Mutual Contrastive Learning for Visual Representation Learning

Chuanguang Yang[1,2], **Zhulin An**[1*], **Linhang Cai**[1,2], **Yongjun Xu**[1]

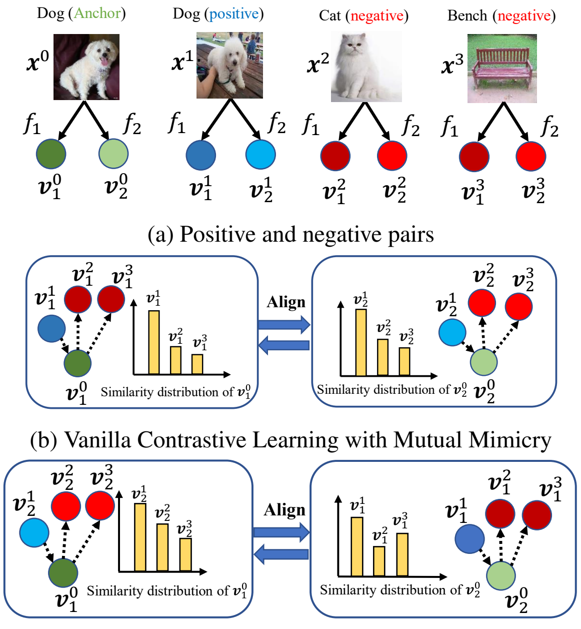
[1]Institute of Computing Technology, Chinese Academy of Sciences, Beijing, China
[2]University of Chinese Academy of Sciences, Beijing, China
{yangchuanguang, anzhulin, cailinhang19g, xyj}@ict.ac.cn

Abstract

We present a collaborative learning method called Mutual Contrastive Learning (MCL) for general visual representation learning. The core idea of MCL is to perform mutual interaction and transfer of contrastive distributions among a cohort of networks. A crucial component of MCL is Interactive Contrastive Learning (ICL). Compared with vanilla contrastive learning, ICL can aggregate cross-network embedding information and maximize the lower bound to the mutual information between two networks. This enables each network to learn extra contrastive knowledge from others, leading to better feature representations for visual recognition tasks. We emphasize that the resulting MCL is conceptually simple yet empirically powerful. It is a generic framework that can be applied to both supervised and self-supervised representation learning. Experimental results on image classification and transfer learning to object detection show that MCL can lead to consistent performance gains, demonstrating that MCL can guide the network to generate better feature representations. Code is available at https://github.com/winycg/MCL.

Introduction

Contrastive learning has been widely demonstrated as an effective framework for both supervised (Schroff, Kalenichenko, and Philbin 2015; Khosla et al. 2020) and self-supervised (Luo et al. 2020; Yao et al. 2020; He et al. 2020; Chen et al. 2020b; Li et al. 2021; Zhang et al. 2021) visual representation learning for artificial intelligence applications (Xu et al. 2021). The core idea of contrastive learning is to pull positive pairs together and push negative pairs apart in the feature embedding space by a contrastive loss. The current pattern of contrastive learning consists of two aspects: (1) how to define positive and negative pairs; (2) how to form a contrastive loss. The main difference between supervised and self-supervised contrastive learning lies in the aspect (1). In the supervised scenario, labels often guide the definition of contrastive pairs. A positive pair is formed by two samples from the same class, while two samples from different classes form a negative pair. In the self-supervised scenario, since we do not have label information, a positive pair is often formed by two views (*e.g.* different data augmentations) of the same sample, while negative pairs are formed by different samples. Given the positive and negative pairs, we can apply a contrastive loss to generate a meaningful feature embedding space. In general, loss functions are independent of how to define pairs. This paper focuses on a generic mutual contrastive loss among multiple networks.



(a) Positive and negative pairs

(b) Vanilla Contrastive Learning with Mutual Mimicry

(c) Interactive Contrastive Learning with Mutual Mimicry

Figure 1: Overview of the proposed *Mutual Contrastive Learning*. f_1 and f_2 denote two different networks. v_m^i is the embedding vector inferred from f_m with the input sample x^i. The dashed and dotted arrow denotes the direction we want to push close or apart by a contrastive loss. We also perform mutual alignment between two different softmax-based similarity distributions.

Beyond feature embedding-based contrastive learning, another vein for supervised learning focuses on logit-based learning. The conventional way is to train a network using cross-entropy loss between predictive class probability distribution and the one-hot ground-truth label. Some logit-based online Knowledge Distillation (KD) (Zhang et al.

[*]Corresponding author.

2018; Zhu, Gong et al. 2018; Song and Chai 2018) methods demonstrate that a cohort of models can benefit from mutual learning of *class probability distributions*. Each model in such a peer-teaching manner learns better compared with learning alone in conventional supervised training. From this perspective, we hypothesize that it may be desirable to perform *mutual contrastive learning* among a cohort of models for learning better *feature representations*. Unlike the class posterior, feature embeddings contain structured knowledge and are more tractable to capture dependencies among various networks. However, existing works (Khosla et al. 2020; He et al. 2020; Chen et al. 2020b) often train a single network to encode data points and apply contrastive learning for its own feature embedding space. Thus it makes sense to take advantage of collaborative learning for better visual representation learning.

To this end, we propose a simple *Mutual Contrastive Learning* (MCL) framework. The main core of MCL is to perform mutual interaction and transfer of contrastive distributions among a cohort of models. MCL includes Vanilla Contrastive Learning (VCL) and Interactive Contrastive Learning (ICL). Compared with the conventional VCL, our proposed ICL forms contrastive similarity distributions between diverse embedding spaces derived from two different networks. We demonstrate that the objective of ICL is equivalent to maximizing the lower bound to the mutual information between two peer networks. This can be understood to capture dependencies and enable a network to learn extra contrastive knowledge from another network.

Inspired by the idea of DML (Zhang et al. 2018), we also perform mutual alignment between different softmax-based contrastive distributions from various networks formed by the same data samples. Similar to DML (Zhang et al. 2018), the distributions can be seen as soft labels to supervise others. Such a peer-teaching manner with soft labels takes advantage of representation information embedded in different networks. Over two types of contrastive learning, we can derive *soft VCL label* and *soft ICL label*. Although the soft VCL label has been applied in previous KD works (Ge, Chen, and Li 2020; Fang et al. 2021), its anchor and contrastive embeddings are still formed from the same network, limiting the information interactions. Instead, our proposed soft ICL label aggregates cross-network embeddings to construct contrastive distributions, which is demonstrated to be more informative than the conventional soft VCL label.

To maximize the effectiveness of MCL, we summarize VCL and ICL with mutual mimicry into a unified framework, as illustrated in Fig. 1. MCL helps each model capture extra contrastive knowledge to construct a better representation space. Inspired by DML, since networks start from different initial conditions, each one can learn knowledge that others have not. In fact, MCL can be regarded as a group-wise contrastive loss and is orthogonal to defining positive and negative pairs. Therefore, we can readily apply MCL for both supervised and self-supervised contrastive learning.

