HNSSL: Hard Negative-Based Self-Supervised Learning

Wentao Zhu
Amazon

Jingya Liu
City College of New York

Yufang Huang
Cornell University

Abstract

Recently, learning from vast unlabeled data, especially self-supervised learning, has been emerging and attracting widespread attention. Self-supervised learning followed by supervised fine-tuning on a few labeled examples can significantly improve label efficiency and outperform standard supervised training using fully annotated data [6]. In this work, we present a novel hard negative-based self-supervised deep learning paradigm, named HNSSL. Specifically, we design a student-teacher network to generate a multi-view of the data for self-supervised learning and integrate an online hard negative pair mining into the training. Then we derive a new triplet-type loss considering both positive sample pairs and online mined hard negative sample pairs. Extensive experiments demonstrate the effectiveness of the proposed method and its components on ILSVRC-2012 based on the same backbone network. Specifically, for the linear evaluation task, the proposed HNSSL with a ResNet-50 encoder achieves the top-1 accuracy of 77.1%, which outperforms its previous counterparts by 2.8%. For the semi-supervised learning task, HNSSL with a ResNet-50 encoder obtains the top-1 accuracy of 73.4%, which outperforms the previous ResNet-50 encoder-based semi-supervised learning results by 4.6% using only 10% labels. For the task of transfer learning with linear evaluation, HNSSL with a ResNet-50 encoder achieves the best accuracy on six of seven widely used transfer learning datasets, which averagely outperforms previous ResNet-50 encoder-based transfer learning results by 2.5%.

1. Introduction

Learning from a large-scale unlabeled dataset has long been a hot topic in the computer vision community because a large number of high-quality labels require laborious and costly annotation for each task and there exists a huge amount of unlabeled data from various data servers and sources. Unsupervised or self-supervised learning can effectively learn a *task-agnostic* representation from vast unlabeled data. The downstream tasks, such as image classification, can be well performed by fine-tuning on a few *task-specific* labels. This strategy has become a main-stream pipeline for the transformer-based self-supervised learning approaches [46]. Recent advanced self-supervised learning achieves promising results and outperforms conventional fully supervised learning methods on the image classification [6].

The main effort of general self-supervised learning approaches mainly focuses on pretext task construction [18]. The pretext task can be designed to be predictive tasks [31], generative tasks [1], contrastive tasks [34, 44], or a combination of them. The supervision signal for the pretext task, *i.e.*, *pseudo label*, typically is yielded from a pretext construction process which generally involves exhausted multi-view construction to model various variations [37]. Through solving the pretext task with a specific objective function, the network learns transferable visual features for various downstream tasks.

The study of conventional self-supervised learning methods mainly involves data-related pretext task designs, such as test time training [53], data augmentation prediction [40, 52, 54], cycle consistency loss [17, 41], masked auto-encoder [13], self-distillation loss [4]. Popular pretext tasks include colorizing gray scale images [50], image inpainting [35], playing image jigsaw puzzle [33], *etc*. For the video-related self-supervised learning approaches, the data-related pretext tasks can be sequence order verification [32], solving sequence sorting [25], predicting the odd or unrelated element [10], classifying clip order [48], *etc*.

The recent tremendous success of self-supervised learning is mainly introduced by advanced learning strategies. The InfoNCE loss is widely adopted for contrastive learning, which maximizes a lower bound of mutual information based on the *pseudo label* in the pretext task [34]. SimCLR employs larger batch sizes, more training steps, and a composition of data augmentations, which matches the performance of a fully supervised ResNet-50 simply by adding one additional linear classifier [5, 15]. Wu *et al*. [47] maintains a large feature memory bank to store training image representation. MoCo builds a large and consistent dictionary through a dynamic queue and a

momentum-updated encoder, which outperforms its supervised pretraining counterpart in the detection and segmentation [14]. SimSiam employs a stop-gradient operation in the Siamese architectures to prevent collapsing solutions of self-supervised learning [8]. SimCLRv2 employs big, *i.e*., deep and wide, networks during pretraining and fine-tuning, and it achieves surprisingly good performance for semi-supervised learning on ImageNet [6, 39]. BYOL trains an online network to predict a target network representation of the same image where the target network is a slow-moving average of the online network [11].

