Distance Enhancement Loss: Improving Separation of Visual Representations in Self-Supervised Learning

Junho Lee, HoJoon Song, Konstantin Schuetze, Ren Wang

Abstract

Self-supervised learning (SSL) methods achieve comparable performance with supervised learning methods. However, the distribution of embedding space in SSL is not separable clearly. To address this issue, this paper proposes a novel loss function, called Distance Enhancement (DE) loss, to enlarge the distance between the embedding space of different classes. A new metric is also proposed, which is Negative Log Absolute Determinant(NLAD) metric, to evaluate the quality of embedding space.

Experimental results show that the proposed DE loss can significantly improve the quality of embedding space for both SSL and supervised learning methods. It reduces 90.4% and 20.9% NLAD for ResNet-50 on the CIFAR-10 dataset with the SimCLR and supervised cross entropy learning paradigms, respectively. The code is available at `https://github.com/hojunroks/EmbeddingAnalysis`.

1. Introduction

Convolutional neural networks (CNNs) [14, 16, 22, 23] have achieved great success in the computer vision research community. With the sophisticated architectures, the performance in classification tasks [7] is significantly improved via supervised learning on large-scale datasets. However, supervised learning relies heavily on time-consuming and expensive data annotations. To address this issue, many self-supervised methods have been proposed to learn visual features from large-scale unlabeled images without human annotations.

Traditional self-supervised methods [2,8,9,11,17,19–21, 31, 32] learn representative features by accomplishing pretext tasks. Clustering-based methods [1, 3, 4, 26, 28, 29, 34] learn with unlabeled data in an end-to-end manner. The samples are clustered into n clusters, and each one is mapped with a corresponding class to evaluate the accuracy performance. Recently, contrastive methods [5, 6, 12, 13, 18, 24, 27, 30] achieve state-of-the-art performance in self-supervised learning. They aim at embedding augmented views of the same sample close to each other while pushing away embeddings from different samples in the latent space.

Although the performance of self-supervised methods is rapidly approaching the supervised learning methods, the distribution of the class-specific embedding space in self-supervised methods is not as well discriminative as supervised methods. As shown in Fig 2, the distance between the embedding space of two different classes is short in self-supervised methods, which means the visual representations from different classes are not separated well and may degrade the performance in downstream tasks, such as classification tasks. Intuitively, the samples from the same class should have high cosine similarity while low for different classes.

Based on that, this paper proposes a novel loss function, called Distance Enhancement loss, which helps SSL methods generate more separable embedding space. To evaluate the quality of the embedding space, a new metric, called Negative Log Absolute Determinant (NLAD) metric, is proposed based on the cosine similarity matrix. Experimental results show that the proposed method can help CNNs learn better visual representations in both self-supervised and supervised learning methods.

The contributions of this paper can be summarized as follows:

- We show that neither supervised and self-supervised methods are fully leveraging the embedding space and propose a novel problem.
- We propose Distance enhancement loss and show that CNNs can leverage a wider embedding space by applying it during the training.
- We propose a Negative Log Absolute Determinant (NLAD) metric to evaluate the quality of the embedding space generated by CNNs.

The rest of this paper is organized as follows. Section 2 briefly reviews self-supervised learning methods and unsupervised learning methods in classification tasks. The proposed loss function is presented in Section 3. Section 4 shows the experimental results of the proposed work while Section 5 summarizes our findings.

2. Related works

2.1. Self-Supervised Learning

Unsupervised visual representation learning, or self-supervised learning, aims at learning effective visual representations without data annotations. NPID [27] learns instance-level representations via a non-parametric softmax classifier and uses a memory bank to store them. MoCo [13] builds dynamic dictionaries with a queue and a moving-averaged encoder, which enables a large and consistent dictionary for learning visual representations. SimCLR [5] requires neither specialized architectures nor a memory bank and performs well through a composition of multiple data augmentation operations, a learnable nonlinear transformation, and large batch sizes. BYOL [12] learns representations without using negative pairs by iteratively bootstrapping the outputs of a network to serve as targets. The approach is more resilient to changes in the batch size and the set of image augmentations compared to SimCLR. Barlow Twins [30] learns representations through a joint embedding of distorted images and the proposed objective function naturally avoids collapse by measuring the cross-correlation matrix. The method does not require large batch sizes or any asymmetric mechanisms. SimSiam [6] avoids collapsing and can perform competitively without negative sample pairs, large batch sizes, and momentum encoders.