We apply MCL to representation learning for a broad range of visual tasks, including supervised and self-supervised image classification and transfer learning to object detection. MCL can lead to consistent performance gains upon the baseline methods. Note that collaborative learning among a cohort of models is conducted during the training stage. Any network in the cohort can be kept during the inference stage. Compared with the original network, the kept network does not introduce additional inference costs.

Our main contributions are listed as follows: (1) We propose a MCL framework that aims to facilitate mutual interaction and transfer of contrastive knowledge among a cohort of models. (2) MCL is a new collaborative training scheme in terms of representation learning. It is a simple yet powerful framework that can be applied to both supervised and self-supervised representation learning. (3) Thorough experimental results show that MCL can lead to significant performance improvements across frequently-used visual tasks.

Related Work

Contrastive Learning. Contrastive learning has been extensively exploited for both supervised and self-supervised visual representation learning. The main idea of contrastive learning is to push positive pairs close and negative pairs apart by a contrastive loss (Hadsell, Chopra, and LeCun 2006) to obtain a discriminative space. For supervised learning, contrastive learning is often used for image classification (Khosla et al. 2020) and deep metric learning (Schroff, Kalenichenko, and Philbin 2015). Recently, self-supervised contrastive learning can guide networks to learn general features and achieve state-of-the-art performance for downstream visual recognition tasks. The core idea is to learn invariant representations over human-designed pretext tasks by a contrastive loss (Oord, Li, and Vinyals 2018). Typical pretext tasks are jigsaw (Noroozi and Favaro 2016) used in PIRL (Misra and Maaten 2020) and data augmentations used in SimCLR (Chen et al. 2020b) and MoCo (He et al. 2020). This paper does not design a new contrastive method. Instead, our focus is to propose a generic mutual contrastive learning framework. We can incorporate MCL with the above advanced contrastive works to learn better feature representations by taking advantage of collaborative learning. Although previous MVCL (Yang, An, and Xu 2021) also conducts contrastive representation learing with multiple networks, it does not perform explicit transfer of contrastive distributions. Moreover, MVCL can only be applied to the supervised scenario and can not generalize to self-supervised learning.

Collaborative Learning. The idea of collaborative learning has been explored in online knowledge distillation. DML (Zhang et al. 2018) shows that a group of models can benefit from mutual learning of predictive class probability distributions. CL (Song and Chai 2018) further extends this idea to a hierarchical architecture with multiple classifier heads. PCL (Wu and Gong 2021) introduces an extra temporal mean network for each peer as the teacher role. HSSAKD (Yang et al. 2021b) proposes mutual transfer of self-supervision augmented distributions by extending the teacher-student based counterpart (Yang et al. 2021a). In contrast to mutual mimicry, ONE (Zhu, Gong et al. 2018), OKDDip (Chen et al. 2020a) and KDCL (Guo et al. 2020) construct an online teacher via a weighted ensemble logit distribution but differ in various aggregation strategies. Beyond logit level, we take advantage of collaborative learning from the perspective of *representation learning*. Moreover, we can readily incorporate MCL with previous logit-based methods together.

Embedding-based Relation Distillation. Compared with the final class posterior, the latent feature embeddings encapsulates more structural information. Some previous KD methods transfer the embedding-based relational graph where each node represents one sample (Park et al. 2019; Peng et al. 2019). More recently, MMT (Ge, Chen, and Li 2020) employs soft softmax-triplet loss to learn relative similarities from other networks for unsupervised domain adaptation on person Re-ID. To compress networks over self-supervised MoCo (He et al. 2020), SEED (Fang et al. 2021) transfers soft InfoNCE-based (Oord, Li, and Vinyals 2018) contrastive distributions from a teacher to a student. A common characteristic of previous works is that contrastive distributions are often constructed from the same embedding space, restricting peer information interactions. Instead, we aggregate cross-network embeddings to model interactive contrastive distributions.

Methodology

Collaborative Learning Architecture

Notation. A classification network $f(\cdot)$ can be divided into a feature extractor $\varphi(\cdot)$ and a linear FC layer $FC(\cdot)$. f maps an input image $\boldsymbol{x}$ to a logit vector $\boldsymbol{z}$, *i.e.* $\boldsymbol{z} = f(\boldsymbol{x}) = FC(\varphi(\boldsymbol{x}))$. Moreover, we add an additional projection module $\phi(\cdot)$ that includes two sequential FC layers with a middle ReLU. $\phi(\cdot)$ is to transform a feature embedding from the feature extractor $\varphi(\cdot)$ into a latent embedding $\boldsymbol{v} \in \mathbb{R}^d$, *i.e.* $\boldsymbol{v} = \phi(\varphi(\boldsymbol{x}))$, where d is the embedding size. The embedding $\boldsymbol{v}$ is used for contrastive learning.

Training Graph. The overall training graph contains $M(M \geqslant 2)$ classification networks denoted by $\{f_m\}_{m=1}^M$ for collaborative learning. When $M = 2$, we use two independent networks f_1 and f_2. When $M > 2$, the low-level feature layers across $\{f_m\}_{m=1}^M$ are shared to reduce the training complexity. All the same networks in the cohort are initialized with various weights to learn diverse representations. This is a prerequisite for the success of knowledge mutual learning. Each f_m in the cohort is equipped with an additional embedding projection module ϕ_m. The overall training graph is shown in Fig. 3.

Inference Graph. During the test stage, we discard all projection modules $\{\phi_m\}_{m=1}^M$ and keep one network for inference. The architecture of the kept network is identical to the original network. That is to say that we do not introduce extra inference costs. Moreover, we can select any f_m in the cohort for the final deployment.

Mutual Contrastive Learning

Vanilla Contrastive Learning (VCL). The idea of contrastive loss is to push positive pairs close and negative pairs apart in the latent embedding space. Given an input sample $\boldsymbol{x}^0$ as the anchor, we can obtain 1 positive sample $\boldsymbol{x}^1$ and $K(K \geqslant 1)$ negative samples $\{\boldsymbol{x}^k\}_{k=2}^{K+1}$. For supervised learning, the positive sample is from the same class with the anchor, while negative samples are from different classes. For self-supervised learning, the anchor and positive samples are often two copies from different augmentations applied on the same instance, while negative samples are different instances. For ease of notation, we denote the anchor embedding as $\boldsymbol{v}_m^0$, the positive embedding as $\boldsymbol{v}_m^1$ and K negative embeddings as $\{\boldsymbol{v}_m^k\}_{k=2}^{K+1}$. m represents that the embedding is generated from f_m. Here, feature embeddings are preprocessed by l_2-normalization.