In this work, we propose a novel self-supervised learning paradigm by introducing an effective negative image pair mining in the contrastive learning framework. Specifically, we introduce a student-teacher network into the contrastive learning framework to construct a multi-view representation of data. To effectively learn from unlabeled data in contrastive learning, we further construct the negative image pairs by online hard negative image pair mining. The overall objective function can be derived as a form of triplet-type loss facilitated by the collected positive and negative image pairs. To avoid the collapsing solution and improve the accuracy of self-supervised learning, we block the gradient of the student sub-network in the training inspired by SimSiam [8].

We conduct extensive experiments including linear evaluation, semi-supervised learning, transfer learning, and ablation study to evaluate our method on the ImageNet dataset [39]. The proposed method achieves 77.1% top-1 accuracy using a ResNet-50 encoder for the linear evaluation, which outperforms previous ResNet-50 encoder-based state-of-the-art methods by 2.8%. For the semi-supervised learning task, our method with a ResNet-50 encoder obtains the top-1 accuracy of 73.4%, which outperforms the previous ResNet-50 encoder-based best result by 4.6% using 10% labels. For transfer learning with linear evaluation, our method with a ResNet-50 encoder achieves the best accuracy on six of seven widely used transfer learning datasets, which averagely outperforms the previous ResNet-50 encoder-based best results by 2.5%. More specifically, our major contributions are summarized as follows.

- First, we build a student-teacher network to construct multi-view representations in the contrastive learning framework. The gradient of the student sub-network is blocked to ease the training difficulty and stabilize the training of self-supervised learning.

- Second, we collect hard negative image pairs on-the-fly and add the hard negative image pairs into the training of contrastive self-supervised learning.

- Third, extensive experiments demonstrate that hard negative-based self-supervised learning outperforms previous state-of-the-art self-supervised learning approaches for linear evaluation, semi-supervised learning, and transfer learning based on the same encoder on the ImageNet dataset.

2. Related Work

The mainstream unsupervised or self-supervised learning literature generally involves two aspects: data or feature-related pretext tasks and loss functions [14]. The data or feature-related pretext tasks typically can be specially constructed by the multi-view data or feature generation process [18]. Through solving the pretext task, the deep network of self-supervised learning is expected to learn a good representation for the downstream tasks. Loss objective functions can often improve the performance of self-supervised learning significantly. Our hard negative-based self-supervised learning focuses on the novel loss function based on advanced student-teacher network design. Next, we discuss the related studies with respect to these aspects.

Pretext tasks Without a large scale fully annotated dataset, self-supervised learning can be designed to solve a pretext task where *pseudo labels* are typically generated based on data attributes [18]. Pathak *et al*. [36] use unsupervised motion-based segmentation as the pretext task for transfer learning on object detection. Context-based pixel prediction is used as a pretext task to improve downstream tasks of image classification, detection, and segmentation [35]. DeepCluster uses a standard clustering algorithm, k-means, to generate *pseudo labels* and employs its assignments as supervision to update the weights of the network [2]. Larsson *et al*. [24] first conduct an in-depth analysis of self-supervision via colorization showing that colorization provides a powerful supervisory signal for the ImageNet pretraining. ViLBERT extends the popular BERT model to learn joint visual-linguistic representations [20, 30].

Loss functions Contrastive loss measures the similarity of image pairs in the feature space [12]. In contrastive learning framework, the target can be defined and generated on the fly during training [12]. The recent significant success of self-supervised learning has witnessed the widespread adoption of contrastive learning [16]. Zhuang *et al*. [57] train an embedding function to maximize a metric of local aggregation, causing similar data instances to move together in the embedding space while allowing dissimilar instances to separate. Contrastive multi-view learning trains deep networks by maximizing mutual information between different views of the same scene [44].

Self-supervised student-teacher learning The student-teacher network can be used to generate multi-view representations of unlabeled data. Temporal ensembling maintains an exponential moving average (EMA) prediction as the *pseudo label* for the self-supervised training [23]. In-

stead of averaging label predictions, the mean-teacher uses EMA to update model weights [43]. MoCo further uses momentum to update the encoder for the new keys on-the-fly and maintains a queue of keys in the contrastive learning framework [9, 14]. BYOL maintains a student-teacher network to yield a multi-view of samples in the training [11]. Without negative sample pairs in the training, BYOL uses a large batch size and achieves surprisingly good performance. Momentum teacher performs two independent momentum updates for the teacher's weight and the teacher's batch normalization statistics to maintain a stable training process [27]. Kalantidis *et al*. [19] propose hard negative mixing, which synthesizes hard negatives directly in the embedding space instead.