2.2. Unsupervised Learning

DeepCluster [3] jointly learns the parameters of a neural network and predicts the cluster assignments as pseudo-labels to update the weights. Compared to DeepCluster, SeLa [1] adds the constraint that the number of samples should be equal across clusters to avoid ill posed learning problems with degenerate solutions. SCAN [26] is a two-step framework that firstly learns feature representations through a pretext task and then clusters the semantically meaningful nearest neighbors as a prior into a learnable approach. NCD [33] adopts an end-to-end clustering technique via the use of pairwise similarity of samples and directly explore neighborhood by k-nearest neighbors.

3. Method

In this section, we introduce the existing self-supervised methods and formulate the problems through their limitations. We then describe our Distance Enhancement loss in detail.

3.1. Problem Formulation

Representation learning with SSL methods. Recent studies in self-supervised methods [5, 6, 12, 13, 18, 24, 27, 30] from representation learning aim to learn invariant representations for distorted images. More specifically, they produce two distorted views for all images of a batch X sampled from a dataset. The distorted views are obtained via a distribution of data augmentations. The two batches of distorted views are then fed to a deep network with trainable parameters, producing batches of embeddings. The training uses contrastive loss as objective function, an example of which is given in Eq.1,

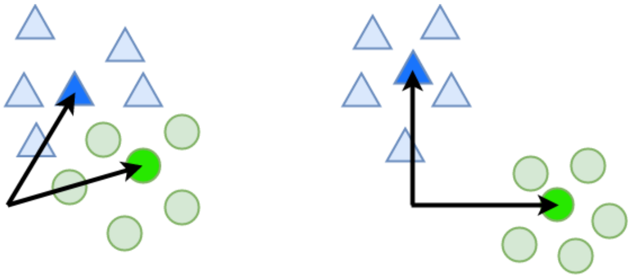


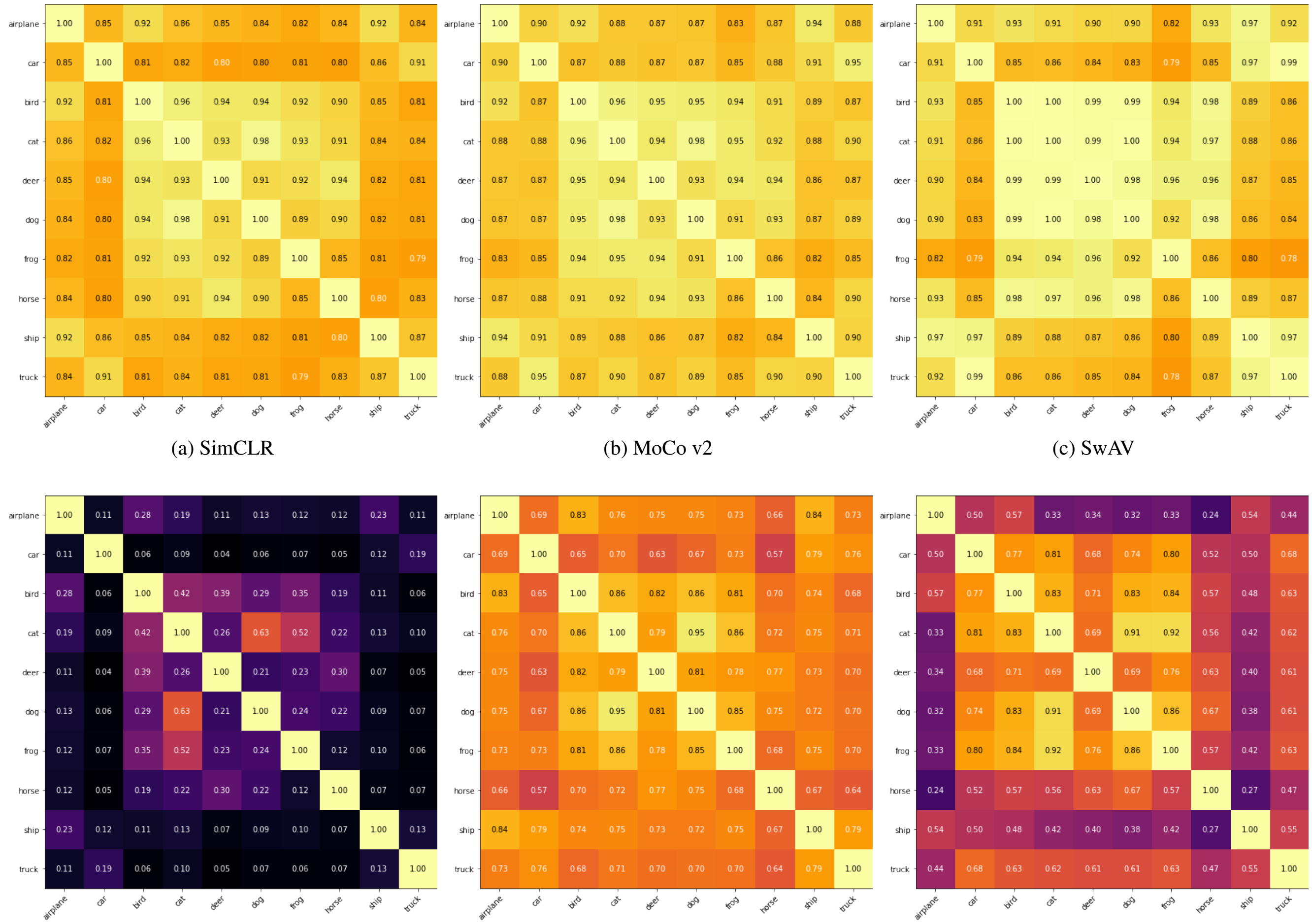
Figure 1. Left: The embedding space is not fully utilized. It cannot be seen as a good presentation because it has ambiguous boundaries. Right: The two clusters are clearly divided. The inputs are well embedded and make good use of the given space.