We use the dot product to measure similarity distribution between the anchor and contrastive embeddings with *softmax* normalization. Thus, we can obtain contrastive probability distribution $\boldsymbol{p}_m = softmax([(\boldsymbol{v}_m^0 \cdot \boldsymbol{v}_m^1/\tau), (\boldsymbol{v}_m^0 \cdot \boldsymbol{v}_m^2/\tau), \cdots, (\boldsymbol{v}_m^0 \cdot \boldsymbol{v}_m^{K+1}/\tau)])$, where τ is a constant temperature. $\boldsymbol{p}_m$ measures the relative sample-wise similarities with a normalized probability distribution. A large probability represents a high similarity between the anchor and a contrastive embedding. We use cross-entropy loss to force the positive pair close and negative pairs away upon the contrastive distribution $\boldsymbol{p}_m$:

$$\mathcal{L}_m^{VCL} = -\log \boldsymbol{p}_m^1 = -\log \frac{\exp(\boldsymbol{v}_m^0 \cdot \boldsymbol{v}_m^1/\tau)}{\sum_{k=1}^{K+1} \exp(\boldsymbol{v}_m^0 \cdot \boldsymbol{v}_m^k/\tau)}. \quad (1)$$

Here, $\boldsymbol{p}_m^k$ is the k-th element of $\boldsymbol{p}_m$. This loss is equivalent to a $(K+1)$-way softmax-based classification loss that forces the network to classify the positive sample correctly. In fact, the form of Eq.(1) is an InfoNCE loss (Oord, Li, and Vinyals 2018), which has been widely used in recent self-supervised contrastive learning (He et al. 2020). When applying contrastive learning to a cohort of M networks, the vanilla method is to summarize each contrastive loss:

$$\mathcal{L}_{1\sim M}^{VCL} = \sum_{m=1}^{M} (\mathcal{L}_m^{VCL}). \quad (2)$$

Interactive Contrastive Learning (ICL). However, vanilla contrastive learning does not model cross-network relationships for collaborative learning. This is because the contrastive distribution is learned from the network's own embedding space. To take full advantage of information interaction among various peer networks, we propose a novel *Interactive Contrastive Learning* (ICL) to model cross-network interactions to learn better feature representations. We formulate ICL for the case of two parallel networks f_a and f_b, where $a, b \in \{1, 2, \cdots, M\}, a \neq b$, and then further extend ICL to more than two networks among $\{f_m\}_{m=1}^M$.

To conduct ICL, we first fix f_a and enumerate over f_b. Given the anchor embedding $\boldsymbol{v}_a^0$ extracted from f_a, we enumerate the positive embedding $\boldsymbol{v}_b^1$ and negative embeddings $\{\boldsymbol{v}_b^k\}_{k=2}^{K+1}$ extracted from f_b. Here, both $\{\boldsymbol{v}_a^k\}_{k=0}^{K+1}$ and $\{\boldsymbol{v}_b^k\}_{k=0}^{K+1}$ are generated from the same $K+1$ samples $\{\boldsymbol{x}^k\}_{k=0}^{K+1}$ correspondingly, as illustrated in Fig. 1a. The contrastive probability distribution from f_a to f_b can be formulated as $\boldsymbol{q}_{a\to b} = softmax([(\boldsymbol{v}_a^0 \cdot \boldsymbol{v}_b^1/\tau), (\boldsymbol{v}_a^0 \cdot$

$\boldsymbol{v}_b^2/\tau), \cdots, (\boldsymbol{v}_a^0 \cdot \boldsymbol{v}_b^{K+1}/\tau)])$. Similar to Eq.(1), we use cross-entropy loss upon the contrastive distribution $\boldsymbol{q}_{a\to b}$:

$$\mathcal{L}_{a\to b}^{ICL} = -\log \boldsymbol{q}_{a\to b}^1 = -\log \frac{\exp(\boldsymbol{v}_a^0 \cdot \boldsymbol{v}_b^1/\tau)}{\sum_{k=1}^{K+1} \exp(\boldsymbol{v}_a^0 \cdot \boldsymbol{v}_b^k/\tau)}. \quad (3)$$

Here, $\boldsymbol{q}_{a\to b}^k$ is the k-th element of $\boldsymbol{q}_{a\to b}$. We can observe that the main difference between Eq.(1) and Eq.(3) lies in various types of embedding space for generating contrastive distributions. Compared with Eq.(1), Eq.(3) employs contrastive embeddings from another network instead of the network's own embedding space. It can model explicit corrections or dependencies in various embedding spaces among multiple peer networks, facilitating information communications to learn better feature representations.

Theoretical Analysis. Compared to Eq.(1), we attribute the superiority of minimizing Eq.(3) to maximizing the lower bound on the mutual information $I(\boldsymbol{v}_a^0, \boldsymbol{v}_b^1)$ between f_a and f_b, which is formulated as:

$$I(\boldsymbol{v}_a^0, \boldsymbol{v}_b^1) \geq \log(K) - \mathbb{E}_{(\boldsymbol{v}_a^0, \boldsymbol{v}_b^1)} \mathcal{L}_{a\to b}^{ICL}. \quad (4)$$

Inspired by (Tian, Krishnan, and Isola 2020), detailed proof from Eq.(3) to derive Eq.(4) is provided in Appendix. Intuitively, the mutual information $I(\boldsymbol{v}_a^0, \boldsymbol{v}_b^1)$ measures the reduction of uncertainty in contrastive feature embeddings from f_b when the anchor embedding from f_a is known. This can be understood that each network could gain extra contrastive knowledge from others benefiting from Eq.(3). Thus, it can lead to better representation learning than independent contrastive learning of Eq.(1). As K increases, the mutual information $I(\boldsymbol{v}_a^0, \boldsymbol{v}_b^1)$ would be higher, indicating that f_a and f_b could learn more common knowledge from each other.

When extending to $\{f_m\}_{m=1}^M$, we perform ICL in every two of M networks to model fully connected dependencies, leading to the overall loss as:

$$\mathcal{L}_{1\sim M}^{ICL} = \sum_{1\leq a<b\leq M}^{M} (\mathcal{L}_{a\to b}^{ICL} + \mathcal{L}_{b\to a}^{ICL}) \quad (5)$$

Soft Contrastive Learning with Online Mutual Mimicry The success of *Deep Mutual Learning* (Zhang et al. 2018) suggests that each network can generalize better from mutually learning other networks' soft class probability distributions in an online peer-teaching manner. This is because the output of class posterior from each network can be seen as a natural *soft label* to supervise others. Based on this idea, it is desirable to derive soft contrastive distributions as *soft labels* from contrastive learning, for example, $\boldsymbol{p}_m$ from VCL and $\boldsymbol{q}_{a\to b}$ from ICL. In theory, both $\boldsymbol{p}_m$ and $\boldsymbol{q}_{a\to b}$ can also be seen as class posteriors. Thus it is theoretically reasonable to perform mutual mimicry of these contrastive distributions for better representation learning.