3. Method

In this section, we describe each component of the proposed hard negative-based self-supervised learning (HNSSL).

3.1. Overall Framework

We employ a student-teacher network to construct two representational views of the sample, as illustrated in Fig. 1. At the top of the student and teacher sub-networks, we construct both the positive sample pairs and negative sample pairs. Specifically, we consider the representations of the same sample from student and teacher sub-networks as the positive pair, and we only retain the most similar pair of two different samples to construct the negative pair, *i.e*., hard negative pair. We block the gradient update of the student sub-network and employ the exponential moving average (EMA) to update its parameters to stabilize the self-supervised training. The problem definition and network configuration including the general loss function are described in the section 3.2. The hard negative pair mining (HNPM) is shown in section 3.3 and the detailed network update rule is in section 3.4. We also analyze the stability of our method and the connection between our method and InfoNCE [34] in section 3.5. The implementation details are provided in section 3.6.

3.2. Student-Teacher Network

Problem definition: Unsupervised or self-supervised learning tries to learn a good representation from a large scale unlabeled dataset $\mathcal{D} = \{I_1, I_2, \cdots, I_N\}$, where each I represents an image. For an image sampled from the dataset $I_i \sim \mathcal{D}$, we can obtain two representational views of I_i by constructing one student sub-network $\mathcal{S}(\cdot; \theta_S)$ and one teacher sub-network $\mathcal{T}(\cdot; \theta_T)$ [42]. To learn various invariants, we employ advanced data augmentation $\mathcal{A}$, including color jittering, horizontal flipping, Gaussian blurring, and random cropping, in the data generation process for the teacher sub-network. We then obtain two views of representations for image I_i as

$$U_i = \mathcal{T}(\mathcal{A}(I_i); \theta_T), \quad U_i' = \mathcal{S}(I_i; \theta_S), \tag{1}$$

where U_i is the representation from the teacher sub-network and U_i' is the representation from the student sub-network.

The self-supervised learning tries to build pretext tasks from these unlabeled data. The generated representation views U_i and U_i' from the student and teacher sub-networks can be considered as a positive pair, which belong to the same cluster. The hard negative-based contrastive self-supervised learning tries to yield compact representations for images of the same cluster by minimizing their normalized L_2 distance in the representational space. The intra-cluster distance can be defined

$$\mathcal{L}_1 = \mathbb{E}_{I_i \sim \mathcal{D}} \left[\left(\frac{U_i}{\|U_i\|_\infty} - \frac{U_i'}{\|U_i'\|_\infty} \right)^2 \right], \tag{2}$$

where images are randomly sampled from the dataset $I_i \sim \mathcal{D}$, $\|\cdot\|_\infty$ is the infinity norm, *i.e*., the maximum of the absolute value of elements in the vector.

3.3. Hard Negative Pair Mining (HNPM)

It is not efficient to train a self-supervised network by solely using positive pairs of samples. Current self-supervised learning uses large batch size [5], memory bank [47], or large dynamic dictionary [14] to achieve promising results. Adding negative image pairs can significantly improve the training efficiency of a self-supervised learning model.

We heuristically construct negative pairs in the self-supervised learning framework by online mining hard negative pairs of images. For two different images I_i and image I_j, we measure the dissimilarity of the two images by the normalized L_2 distance in the representation space

$$\begin{aligned} &U_j = \mathcal{T}(\mathcal{A}(I_j); \theta_T), \\ &\text{DisSim}(U_i', U_j) = \left(\frac{U_i'}{\|U_i'\|_\infty} - \frac{U_j}{\|U_j\|_\infty} \right)^2. \end{aligned} \tag{3}$$

There exist large numbers of negative pairs of samples. Hard samples have been widely proven to improve the performance of a deep learning model [28, 38]. In the self-supervised learning framework, we define the hard negative pairs to be image pairs of small dissimilarity according to Eq. (3). We try to maximize the normalized L_2 distance or dissimilarity of negative image pairs in the latent space. The contrastive loss for negative pairs can be derived

$$\mathcal{L}_2 = -\mathbb{E}_{I_i \sim \mathcal{D}} [\log \big(\sum_{I_j \in \tilde{\mathcal{B}}_i} (\text{DisSim}(U_i', U_j)) \big)], \tag{4}$$