$$\ell_{i,j} = -\log \frac{exp(sim(z_i, z_j)/\tau)}{\sum_{k=1}^{2N} \mathbb{1}_{[k\neq i]} exp(sim(z_i, z_k)/\tau)}, \qquad (1)$$

where z denotes output of model, $\mathbb{1}_{[k\neq i]} \in 0, 1$ is an indicator function evaluating to 1 iff $k \neq i$, τ denotes a temperature, and N is the number of original image. In short, the denominator makes the distance between distorted views from the same data close, and the numerator makes the distance between distorted views from other data far, and the used distance metric is cosine similarity.

Limitations. If the model is trained well as designed by contrastive loss, the same class has a high cosine similarity value and different classes have low cosine similarity, so a large portion of the embedding space is used. However, as can be seen from Fig 2, which is the cosine similarities between the mean representation vector of each class, we can observe that the angle between the mean representation vectors of each class is not large enough.

We confirmed that this phenomenon also appears in supervised learning trained with cross entropy loss (see Fig 2 and Tab 2). The low distances between class clusters is means that they are not using the embedding space efficiently. If we use a wider embedding space, we can obtain better representations and better results for downstream tasks such as classification. To the best of our knowledge, there have been no studies to evaluate and tackle this problem.



(a) SimCLR

(b) MoCo v2

(c) SwAV

(d) Self-supervised (ours)

(e) Supervised (CE)

(f) Supervised (Ours)

Figure 2. **Cosine similarity matrix between mean representations** The matrix is visualized with color so that it could be easily recognized. Each block is brighter with a higher value and darker with a lower value. We can see that when our method is used, the cosine similarity between mean presentations definitely decreases. CE denotes cross-entropy loss.

3.2. Evaluation of Representations

One main focus of representation learning is not just to obtain clear representations, but also using the representations to apply it to downstream tasks such as object classification and detection. It is common that the representations themselves are fine-tuned in the process of training for these downstream tasks.

However, although the performance of downstream tasks is closely correlated with the quality of the representations, it is affected by many other factors, such as the training algorithm itself. Since our main purpose is how to create semantically reasonable representations in a larger representation space, we evaluate the representations without any further training.

First, we apply t-distributed stochastic neighbor embedding (t-SNE) [25] to reduce the dimensions of the representations. Since t-SNE itself clusters similar representations, we can assume that better representations will be clustered easily without any further operations.

Instead of evaluating on all of the samples, we define a confidence metric and evaluate on the confident samples C. This is an attempt to focus more on the representative power of the model on finding the clear samples near the cluster center, and not the ambiguous samples near the border. Since t-SNE embeds near-border samples closer to the center, the confidence scores are based on the distance of the embedding from the center.

For assignment of the labels to the clusters, any clustering method such as k-means clustering or agglomerative clustering can be used. We apply agglomerative clustering method to assign the labels. Then, the accuracy of the labels is evaluated using the hungarian assignment algorithm. The overall evaluation metric is shown in Algorithm 1.