Specifically, we utilize Kullback Leibler (KL)-divergence to force each network's contrastive distributions to align corresponding soft labels provided from other networks within the cohort. This paper focuses on mutually mimicking two types of contrastive distributions from VCL and ICL:

Soft Vanilla Contrastive Learning (Soft VCL). For refining $\boldsymbol{p}_m$ from f_m, the soft pseudo labels are peer contrastive distributions $\{\boldsymbol{p}_l\}_{l=1,l\neq m}^{l=M}$ generated from $\{f_l\}_{l=1,l\neq m}^{l=M}$, respectively. We use $\mathbf{KL}$ divergence to force $\boldsymbol{p}_m$ to align them. For applying soft VCL to the cohort of $\{f_m\}_{m=1}^M$, the overall loss can be formulated as:

$$\mathcal{L}_{1\sim M}^{Soft\_VCL} = \sum_{m=1}^{M} \sum_{l=1,l\neq m}^{M} \mathbf{KL}(\boldsymbol{p}_l \parallel \boldsymbol{p}_m). \quad (6)$$

Here, $\boldsymbol{p}_l$ is the soft label detached from gradient back-propagation for stability.

Soft Interactive Contrastive Learning (Soft ICL). Given two networks f_a and f_b, we can derive interactive contrastive distributions $\boldsymbol{q}_{a\to b}$ and $\boldsymbol{q}_{b\to a}$ using ICL. It makes sense to force the consistency between $\boldsymbol{q}_{a\to b}$ and $\boldsymbol{q}_{b\to a}$ for mutual calibration by Soft ICL. When extending to $\{f_m\}_{m=1}^M$, we perform Soft ICL in every two of M networks, leading to the overall loss as:

$$\mathcal{L}_{1\sim M}^{Soft\_ICL} = \sum_{a=1}^{M} \sum_{b=1,b\neq a}^{M} \mathbf{KL}(\boldsymbol{q}_{b\to a} \parallel \boldsymbol{q}_{a\to b}). \quad (7)$$

Here, $\boldsymbol{q}_{b\to a}$ is the soft label detached from gradient back-propagation for stability.

Discussion with Soft VCL and Soft ICL. We remark that using a vanilla contrastive distribution $\boldsymbol{p}$ as a soft label has been explored by some previous works (Ge, Chen, and Li 2020; Fang et al. 2021). These works often construct contrastive relationships using embeddings from the same network, as illustrated in Eq.(1). In contrast, we propose an interactive contrastive distribution $\boldsymbol{q}$ to perform Soft ICL. Intuitively, $\boldsymbol{q}$ aggregates cross-network embeddings to model the soft label, which is more informative than $\boldsymbol{p}$ constructed from a single embedding space. Moreover, refining a better $\boldsymbol{q}$ may decrease $\mathcal{L}_{a\to b}^{ICL}$, further maximizing the lower bound on the mutual information $I(\boldsymbol{v}_a^0, \boldsymbol{v}_b^1)$ between f_a and f_b. Compared with soft VCL, soft ICL can facilitate more adequate interactions among multiple networks. Empirically, we found soft ICL excavates better performance gains by taking full advantage of collaborative contrastive learning.

Overall loss of MCL. To take full advantage of collaborative learning, we summarize all contrastive loss terms as the overall loss for MCL among a cohort of M networks:

$$\begin{aligned}\mathcal{L}_{1\sim M}^{MCL} = &\alpha\mathcal{L}_{1\sim M}^{VCL} + \beta\mathcal{L}_{1\sim M}^{ICL} \\ &+ \gamma\mathcal{L}_{1\sim M}^{Soft\_VCL} + \lambda\mathcal{L}_{1\sim M}^{Soft\_ICL},\end{aligned} \quad (8)$$

where α, β, γ and λ are weight coefficients. We set $\alpha = \beta = 0.1$ in supervised learning and $\alpha = \beta = 1$ in self-supervised learning. Moreover, we set $\gamma = \lambda = 1$ for KL-divergence losses.

Apply MCL to Supervised Learning

For the small-scale dataset like CIFAR-100, we create a class-aware sampler to derive contrastive samples from the mini-batch. The mini-batch with a batch size of B consists of $B/2$ classes. Each class has two samples, and others from

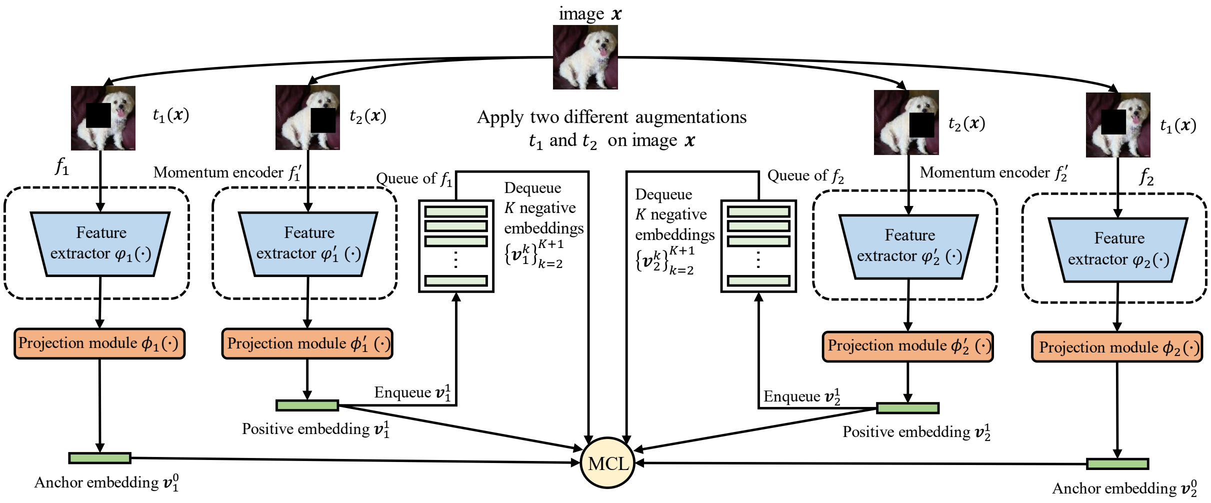


image $\boldsymbol{x}$
$t_1(\boldsymbol{x})$
$t_2(\boldsymbol{x})$
Apply two different augmentations
t_1 and t_2 on image $\boldsymbol{x}$
$t_2(\boldsymbol{x})$
$t_1(\boldsymbol{x})$
f_1
Momentum encoder f_1'
Queue of f_1
Dequeue
K negative
embeddings
$\{\boldsymbol{v}_1^k\}_{k=2}^{K+1}$
Dequeue
K negative
embeddings
$\{\boldsymbol{v}_2^k\}_{k=2}^{K+1}$
Queue of f_2
Momentum encoder f_2'
f_2
Feature extractor $\varphi_1(\cdot)$
Feature extractor $\varphi_1'(\cdot)$
Feature extractor $\varphi_2'(\cdot)$
Feature extractor $\varphi_2(\cdot)$
Projection module $\phi_1(\cdot)$
Projection module $\phi_1'(\cdot)$
Projection module $\phi_2'(\cdot)$
Projection module $\phi_2(\cdot)$
Enqueue $\boldsymbol{v}_1^1$
Enqueue $\boldsymbol{v}_2^1$
Positive embedding $\boldsymbol{v}_1^1$
Positive embedding $\boldsymbol{v}_2^1$
MCL
Anchor embedding $\boldsymbol{v}_1^0$
Anchor embedding $\boldsymbol{v}_2^0$

Figure 2: Overview of incorporate MCL with MoCo (He et al. 2020) for self-supervised learning.

