Contrastive Self-supervised Learning via Minimizing Distance between Cosine Similarities of Negative Pair

Anonymous Author(s)
Affiliation
Address
email

Abstract

Contrastive self-supervised learning in image classification is a method that trains one image to be randomly augmented with two images(positive pair) so that they get closer to each other in the latent space or away from other images(negative pair) in the same set. Existing methods using negative pairs have a problem that images of the same class in the same batch are incorrectly classified as negative pairs. To prevent these negative pairs from being classified incorrectly, we make the cosine similarities between the negative pairs similar rather than maximizing the distance between them. When the proposed method was trained on the unlabeled ImageNet dataset and then compared with the existing methods, the best accuracy was achieved in the linear evaluation and transfer learning. Surprisingly, we also achieved meaningful results in experiments trained using only negative pairs.

1 Introduction

Processing the immense data without annotation on the Internet or social networking service takes a lot of time and cost to utilize them in supervised learning. Self-supervised learning aims to obtain good features and transfer them to downstream tasks without relying on human annotations and is currently performing very close to supervised. The performance is further improved by the method[1, 8, 4] using only the positive pair rather than the method[2, 9] using both the positive and negative pairs. This improved performance is observed because the use of a negative pair may include a tentative positive image among negative images, and a positive image may be misclassified as a negative pair. However, there is no need to find a potential positive sample belonging to a negative sample since there was no negative batch in the previous methods. Still, a positive batch can potentially have a positive sample of the same class.

In this paper, we present a method that can be used in addition to the contrastive learning method using only positive pairs. Negative samples are used by implementing cosine similarities of negative pairs, but the performance can be improved regardless of whether they are of the same class or not in a mini-batch. The performance can be further enhanced irrespective of whether the image is of the same class in the mini-batch by minimizing the cosine similarity distance of negative pairs, even if negative samples are used. Thus, a correlation is made by considering the cosine similarity for a tentative positive sample in a mini-batch. To prove this, we evaluated our method using several standard self-supervised benchmarks. In particular, we achieved 68.4% top-1 accuracy with a standard ResNet-50 on the ImageNet linear evaluation protocol. The contributions of this paper can be summarized as follows:

- We utilized cosine similarity for all images in mini-batch regardless of class.
- We made a correlation between positive samples among different images in mini-batch.
- We showed the best performance among state-of-the-art contrastive self-sueprvised methods.

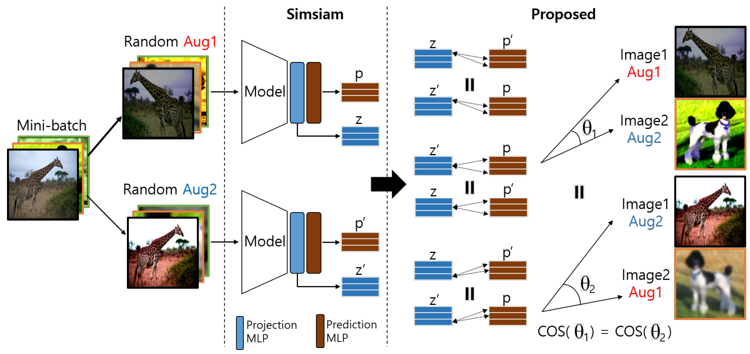


Figure 1: Proposed architecture.

2 Proposed Method

Our architecture(Fig. 1) utilizes the values of projection(z) and prediction(p) from the encoder structure of SimSiam [4]. $p_1 \triangleq q(g(f(v)))$ and $z_2 \triangleq g(f(v'))$ to minimize negative cosine similarity :

$$D(p_1, z_2) = -\frac{p_1}{\|p_1\|_2} \cdot \frac{z_2}{\|z_2\|_2}, \tag{1}$$

where $\|\cdot\|_2$ is ℓ_2-norm. This is equivalent to the mean squared error of ℓ_2-normalized vectors [8] up to a scale of 2.