Algorithm 1 Evaluation of Respresentation Pseudocode

```
# X: input (NxD)
# GT: ground truth (N)
# TSNE: t-SNE function
# F: encoder function
# k: the number of samples
# argpartition: function returning descending indices
# C: Clustering function

## sampling confident feature

# t-SNE feature: (N, 2)
zs = TSNE(F(X))

# calculate the confidence of features: (N, 1)
conf = sqrt(sum(zs.pow(2), dim=-1))

# indices of top k confident feature
indices = argpartition(conf)[:k]

# sampling k most confidence samples: (K, D)
sample, sample_gt = X[indices], GT[indices]

## Calculating clustering accuracy

acc = (C(F(sample)) == sample_gt) / k
```

3.3. Distance enhancement loss for negative pairs

As shown in Eq.1, the denominator of the contrastive loss sums the exponential values of the cosine similarity of the negative pairs so that they move away from each other. The loss was designed this way knowing that ensuring the proximity of positive pairs is a more important objective than keeping the negative pairs apart. The ensuing problem is that only a small portion of the embedding space is used. Therefore, we propose adding Distance enhancement loss, which forces the negative pairs apart from each other so that a wider portion of embedding space is used. The Distance enhancement loss is as follows:

$$\mathcal{D} = \frac{1}{(N-1)^2} \sum_{i=1}^{B} \sum_{j=1}^{B} \mathbb{1}_{[j \neq i]} sim(z_i, z_j), \tag{2}$$

where B is batch size. Adding Distance enhancement loss to the contrastive loss, balancing the two objectives by λ, we reach our novel loss function:

$$\mathcal{L} = \frac{1}{2B} \sum_{i=1}^{2B} -\log \frac{exp(sim(z_i, z_j)/\tau)}{\sum_{k=1}^{2B} \mathbb{1}_{[k \neq i]} exp(sim(z_i, z_k)/\tau)} + \lambda \mathcal{D}. \tag{3}$$

We confirm that adding a simple Distance enhancement loss the mean representations of the classes are strongly separated as shown in Fig 2.

Applying to supervised method. We can also add Distance enhancement loss to the cross entropy loss commonly use in supervised methods. Knowing the label, we use the mean representation vector m_i of each class in the batch, redefining the Enhancement loss as follows:

$$\mathcal{D} = \frac{1}{(C-1)^2} \sum_{i=1}^{C} \sum_{j=1}^{C} \mathbb{1}_{[j \neq i]} sim(m_i, m_j), \tag{4}$$

where C denotes the number of class. And the final loss for supervised learning with Distance enhancement loss is as follows:

$$\mathcal{L} = \frac{1}{B} \sum_{i=1}^{B} CE(y_i, p_i) + \lambda \mathcal{D}, \tag{5}$$

where CE denotes cross entropy loss, y denotes a label, and p is a prediction of the model.

3.4. Negative Log Absolute Determinant

We obtain a cosine similarity matrix by computing the distances between the mean representations of all members for each class. We propose Negative Log Absolute Determinant (NLAD) to measure how well the model learned to use the embedding space. NLAD is defined as follows:

$$\textbf{NLAD} = -\log |Det(M)|, \tag{6}$$

where M is cosine similarity matrix and $Det(\cdot)$ denotes determinant. If the mean representations are orthogonal to others except themselves, NLAD becomes 0. If they all have the same direction, it becomes infinity. Therefore, the smaller the NLAD value, the larger the embedding space is used. Mean representation did not exceed the right angle with each other, so such cases are excluded.

In Tab 2, we compared several methods by calculating NLAD.