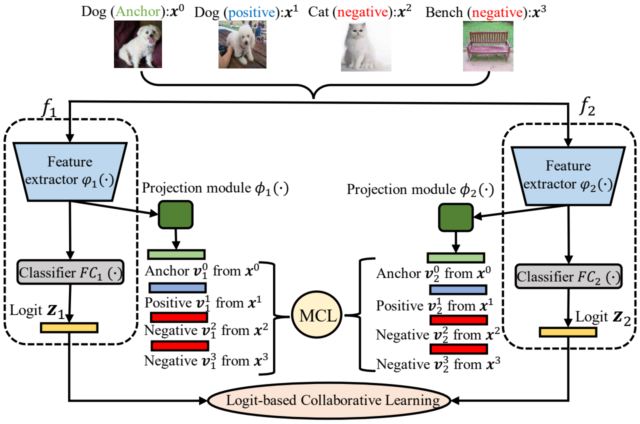

Dog (Anchor):$\boldsymbol{x}^0$
Dog (positive):$\boldsymbol{x}^1$
Cat (negative):$\boldsymbol{x}^2$
Bench (negative):$\boldsymbol{x}^3$
f_1
f_2
Feature extractor $\varphi_1(\cdot)$
Projection module $\phi_1(\cdot)$
Projection module $\phi_2(\cdot)$
Feature extractor $\varphi_2(\cdot)$
Classifier $FC_1(\cdot)$
Classifier $FC_2(\cdot)$
Logit $\boldsymbol{z}_1$
Logit $\boldsymbol{z}_2$
Anchor $\boldsymbol{v}_1^0$ from $\boldsymbol{x}^0$
Positive $\boldsymbol{v}_1^1$ from $\boldsymbol{x}^1$
Negative $\boldsymbol{v}_1^2$ from $\boldsymbol{x}^2$
Negative $\boldsymbol{v}_1^3$ from $\boldsymbol{x}^3$
MCL
Anchor $\boldsymbol{v}_2^0$ from $\boldsymbol{x}^0$
Positive $\boldsymbol{v}_2^1$ from $\boldsymbol{x}^1$
Negative $\boldsymbol{v}_2^2$ from $\boldsymbol{x}^2$
Negative $\boldsymbol{v}_2^3$ from $\boldsymbol{x}^3$
Logit-based Collaborative Learning

Figure 3: Overview of MCL for supervised learning.

different classes are negative samples. We regard each sample as an anchor instance and others as contrastive instances within the current mini-batch. For the large-scale dataset like ImageNet, we create an online memory bank (Wu et al. 2018) to store massive embeddings since the batch size limits the number of available contrastive samples.

For supervised learning, we also perform the conventional sample-independent logit-based learning. For M networks $\{f_m\}_{m=1}^M$ with the input $\boldsymbol{x}$, their generated logit vectors are $\{\boldsymbol{z}_m\}_{m=1}^M$. Each network is supervised by a cross-entropy loss $\mathcal{L}_{ce}$ between the predictive probability distribution and the ground-truth label y. The total loss is: $\mathcal{L}_{1\sim M}^{vanilla}=\sum_{m=1}^M\mathcal{L}_{ce}(softmax(\boldsymbol{z}_m),y)$.

We summarize the logit-based classification loss and embedding-based MCL as the overall loss for collaborative learning. The overall loss is $\mathcal{L}_{1\sim M}^{sup}$:

$$\mathcal{L}_{1\sim M}^{sup}=\mathcal{L}_{1\sim M}^{vanilla}+\mathcal{L}_{1\sim M}^{MCL} \tag{9}$$

We illustrate the overview of collaborative learning under the supervised scenario in Fig. 3.

MCL on Self-Supervised Learning

When MCL is used for self-supervised learning, a positive pair includes two copies from different augmentations applied on the same sample, while negative pairs are often constructed from different samples. Because recent contrastive-based self-supervised learning often uses an InfoNCE loss, *i.e.* the form of Eq.(1), MCL can be readily incorporated with those great works, *e.g.* MoCo (He et al. 2020). As shown in Fig. 2, we illustrate the overview of how to incorporate MCL with MoCo for learning visual representation among a cohort of models. Because self-supervised learning often needs a large number of negative samples, MoCo constructs a momentum encoder and a queue for providing contrastive embeddings. Self-supervised learning only involves feature embedding-based learning so that the overall loss is formulated as the MCL loss.

Experiments

Supervised Image Classification

Datasets. We use CIFAR-100 (Krizhevsky, Hinton et al. 2009) and ImageNet (Deng et al. 2009) datasets for image classification, following the standard data augmentation and preprocessing pipeline (Huang et al. 2017).

Hyper-parameters settings. Following SimCLR (Chen et al. 2020b), we use $\tau=0.1$ on CIFAR and $\tau=0.07$ on ImageNet for similarity calibration of $\mathcal{L}^{VCL}$ and $\mathcal{L}^{ICL}$. The contrastive embedding size d is 128. In soft losses of $\mathcal{L}^{Soft\_VCL}$ and $\mathcal{L}^{Soft\_ICL}$, we utilize $\tau=0.1\times3=0.3$ on CIFAR and $\tau=0.07\times3=0.21$ on ImageNet to smooth similarity distributions. For CIFAR-100, we use $K=126$ as the number of negative samples due to a batch size of 128. For ImageNet, we retrieve one positive and $K=8192$ negative embeddings from the memory bank.