The symmetric loss defined in Simsiam is as follows :

$$L_p = \frac{1}{2}D(p_1, z_2) + \frac{1}{2}D(p_2, z_1). \tag{2}$$

The contents of the proposed method are described in this section. We define $\boldsymbol{x} \cdot \boldsymbol{y} \triangleq \boldsymbol{x}^\top \boldsymbol{y}/(\|\boldsymbol{x}\|\|\boldsymbol{y}\|)$ in Fig. 2). The number of all cases that can be paired for z and p is expressed as a $2N$ x $2N$ matrix. This matrix($S_{i,j}$) is divided into quarters, as shown in $m_1(i \leq N, j \leq N)$, $m_2(i \leq N, j > N)$, $m_3(i > N, j \leq N)$, and $m_4(i > N, j > N)$. The diagonal element of each quartered matrix m represents a positive pair. As in $NN, N^{'}N, NN^{'}, N^{'}N^{'}$. N is related to the first random augmentation, and $N^{'}$ is related to the second random augmentation(Fig. 1). Except for gray, which is the diagonal element in Fig. 2, when the remaining elements are arranged horizontally and expressed as a one-dimensional array, each element is as shown in s_1, s_2, s_3, and s_4 as an element of S (equation 3). The equation representing only the negative pair is as follows:

$$M(p_1, z_2) = S\begin{bmatrix} M_1 - diag(M_1) & M_2 - diag(M_2) \\ M_3 - diag(M_3) & M_4 - diag(M_4) \end{bmatrix} = S\begin{bmatrix} S_1 & S_2 \\ S_3 & S_4 \end{bmatrix}, \tag{3}$$

where $diag$ denotes a diagonal matrix. M_1, M_2, M_3, and M_4 are elements when the $S_{i,j}$ matrix($2N$ x $2N$) is simply divided into four matrices($N \times N$), as shown in Fig. 2). The cosine similarities of negative pairs were minimized by applying the root mean square error (RMSE) to each of the four negative pairs. Multiplying by α in the previous formula is the proposed λ loss as:

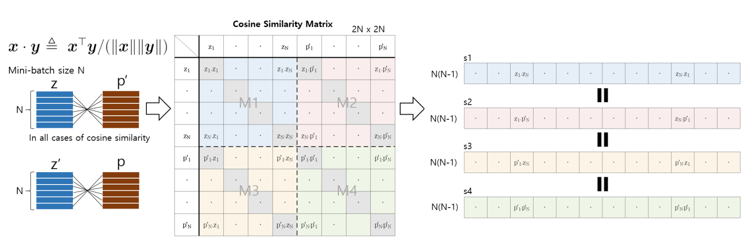


Figure 2: Cosine similarity extraction of negative pairs.

$$L_N = \|S_1 - S_2\| + \|S_2 - S_3\| + \|S_3 - S_4\|, \tag{4}$$

where $\|\cdot\|$ is the matrix $\ell_2$norm. In the L_N loss, each $S1$, $S2$, $S3$, and $S4$ matrices corresponding to the cosine similarity of the native pair are similar to each other.

The addition of our proposed L_N loss to the Simsiam loss is as:

$$L = L_P + L_N * \alpha, \tag{5}$$

where α is a scale parameter that determines the weight loss of L_N. L is the final loss that we use in our proposed method.

3 Experiments and Results

3.1 Evaluation on ImageNet training

As the most important experiment among many experiments, The pretrained model was tested with the unlabeled ImageNet dataset. Supervised training only linear classifiers without updating all network parameters using the standard linear evaluation protocol on ImageNet, as described in [12, 13, 3].

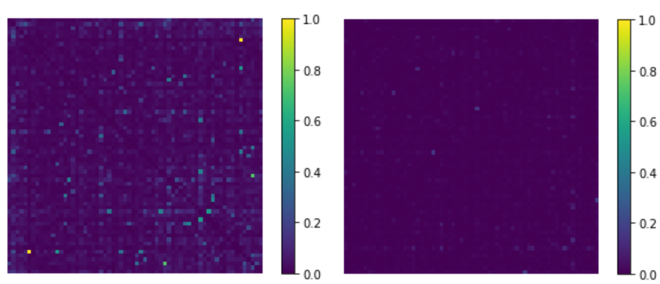
The results of the method excluding the proposed method are presented in the review article cited [4] as shown in Table 1. The momentum encoder method is inefficient because it uses two networks. It is very difficult to improve the performance in decimal units in self-supervised learning, but as can be seen in Table 1, the proposed method recorded 68.4% and showed 0.3% higher performance than the existing methods.