4. Experiments

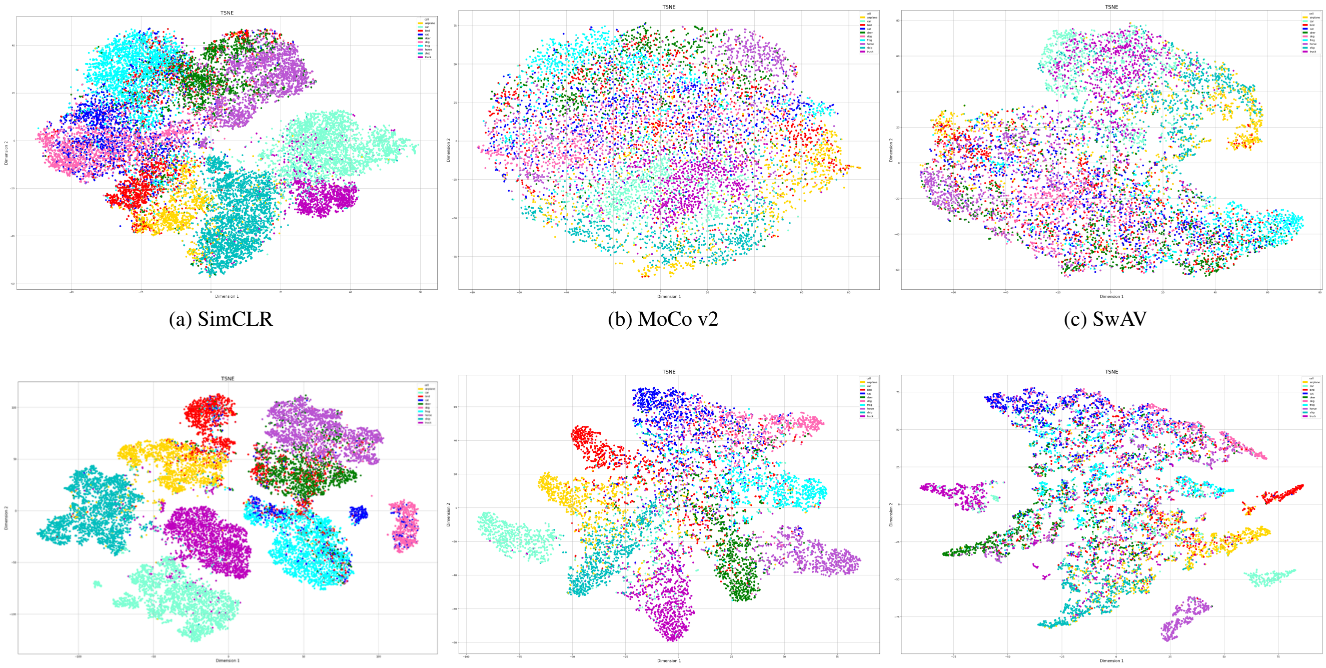
4.1. Experimental setup

Dataset. The experimental evaluation is performed on CIFAR10 [15]. We only proceeded with small dataset due to training time problems, but we encourage future experiments with large datasets.

Training setup. We use a ResNet-50 backbone. We adopt the training procedure from SimCLR [5], including its key strength, the data augmentation strategy, as well as the learning rate scheduling method and other training details. The hyperparameter λ multiplied to scale Distance

Table 1. **Classification Accuracy Ablation Study** In order to evaluate whether Distance Enhancement loss makes a better representation, the ablation study is conducted using a supervision method. As can be seen from the table, when our loss is added, it showed 1.01% better accuracy. CE denotes cross-entropy loss.

Method	Classification accuracy
Supervised (CE)	95.57%
Supervised (CE + ours)	96.58%



(a) SimCLR

(b) MoCo v2

(c) SwAV

(d) Self-Supervised (Ours)

(e) Supervised (CE)

(f) Supervised (CE+Ours)

Figure 3. CIFAR-10 t-SNE visualization in embedding space.

Enhancement loss is set to 0.1 in self-supervised learning and 1 in supervised learning.

For a fair comparison under our GPU, all the models compared by us use the implementation from Pytorch Lighting Bolts [10]. Since SimCLR [5], MoCo v2 [13], and SwAV [4] did not perform CIFAR10 training in the papers, we do hyperparameter tuning to achieve the good results.

4.2. Cosine similarity matrix analysis

We compared the cosine similarity between the mean representation vectors obtained from various pretrained self-supervised models(SimCLR, MoCo v2, SwAV) as shown in figure 2. The matrix shows that all the classes show similar representations. We can see in Fig 2d that the added Distance enhancement loss was successful at separating the mean representations from each other.

Table 2. **Negative Log Absolute Determinant (NLAD) Comparison** We used our proposed NLAD as a quantitative metric comparing the area of the embedding space. The lower the NLAD, the wider the model uses the embedding space. CE denotes cross-entropy loss.

Metric	**NLAD**	
Train Dataset	ImageNet	CIFAR10
Untrained	80.68	80.68
SimCLR	21.25	16.90
MoCo v2	24.84	22.92
SwAV	18.58	36.36
Ours	-	1.62
Supervised (CE)	26.38	12.46
Supervised (CE + ours)	-	9.86

This not only enhances the representative power of the model by using a larger latent space, but shows that the representation themselves are much better when we analyze the quantities of the similarities between classes. Since representation vectors are semantic embeddings of an image, we expect similar samples to have similar representation vectors.

This may seem obvious inside the class, but should also hold true between classes. By common knowledge we know that cats are similar to dogs, and trucks and horses are very different. Before applying the Distance enhancement loss, it is hard to say the representation vectors have successfully captured semantic characteristics since every vector is similar to another.