Training settings. For CIFAR-100, all networks are trained by SGD with a momentum of 0.9, a batch size of 128 and a weight decay of 5×10^{-4}. We use a cosine learning rate that starts from 0.1 and gradually decreases to 0 throughout

Network	Baseline	DML	CL	ONE	OKDDip	MVCL	MCL(×4)+Logit	Gain (↑)
WRN-16-2	$72.55_{\pm0.24}$	$75.04_{\pm0.22}$	$74.18_{\pm0.34}$	$74.04_{\pm0.19}$	$74.99_{\pm0.45}$	$\underline{75.76}_{\pm0.21}$	$\mathbf{76.34}_{\pm0.22}$	0.58
WRN-40-2	$76.89_{\pm0.29}$	$78.45_{\pm0.42}$	$78.64_{\pm0.31}$	$79.05_{\pm0.22}$	$\underline{79.21}_{\pm0.06}$	$79.16_{\pm0.36}$	$\mathbf{80.02}_{\pm0.45}$	0.81
WRN-28-4	$79.17_{\pm0.29}$	$80.54_{\pm0.38}$	$80.83_{\pm0.27}$	$80.58_{\pm0.17}$	$80.47_{\pm0.27}$	$\underline{81.16}_{\pm0.36}$	$\mathbf{81.68}_{\pm0.31}$	0.52
ShuffleNetV2 1×	$70.93_{\pm0.24}$	$75.35_{\pm0.30}$	$\underline{75.94}_{\pm0.25}$	$75.74_{\pm0.33}$	$75.24_{\pm0.30}$	$75.88_{\pm0.13}$	$\mathbf{77.02}_{\pm0.32}$	1.08
HCGNet-A2	$79.00_{\pm0.41}$	$\underline{82.10}_{\pm0.29}$	$81.94_{\pm0.11}$	$80.64_{\pm0.20}$	$80.11_{\pm0.19}$	$82.04_{\pm0.15}$	$\mathbf{82.47}_{\pm0.20}$	0.37

Table 1: Top-1 accuracy (%) of jointly training *four networks* with the same architecture on CIFAR-100. The bold number represents the best result among various methods. 'Gain' indicates the accuracy improvement of MCL upon the second-best result.

Network	Baseline	DML	CL	ONE	OKDDip	PCL	MVCL	MCL(×3)+Logit	Gain (↑)
ResNet-18	69.76	$69.82_{\pm0.08}$	$70.04_{\pm0.05}$	$70.18_{\pm0.13}$	$69.93_{\pm0.06}$	$70.42_{\pm0.13}$	$\underline{70.46}_{\pm0.09}$	$\mathbf{70.82}_{\pm0.06}$	0.36

Table 2: Top-1 accuracy (%) of jointly training *three networks* with the same architecture on ImageNet. Part of compared results are obtained from PCL (Wu and Gong 2021). 'Logit' represents logit-based collaborative learning (Yang, An, and Xu 2021).

Network	Baseline	MCL(×2)	Gain (↑)	MCL(×4)	Gain (↑)
ResNet-32	$70.91_{\pm0.14}$	$72.96_{\pm0.28}$	2.05	$74.04_{\pm0.07}$	3.13
ResNet-56	$73.15_{\pm0.23}$	$74.48_{\pm0.23}$	1.33	$75.74_{\pm0.16}$	2.59
ResNet-110	$75.29_{\pm0.16}$	$77.12_{\pm0.20}$	1.83	$78.82_{\pm0.14}$	3.53
WRN-16-2	$72.55_{\pm0.24}$	$74.56_{\pm0.11}$	2.01	$75.79_{\pm0.07}$	3.24
WRN-40-2	$76.89_{\pm0.29}$	$77.51_{\pm0.42}$	0.62	$78.84_{\pm0.22}$	1.95
HCGNet-A1	$77.42_{\pm0.16}$	$78.62_{\pm0.26}$	1.20	$79.50_{\pm0.15}$	2.08
ShuffleNetV2 0.5×	$67.39_{\pm0.35}$	$69.55_{\pm0.22}$	2.16	$70.92_{\pm0.28}$	3.53
ShuffleNetV2 1×	$70.93_{\pm0.24}$	$73.26_{\pm0.18}$	2.33	$75.18_{\pm0.25}$	4.25

Table 3: Top-1 accuracy (%) of jointly training *two or four networks* with the same architecture on CIFAR-100. $\times M$ indicates the cohort has M networks for MCL. 'Gain' indicates the accuracy improvement of MCL upon the Baseline.

Method	ResNet-18		ResNet-34	
	ImageNet	Pascal VOC	ImageNet	Pascal VOC
Baseline	69.76	76.18	73.30	79.81
MCL (×2)	70.32 (↑ 0.56)	77.20 (↑ 1.02)	74.13 (↑ 0.83)	80.37 (↑ 0.56)
MCL (×4)	**70.77** (↑ 1.01)	**77.68** (↑ 1.50)	**74.34** (↑ 1.04)	**80.81** (↑ 1.00)

Table 4: Top-1 classification accuracy (%) on ImageNet by jointly training *two or four networks* and mAP(%) of downstream transfer learning to object detection on Pascal VOC over Faster-RCNN (Ren et al. 2016) framework. The number in brackets is the gain upon Baseline.

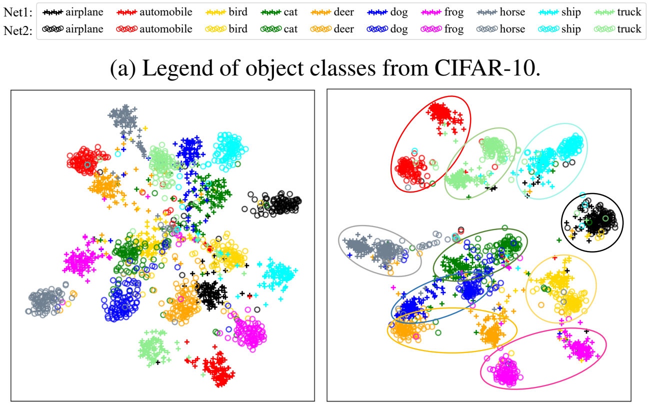
the 300 epochs. For ImageNet, all networks are trained by SGD with a momentum of 0.9, a batch size of 256 and a weight decay of 1×10^{-4}. The initial learning rate starts at 0.1 and is decayed by a factor of 10 at 30 and 60 epochs within the total 90 epochs. We conduct all experiments with the same training settings and report the mean result over three runs for a fair comparison.

Apply MCL upon baseline on CIFAR-100. As shown in Table 3, we first investigate the efficacy of MCL upon the conventional supervised training. We apply widely used ResNets (He et al. 2016), WRNs (Zagoruyko S 2016), HCGNets (Yang et al. 2020) and ShuffleNetV2 (Ma et al. 2018) as the backbone networks to evaluate the performance. All results are achieved from jointly training two networks by MCL(×2) or four networks by MCL(×4) with the same architecture. We observe that our MCL(×2) leads to an average improvement of 1.69% across various architectures upon the independent training for an individual network. The results indicate that MCL can help each network learn better representations effectively. When extending MCL(×2) to MCL(×4), the accuracy gains get more significant. MCL(×4) further advances an average improvement of 3.04% upon baseline. These results verify our claim that more networks in the cohort can capture richer contrastive knowledge, conducive to representation learning.