Experiments are performed using the scale parameter (α) values that directly affect the additional L_N loss from Table 2. In an experiment where alpha was set to 0.003, 0.01, and 0.02 values; 0.01 showed the best performance.

Fig. 3(a) and Fig. 3(b) show the average result of the difference in cosine similarity. It is expressed by dividing into four matrices, S1, S2, S3, and S4, in a single mini-batch experiment with batch size 64 as shown in equation 3. It is expressed in bright color and dark colour when there is a large and small difference in cosine similarity, respectively.

Method	Batch size	Negative pair	Momentum encoder	Top 1 acc(%)
SimCLR [2]	4096	✔		66.5
MoCo v2 [9]	**256**	✔	✔	67.4
BYOL [8]	4096		✔	66.5
SwAV [1]	4096			66.5
SimSiam [4]	**256**			68.1
Proposed	**256**	✔		**68.4**

Table 1: Comparisions on ImageNet linear classification with 100-epoch pre-training. ResNet-50 pre-trained with 224 × 224 views



(a) (b)

Figure 3: (a) Difference between the quadranted cosine similarity matrices (64 × 64) of Simsiam method. (b) Difference between the quadranted cosine similarity matrices (64 × 64) of the proposed method.

Alpha	0.003	0.008	0.01	0.012	0.02
Top 1 Acc (%)	68.09	68.30	**68.41**	68.23	68.26

Table 2: Various results of scale parameter alpha.

3.2 Transfer to other datasets

We experimented with self-supervised learning to determine how well features learned on very large datasets(such as ImageNet) when transferred to downstream tasks. In this experiment, we refer to the downstream transfer method with linear fine-tuning [7, 11].

Nine datasets were used as the downstream tasks and various types were used as: technical, texture, satellite, natural, medical, illustrative, symbolic, and natural. The datasets used are aircraft [17], Cars[14], DTD[5], EuroSAT[10], Flowers[18], ISIC[6], Kaokore[20], Omniglot[16], and Pets[19].

method	**epoch**	**aug++**	mean	Aircraft	Cars	DTD	EuroSAT	Flowers	ISIC	KaoKore	Omniglot	Pets
Supervised	90		65.15	31.05	40.68	64.68	93.85	85.20	72.21	76.98	32.95	**88.74**
Supervised	90	✔	64.13	33.54	41.03	61.49	91.00	82.48	68.72	73.57	37.43	87.93
Examplar-v2 [21]	200	✔	69.64	41.88	47.08	66.65	95.44	85.77	75.47	**78.44**	53.74	82.28
SimCLR [2]	200	✔	63.86	32.16	36.80	64.41	95.09	81.77	74.01	77.95	44.31	68.25
MoCo-v2 [9]	200	✔	69.69	41.01	44.92	68.40	**95.56**	85.87	76.34	**78.44**	57.69	79.01
BYOL [8]	300	✔	70.20	43.71	**55.28**	**68.72**	94.62	**89.01**	72.91	78.20	44.33	85.04
SimSiam [4]	100	✔	70.43	46.02	39.40	66.54	93.46	87.92	78.04	73.96	68.69	79.83
Proposed	100	✔	**71.04**	**46.26**	41.44	67.34	93.75	88.78	**78.10**	73.85	**69.83**	79.99

Table 3: Downstream transferring results with linear fine-tuning. "epoch" indicates their pre-training epochs and "aug++" indicates whether trained with data augmentation method of self-supervised learning [7].

The proposed method shows the best performance compared to other methods in the aircraft, ISIC, and Omniglot datasets, and the average accuracy of all datasets is the highest at 71.04%.