But as shown in Fig 2d, the similarity values coincide with common human knowledge. Given that the training was done self-supervised without any labels, the Distance enhancement loss helped capturing the semantic similarities and differences between each class correctly.

As trained with a label, it can be confirmed that the distance between mean presentation vectors has been learned farther than self-supervised methods. Cross-entropy loss does not explicitly separate negative pairs, but naturally pushes out each mean presentation vector to perform classification well. However, since it still shows distances that are not far enough, we trained the model using the supervised version of the distance enhancement loss. Given a large λ during training, the distance between the mean representation vector increases, but the model does not train well. So we train the model with small λ. As can be seen in Figure 2f, the mean presentation vectors of the model trained in our method utilize a larger space than before.

We also compared the cosine similarity between the mean representation vectors obtained from supervised learning model.

4.3. t-SNE visualization of embedding vectors

We performed t-SNE visualization on the embedding vectors pretrained with various self-supervised learning methods(MoCo V2, SwAV, SimCLR). Following the frequent convention of self-supervised learning, we used a model with ResNet50 as the backbone and pretrained on ImageNet and targeting to evaluate on smaller datasets. We've also evaluated a model that was pretrained on ImageNet using regular cross-entropy loss with supervised learning as baseline. The embedding vectors were taken from the final convolution layer, with CIFAR-10 test images given as input.

As transfer learning using self-supervised learning methods on bigger datasets have shown promising results on image classification, we expected that the embedding vectors would be clustered to a moderate extent for each class even before fine-tuning using labels. However, as shown in Figure 4 , the results show that the embedding vectors were not so nicely clustered. For some classes, the results do show some clustering, but for many the vectors were spread out and mixed up with each other.

We performed an additional experiment using a model that was pretrained with self-supervised learning using CIFAR-10 train images, as we expected that the low performance of clustering was due to the difference between our training and evaluating datasets. The difference between the distribution of the images, and the size of the images may have prevented the model from learning wanted features of the evaluation dataset. As we can see in Figure 3, the input images are somewhat clustered, although no labels were used during training.

However, we can still see that most of the samples are mixed together, hard to distinguish clusters from one another. This means that the representations are similarly mapped to the latent space. When we applied Distance Enhancement loss with confidence, we could easily see that the clusters were clearly separated from each other.

This indicates that the Distance Enhancement loss was successful at clustering representation vectors by separating them apart from each other. The ground truth labels at (d) show the clear border between the samples.

Table 3. **Evaluation of Representations** Comparison of the classification accuracy calculated according to algorithm 1.

Method	Classification accuracy
Self-Supervised (SimCLR)	0.56
Self-Supervised (Ours)	0.84

However, as shown in (d), we can see that many blue labels(cats) were excluded by the confidence-based filtering. This means that the model has difficulties in distinguishing classes that are similar to many other classes. Since the main effect of the Distance Enhancement loss is to map images to a larger latent space by separating distinct representation vectors, the model may lack the ability to create a robust gap between similar classes, and thus be much less confident in clustering similar classes. But for distinct classes, our method shows clear separation between classes that are less confusing.

As an additional experiment, we perform t-SNE visualization of the model learned by supervised learning using cross-entropy loss. Fig 4d shows that the model learned through cross-entropy has good embedding clusters, but the distance between clusters is close and has quite a lot of overlapping.