Training complexity on CIFAR-100. We examine training costs introduced by MCL. For independently training two networks with the same architecture by the conventional cross-entropy loss, the training time and GPU memory are 2× than one network. MCL needs to compute similarity distributions with contrastive embeddings. For supervised contrastive learning on MCL(×2), we use 128-d embedding size, 1 positive and 126 negative embeddings. Extra computation is $4 \times (1 + 126) \times 128 \approx 0.07$M FLOPs for each sample, where 4 represents 2 VCL and 2 ICL distributions. Given two ResNet-110 with 335M FLOPs for an example, applying MCL only introduces an extra 0.02% computation. Since MCL derives contrastive embeddings in mini-batch, we did not find a distinct change of GPU memory cost.

Apply MCL upon baseline on ImageNet. Extensive experiments on more challenging ImageNet further show the scalability of MCL for representation learning to the large-scale dataset. As shown in Table 4, MCL leads to consistent performance improvements over top-1 and top-5 accuracy.

Transferring features to object detection. We use pretrained ResNet-18 on ImageNet as the backbone over Faster-RCNN (Ren et al. 2016) for downstream object detection on Pascal VOC (Everingham et al. 2010). The model is finetuned on `trainval07+12` and evaluated on `test2007` using mAP. The fine-tuning strategy follows the original implementation (Ren et al. 2016). As shown in Table 4, using MCL for training feature extractors of ResNet-18 and ResNet-34 on ImageNet achieves significant mAP gains consistently for downstream detection. The results demonstrate the efficacy of MCL for learning better representations to downstream semantic recognition tasks.



(a) Legend of object classes from CIFAR-10.

(b) Independent training.

(c) Our proposed MCL.

Figure 4: T-SNE visualization of embedding spaces for two ResNet-32 (Net1 and Net2) with independent training (*left*) and our MCL (*right*) on CIFAR-10 dataset. The clusters in the same circle are from the same class.

Network	MoCo	MCL(×2)	MoCoV2	MCL(×2)
ResNet-18	$47.45_{\pm 0.11}$	$48.04_{\pm 0.13}$	$52.30_{\pm 0.09}$	$52.76_{\pm 0.06}$

Table 5: Top-1 accuracy (%) for self-supervised contrastive learning on ImageNet.

Comparison with SOTAs. We compare MCL with recent collaborative learning methods, including DML (Zhang et al. 2018), CL (Song and Chai 2018), ONE (Zhu, Gong et al. 2018), OKDDip (Chen et al. 2020a), PCL (Wu and Gong 2021) and MVCL (Yang, An, and Xu 2021). To maximize performance gains, we also incorporate MCL with logit-based collaborative learning (Yang, An, and Xu 2021) to distill class posterior information. As shown in Table 1 and 2, our MCL achieves the best performance gains against prior works across various networks. It surpasses the previous SOTA MVCL by an average margin of 0.67% on CIFAR-100 and a margin of 0.36% over ResNet-18 on ImageNet. Moreover, it is hard to say which is the second-best method since different methods are superior for various architectures or datasets. These results demonstrate that exploring contrastive representation may be an effective way for collaborative learning beyond class posterior.

Apply MCL to Self-Supervised Learning

We incorporate MCL with recent self-supervised contrastive learning methods of MoCo (He et al. 2020) and MoCoV2 (Chen et al. 2020c). We follow the standard experimental settings and linear classification protocol (He et al. 2020). As illustrated in Fig. 2, we use two networks of f_1 and f_2 with two peer momentum encoders of $f_1^{'}$ and $f_2^{'}$ respectively. As shown in Table 5, MCL improves popular MoCo and MoCoV2 with 0.59% and 0.46% accuracy improvements on ImageNet, respectively. The results indicate that MCL can help these methods to learn better self-supervised feature representations.

Loss	Baseline	MCL(×4)				
$\mathcal{L}^{VCL}$	-	✓	✓	-	-	✓
$\mathcal{L}^{Soft\_VCL}$	-	-	✓	-	-	✓
$\mathcal{L}^{ICL}$	-	-	-	✓	✓	✓
$\mathcal{L}^{Soft\_ICL}$	-	-	-	-	✓	✓
ResNet-32	$70.91_{\pm 0.14}$	$71.57_{\pm 0.09}$	$73.06_{\pm 0.15}$	$71.92_{\pm 0.19}$	$73.68_{\pm 0.13}$	**74.04**$_{\pm 0.07}$
WRN-16-2	$72.55_{\pm 0.24}$	$73.49_{\pm 0.26}$	$74.55_{\pm 0.11}$	$73.89_{\pm 0.16}$	$75.20_{\pm 0.08}$	**75.79**$_{\pm 0.07}$

Table 6: Ablation study of loss terms over MCL(×4) on CIFAR-100.

M	2	3	4	5	6
ResNet-32	$72.96_{\pm 0.28}$	$73.74_{\pm 0.23}$	$74.04_{\pm 0.07}$	**74.10**$_{\pm 0.10}$	$73.98_{\pm 0.09}$
WRN-16-2	$74.56_{\pm 0.11}$	$75.32_{\pm 0.30}$	$75.79_{\pm 0.07}$	**75.86**$_{\pm 0.25}$	$75.78_{\pm 0.24}$

Table 7: Ablation study of the number of networks M for MCL on CIFAR-100.

Ablation Study and Analysis

Does MCL make networks more similar? With the mutual mimicry by MCL, one may ask if output embeddings of different networks in the cohort would get more similar. To answer this question, we visualize the learned embedding spaces of two ResNet-32 with independent training and MCL, as shown in Fig. 4. We observe that two networks trained with MCL indeed show more similar feature distributions compared with the baseline. This observation reveals that various networks using MCL can learn more common knowledge from others. Moreover, compared with the independent training, MCL can enable each network to learn a more discriminative embedding space, which benefits the downstream classification performance.

Ablation study of loss terms in MCL. As shown in Table 6, we can observe that each loss term is conducive to the performance gain. Moreover, $\mathcal{L}^{ICL}+\mathcal{L}^{Soft\_ICL}$ outperforms the counterpart of $\mathcal{L}^{VCL}+\mathcal{L}^{Soft\_VCL}$ with an average accuracy gain of 0.64%. The results verify our claim that ICL and its soft labels are more crucial than the conventional VCL and its soft labels. This is because ICL is more informative than VCL by aggregating cross-network embeddings. Finally, summarizing VCL and ICL into a unified MCL framework can maximize the performance gain for collaborative representation learning.

Impact of the number of networks M. It is interesting to examine performance gains as the number of networks for MCL increases. As shown in Table 7, We start from $M=2$ to $M=6$ and find accuracy steadily increases but saturates at $M=5$ on both ResNet-32 and WRN-16-2.

Conclusion

We propose a simple yet effective Mutual Contrastive Learning method for collaboratively training a cohort of models from the perspective of contrastive representation learning. Experimental results show that it can enjoy broad usage for both supervised and self-supervised learning. We hope our work can foster future research to take advantage of collaborative training from multiple networks to enhance supervised or self-supervised representation learning.

References

Chen, D.; Mei, J.-P.; Wang, C.; Feng, Y.; and Chen, C. 2020a. Online Knowledge Distillation with Diverse Peers. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 3430–3437.

