mapping. In *CVPR*, 2006.

[6] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. *arXiv:1911.05722*, 2019.

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

[8] R Devon Hjelm, Alex Fedorov, Samuel Lavoie-Marchildon, Karan Grewal, Adam Trischler, and Yoshua Bengio. Learning deep representations by mutual information estimation and maximization. In *ICLR*, 2019.

[9] Olivier J. Hnaff, Aravind Srinivas, Jeffrey De Fauw, Ali Razavi, Carl Doersch, S. M. Ali Eslami, and Aaron van den Oord. Data-efficient image recognition with contrastive predictive coding. *arXiv:1905.09272v2*, 2019.

[10] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft COCO: Common objects in context. In *ECCV*. 2014.

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

[12] Ishan Misra and Laurens van der Maaten. Self-supervised learning of pretext-invariant representations. *arXiv:1912.01991*, 2019.

[13] Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv:1807.03748*, 2018.

[14] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster R-CNN: Towards real-time object detection with region proposal networks. In *NeurIPS*, 2015.

[15] Yonglong Tian, Dilip Krishnan, and Phillip Isola. Contrastive multiview coding. *arXiv:1906.05849*, 2019.

[16] Zhirong Wu, Yuanjun Xiong, Stella Yu, and Dahua Lin. Unsupervised feature learning via non-parametric instance discrimination. In *CVPR*, 2018.

[17] Mang Ye, Xu Zhang, Pong C Yuen, and Shih-Fu Chang. Unsupervised embedding learning via invariant and spreading instance feature. In *CVPR*, 2019.