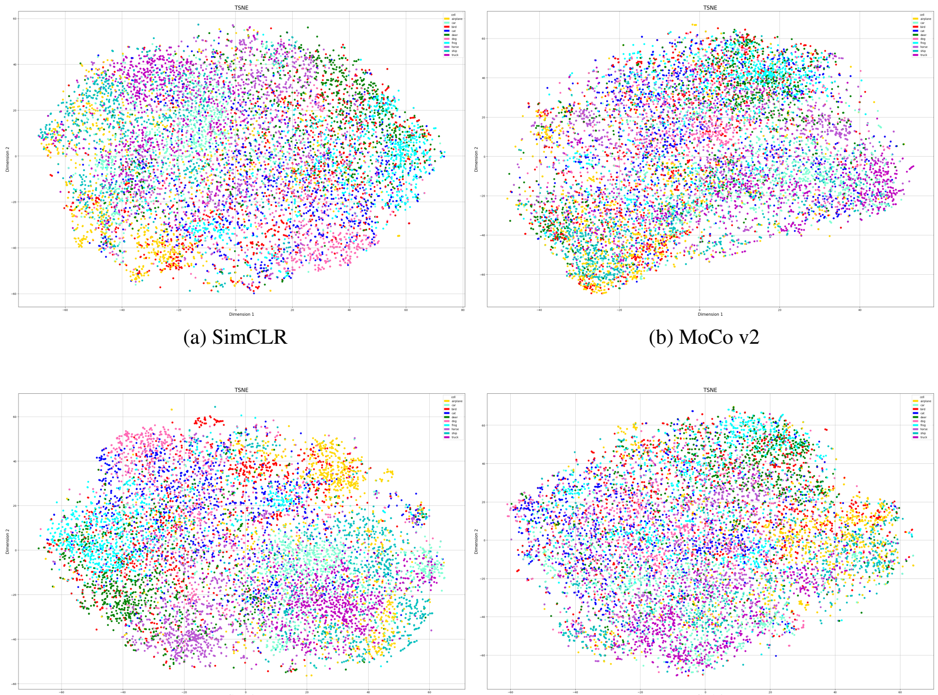
To improve the problems, we train the model with the supervised version of Distance Enhancement loss mentioned in Section 3.3 and perform t-SNE visualization (3f). Contrary to expectations, the t-SNE result is not good, but the distance between clusters becomes far and the variance of some clusters has become very small. And most of all, the classification accuracy is improved as shown in Tab 1.

4.4. Quantitative evaluation of representations

Table 3 shows the result of quantitative evaluation performed on our model following the evaluation process of algorithm 1. Even without any further training, just by assigning labels using agglomerative clustering, we could gain a classification accuracy of 0.84. Although the accuracy was calculated using only confident samples, the results are interesting as they were achieved with a single end-to-end self-supervised training process.

5. Conclusion

We have shown how previous representations do not fully use the possible embedding space, and introduced the NLAD metric to evaluate embedding spaces. Our key contribution is the novel Distance enhancement loss, which makes the model use a wide embedding space. This improves the NLAD, produces a better clustered t-SNE projection and increases classification accuracy compared to the original cross-entropy loss. We therefore see Distance en-



(a) SimCLR

(b) MoCo v2

(c) SwAV

(d) Supervised (CE)

Figure 4. CIFAR-10 t-SNE in the model trained with ImageNet.

hancement loss a promising tool for contrastive learning in general and for self-supervised image representation learning specifically.

References

[1] Yuki Markus Asano, Christian Rupprecht, and Andrea Vedaldi. Self-labelling via simultaneous clustering and representation learning. In *Proc. Int. Conf. Learn. Represent. (ICLR)*, 2020. 1, 2

[2] Piotr Bojanowski and Armand Joulin. Unsupervised learning by predicting noise. In *Int. Conf. Mach. Learn. (ICML)*, pages 517–526. PMLR, 2017. 1

[3] Mathilde Caron, Piotr Bojanowski, Armand Joulin, and Matthijs Douze. Deep clustering for unsupervised learning of visual features. In *Proc. Eur. Conf. Comput. Vis. (ECCV)*, pages 132–149, 2018. 1, 2

[4] Mathilde Caron, Ishan Misra, Julien Mairal, Priya Goyal, and Bojanowski. Unsupervised learning of visual features by contrasting cluster assignments. In *Adv. Neural Inf. Process. Syst. (NeurIPS)*, 2020. 1, 5

[5] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *Int. Conf. Mach. Learn. (ICML)*, pages 1597–1607. PMLR, 2020. 1, 2, 4, 5

[6] Xinlei Chen and Kaiming He. Exploring simple siamese representation learning. In *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pages 15750–15758, 2021. 1, 2

[7] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pages 248–255, 2009. 1

[8] Carl Doersch, Abhinav Gupta, and Alexei A Efros. Unsupervised visual representation learning by context prediction. In *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, pages 1422–1430, 2015. 1

[9] Alexey Dosovitskiy, Philipp Fischer, Jost Tobias Springenberg, Martin Riedmiller, and Thomas Brox. Discriminative unsupervised feature learning with exemplar convolutional neural networks. *IEEE Trans. Pattern Anal. Mach. Intell. (TPAMI)*, 38(9):1734–1747, 2015. 1

[10] William Falcon and Kyunghyun Cho. A framework for contrastive self-supervised learning and designing a new approach. *arXiv preprint arXiv:2009.00104*, 2020. 5

[11] Spyros Gidaris, Praveer Singh, and Nikos Komodakis. Unsupervised representation learning by predicting image rotations. In *Proc. Int. Conf. Learn. Represent. (ICLR)*, 2018. 1