Chen, T.; Kornblith, S.; Norouzi, M.; and Hinton, G. 2020b. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, 1597–1607. PMLR.

Chen, X.; Fan, H.; Girshick, R.; and He, K. 2020c. Improved baselines with momentum contrastive learning. *arXiv preprint arXiv:2003.04297*.

Deng, J.; Dong, W.; Socher, R.; Li, L.-J.; Li, K.; and Fei-Fei, L. 2009. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, 248–255. Ieee.

Everingham, M.; Van Gool, L.; Williams, C. K.; Winn, J.; and Zisserman, A. 2010. The pascal visual object classes (voc) challenge. *International journal of computer vision*, 88(2): 303–338.

Fang, Z.; Wang, J.; Wang, L.; Zhang, L.; Yang, Y.; and Liu, Z. 2021. Seed: Self-supervised distillation for visual representation. *ICLR*.

Ge, Y.; Chen, D.; and Li, H. 2020. Mutual mean-teaching: Pseudo label refinery for unsupervised domain adaptation on person re-identification. *ICLR*.

Guo, Q.; Wang, X.; Wu, Y.; Yu, Z.; Liang, D.; Hu, X.; and Luo, P. 2020. Online knowledge distillation via collaborative learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 11020–11029.

Hadsell, R.; Chopra, S.; and LeCun, Y. 2006. Dimensionality reduction by learning an invariant mapping. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, volume 2, 1735–1742.

He, K.; Fan, H.; Wu, Y.; Xie, S.; and Girshick, R. 2020. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 9729–9738.

He, K.; Zhang, X.; Ren, S.; and Sun, J. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 770–778.

Hinton, G.; Vinyals, O.; and Dean, J. 2015. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*.

Huang, G.; Liu, Z.; Van Der Maaten, L.; and Weinberger, K. Q. 2017. Densely connected convolutional networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 4700–4708.

Khosla, P.; Teterwak, P.; Wang, C.; Sarna, A.; Tian, Y.; Isola, P.; Maschinot, A.; Liu, C.; and Krishnan, D. 2020. Supervised Contrastive Learning. *Advances in Neural Information Processing Systems*, 33.

Krizhevsky, A.; Hinton, G.; et al. 2009. Learning multiple layers of features from tiny images. *Technical Report*.

Li, X.; Zhou, Y.; Zhang, Y.; Zhang, A.; Wang, W.; Jiang, N.; Wu, H.; and Wang, W. 2021. Dense semantic contrast for self-supervised visual representation learning. In *Proceedings of the 29th ACM International Conference on Multimedia*, 1368–1376.

Luo, D.; Liu, C.; Zhou, Y.; Yang, D.; Ma, C.; Ye, Q.; and Wang, W. 2020. Video cloze procedure for self-supervised spatio-temporal learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 11701–11708.

Ma, N.; Zhang, X.; Zheng, H.-T.; and Sun, J. 2018. Shufflenet v2: Practical guidelines for efficient cnn architecture design. In *Proceedings of the European conference on computer vision (ECCV)*, 116–131.

Misra, I.; and Maaten, L. v. d. 2020. Self-supervised learning of pretext-invariant representations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 6707–6717.

Noroozi, M.; and Favaro, P. 2016. Unsupervised learning of visual representations by solving jigsaw puzzles. In *European Conference on Computer Vision*, 69–84. Springer.

Oord, A. v. d.; Li, Y.; and Vinyals, O. 2018. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*.

Park, W.; Kim, D.; Lu, Y.; and Cho, M. 2019. Relational knowledge distillation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 3967–3976.

Peng, B.; Jin, X.; Liu, J.; Li, D.; Wu, Y.; Liu, Y.; Zhou, S.; and Zhang, Z. 2019. Correlation congruence for knowledge distillation. In *Proceedings of the IEEE International Conference on Computer Vision*, 5007–5016.

Ren, S.; He, K.; Girshick, R.; and Sun, J. 2016. Faster r-cnn: Towards real-time object detection with region proposal networks. *IEEE transactions on pattern analysis and machine intelligence*, 39(6): 1137–1149.

Schroff, F.; Kalenichenko, D.; and Philbin, J. 2015. Facenet: A unified embedding for face recognition and clustering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 815–823.

Song, G.; and Chai, W. 2018. Collaborative learning for deep neural networks. In *Advances in Neural Information Processing Systems*, 1832–1841.

Tian, Y.; Krishnan, D.; and Isola, P. 2020. Contrastive multiview coding. In *European Conference on Computer Vision*, 776–794.

Wu, G.; and Gong, S. 2021. Peer collaborative learning for online knowledge distillation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 10302–10310.

Wu, Z.; Xiong, Y.; Yu, S. X.; and Lin, D. 2018. Unsupervised feature learning via non-parametric instance discrimination. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 3733–3742.

Xu, Y.; Wang, Q.; An, Z.; Wang, F.; Zhang, L.; Wu, Y.; Dong, F.; Qiu, C.-W.; Liu, X.; Qiu, J.; et al. 2021. Artificial Intelligence: A Powerful Paradigm for Scientific Research. *The Innovation*, 100179.

Yang, C.; An, Z.; Cai, L.; and Xu, Y. 2021a. Hierarchical Self-supervised Augmented Knowledge Distillation. In *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence (IJCAI)*, 1217–1223.

Yang, C.; An, Z.; Cai, L.; and Xu, Y. 2021b. Knowledge Distillation Using Hierarchical Self-Supervision Augmented Distribution. *arXiv preprint arXiv:2109.03075*.

Yang, C.; An, Z.; and Xu, Y. 2021. Multi-View Contrastive Learning for Online Knowledge Distillation. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 3750–3754.

Yang, C.; An, Z.; Zhu, H.; Hu, X.; Zhang, K.; Xu, K.; Li, C.; and Xu, Y. 2020. Gated convolutional networks with hybrid connectivity for image classification. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 12581–12588.

Yao, Y.; Liu, C.; Luo, D.; Zhou, Y.; and Ye, Q. 2020. Video playback rate perception for self-supervised spatio-temporal representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 6548–6557.

Zagoruyko S, K. N. 2016. Wide residual networks. In *Proceedings of the British Machine Vision Conference*.

Zhang, K.; Yao, P.; Wu, R.; Yang, C.; Li, D.; Du, M.; Deng, K.; Liu, R.; and Zheng, T. 2021. Learning Positional Priors for Pretraining 2D Pose Estimators. In *Proceedings of the 2nd International Workshop on Human-centric Multimedia Analysis*, 3–11.

Zhang, Y.; Xiang, T.; Hospedales, T. M.; and Lu, H. 2018. Deep mutual learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 4320–4328.

Zhu, X.; Gong, S.; et al. 2018. Knowledge distillation by on-the-fly native ensemble. In *Advances in neural information processing systems*, 7517–7527.