3.3 CIFAR Experiments

The experiment was also conducted using the CIFAR-10 dataset [15], similar to the training and linear evaluation experiments in ImageNet.

Similar to the ImageNet observations, the proposed method achieves a reasonable result and does not collapse. Additional kNN and linear evaluation experiments were conducted using only the L_N loss we proposed. Surprisingly, as shown in Table. 4, meaningful results were achieved using only negative pairs without using positive pairs(only negative pairs).

method	**train epoch**	**kNN eval**	**linear eval**
SimSiam [4]	800	87.08	91.72
Proposed	800	**87.59**	**91.90**
Only negative	800	81.30	88.08

Table 4: kNN and linear evaluation accuracy.

4 Conclusion

In this study, we proposed a method for minimizing the distance between the cosine similarities of negative pairs. Negative samples were used such that the cosine similarity with other negative samples in the mini-batch was similar for the two randomly augmented views. The similarity between the negatives was not enough in the method using only positive samples when the similarity difference was visualized in pixels to check whether the cosine similarity between the negative samples was similar. In contrast, our method looked very similar for that case. We achieved state-of-the-art performance with a Top-1 accuracy of 68.4%, which was higher than the existing methods SimCLR, MoCo, BYOL, SwAV, and Simsiam. These results were obtained with linear evaluation after 100-epoch training on an unlabeled ImageNet dataset. Surprisingly, we did not use a positive pair as a loss and showed meaningful results using only negative samples in an experiment performed with kNN and linear evaluation on the CIFAR-10 dataset.

References

[1] 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.

[2] 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.

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

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

[5] 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.

[6] Noel Codella, Veronica Rotemberg, Philipp Tschandl, M Emre Celebi, Stephen Dusza, David Gutman, Brian Helba, Aadi Kalloo, Konstantinos Liopyris, Michael Marchetti, et al. Skin lesion analysis toward melanoma detection 2018: A challenge hosted by the international skin imaging collaboration (isic). *arXiv preprint arXiv:1902.03368*, 2019.

[7] Yutong Feng, Jianwen Jiang, Mingqian Tang, Rong Jin, and Yue Gao. Rethinking supervised pre-training for better downstream transferring. *arXiv preprint arXiv:2110.06014*, 2021.

[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.

[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.

[10] Patrick Helber, Benjamin Bischke, Andreas Dengel, and Damian Borth. Eurosat: A novel dataset and deep learning benchmark for land use and land cover classification. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 12(7):2217–2226, 2019.

[11] Ashraful Islam, Chun-Fu Chen, Rameswar Panda, Leonid Karlinsky, Richard Radke, and Rogerio Feris. A broad study on the transferability of visual representations with contrastive learning. *arXiv preprint arXiv:2103.13517*, 2021.

[12] Alexander Kolesnikov, Xiaohua Zhai, and Lucas Beyer. Revisiting self-supervised visual representation learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 1920–1929, 2019.

[13] Simon Kornblith, Jonathon Shlens, and Quoc V Le. Do better imagenet models transfer better? In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2661–2671, 2019.

[14] 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.

[15] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009.

[16] Brenden M Lake, Ruslan Salakhutdinov, and Joshua B Tenenbaum. Human-level concept learning through probabilistic program induction. *Science*, 350(6266):1332–1338, 2015.

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

[18] 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.

[19] Luis Patino, Tom Cane, Alain Vallee, and James Ferryman. Pets 2016: Dataset and challenge. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 1–8, 2016.

[20] Yingtao Tian, Chikahiko Suzuki, Tarin Clanuwat, Mikel Bober-Irizar, Alex Lamb, and Asanobu Kitamoto. Kaokore: A pre-modern japanese art facial expression dataset. *arXiv preprint arXiv:2002.08595*, 2020.

[21] Nanxuan Zhao, Zhirong Wu, Rynson WH Lau, and Stephen Lin. What makes instance discrimination good for transfer learning? *arXiv preprint arXiv:2006.06606*, 2020.