[12] 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, Bilal Piot, koray kavukcuoglu, Remi Munos, and Michal Valko. Bootstrap your own latent: A new approach to self-supervised learning. In *Adv. Neural Inf. Process. Syst. (NeurIPS)*, 2020. 1, 2

[13] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual repre-

sentation learning. In *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pages 9729–9738, 2020. 1, 2, 5

[14] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pages 770–778, 2016. 1

[15] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009. 4

[16] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Adv. Neural Inf. Process. Syst. (NeurIPS)*, pages 1106–1114, 2012. 1

[17] Gustav Larsson, Michael Maire, and Gregory Shakhnarovich. Learning representations for automatic colorization. In *Proc. Eur. Conf. Comput. Vis. (ECCV)*, pages 577–593. Springer, 2016. 1

[18] Ishan Misra and Laurens van der Maaten. Self-supervised learning of pretext-invariant representations. In *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pages 6707–6717, 2020. 1, 2

[19] Mehdi Noroozi and Paolo Favaro. Unsupervised learning of visual representations by solving jigsaw puzzles. In *Proc. Eur. Conf. Comput. Vis. (ECCV)*, pages 69–84. Springer, 2016. 1

[20] Mehdi Noroozi, Hamed Pirsiavash, and Paolo Favaro. Representation learning by learning to count. In *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, pages 5898–5906, 2017. 1

[21] Deepak Pathak, Philipp Krahenbuhl, Jeff Donahue, Trevor Darrell, and Alexei A Efros. Context encoders: Feature learning by inpainting. In *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pages 2536–2544, 2016. 1

[22] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. In *Proc. Int. Conf. Learn. Represent. (ICLR)*, 2015. 1

[23] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. In *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pages 1–9, 2015. 1

[24] Yonglong Tian, Dilip Krishnan, and Phillip Isola. Contrastive multiview coding. In *Proc. Eur. Conf. Comput. Vis. (ECCV)*, pages 776–794. Springer, 2020. 1, 2

[25] Laurens Van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of machine learning research*, 9(11), 2008. 3

[26] Wouter Van Gansbeke, Simon Vandenhende, Stamatios Georgoulis, Marc Proesmans, and Luc Van Gool. Scan: Learning to classify images without labels. In *Proc. Eur. Conf. Comput. Vis. (ECCV)*, pages 268–285. Springer, 2020. 1, 2

[27] Zhirong Wu, Yuanjun Xiong, Stella X Yu, and Dahua Lin. Unsupervised feature learning via non-parametric instance discrimination. In *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pages 3733–3742, 2018. 1, 2

[28] Xueting Yan, Ishan Misra, Abhinav Gupta, Deepti Ghadiyaram, and Dhruv Mahajan. Clusterfit: Improving generalization of visual representations. In *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pages 6509–6518, 2020. 1

[29] Jianwei Yang, Devi Parikh, and Dhruv Batra. Joint unsupervised learning of deep representations and image clusters. In *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pages 5147–5156, 2016. 1

[30] Jure Zbontar, Li Jing, Ishan Misra, Yann LeCun, and Stéphane Deny. Barlow twins: Self-supervised learning via redundancy reduction. In *Int. Conf. Mach. Learn. (ICML)*. PMLR, 2021. 1, 2

[31] Liheng Zhang, Guo-Jun Qi, Liqiang Wang, and Jiebo Luo. Aet vs. aed: Unsupervised representation learning by autoencoding transformations rather than data. In *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pages 2547–2555, 2019. 1

[32] Richard Zhang, Phillip Isola, and Alexei A Efros. Split-brain autoencoders: Unsupervised learning by cross-channel prediction. In *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pages 1058–1067, 2017. 1

[33] Zhun Zhong, Enrico Fini, Subhankar Roy, Zhiming Luo, Elisa Ricci, and Nicu Sebe. Neighborhood contrastive learning for novel class discovery. In *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, pages 10867–10875, 2021. 2

[34] Chengxu Zhuang, Alex Lin Zhai, and Daniel Yamins. Local aggregation for unsupervised learning of visual embeddings. In *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, pages 6002–6012, 2019. 1

A. Role description

Lee Junho (leader)

- Offer ideas
- Code (Idea Experiment, Visualization, Comparison group)
- Writing (Intro, Method, Result)

Song HoJun

- Offer ideas
- Code (Idea Experiment, Main Method, Visualization)
- Writing (Method, Result)

Konstantin

- Presentation
- Writing (Related works, Conclusion)
- Visualization (Fig 1, 2)

Wang Ren

- PPT
- Writing (Abstract, Intro, Related works)

]