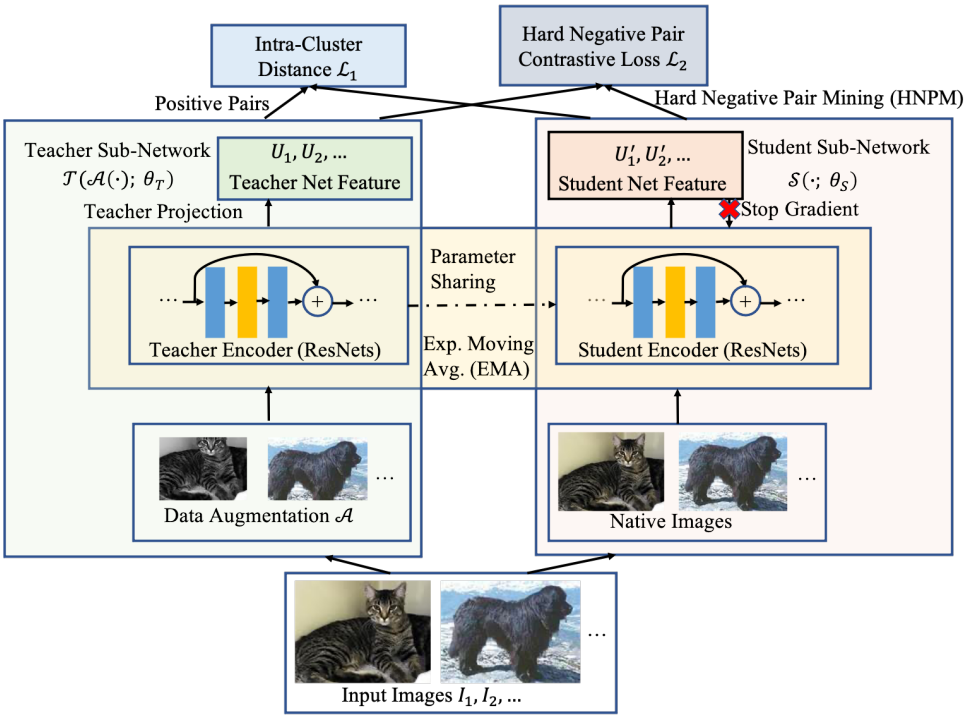

Figure 1. The architecture of hard negative-based self-supervised learning (HNSSL). HNSSL employs triplet-type loss with both positive sample pairs and negative sample pairs. The student and teacher sub-networks yield two representational views of one sample, which forms the positive sample pair in the training. For the negative sample pair, we employ hard negative pair mining (HNPM) to generate negative pairs on-the-fly. We block the gradient update of the student sub-network and employ exponential moving average (EMA) to update its parameters which stabilizes the training and avoids a collapsing solution.

where images are randomly sampled from the dataset $I_i \sim \mathcal{D}$, $\tilde{\mathcal{B}}_i$ is the hard negative sample set of the current batch $\mathcal{B}_i$ for image I_i. The hard negative sample set $\tilde{\mathcal{B}}_i$ can be constructed

$$\tilde{\mathcal{B}}_i = \{I_j | I_j \in \mathcal{B}_i, I_j \neq I_i, \text{DisSim}(U'_i, U_j) \leq 1\}. \quad (5)$$

We construct hard negative pairs on-the-fly in training, which can be used to efficiently train the self-supervised network.

3.4. Network Update

To stabilize the training and avoid a collapsing solution in the self-supervised learning [8], we block the gradient for the student sub-network $\mathcal{S}(\cdot;\theta_S)$. We employ the exponential moving average (EMA) to update the parameters θ_S in the student sub-network [43]

$$\theta_S \leftarrow \tau\theta_S + (1-\tau) \times \theta_T, \quad (6)$$

where τ is a smoothing coefficient to tune the updated strength of the student sub-network.

In the back-propagation, we only use the gradient to update the parameters of the teacher sub-network. The overall loss function can be derived as

$$\mathcal{L}(\theta_T) = \alpha_1\mathcal{L}_1 + \alpha_2\mathcal{L}_2, \quad (7)$$

where $0 < \alpha_1 < 1$ and $0 < \alpha_2 < 1$ are the fixed coefficients to tune the trade-off between the intra-cluster loss and inter-cluster loss. During the back-propagation, we employ gradient clipping to stabilize the training of the teacher sub-network.

3.5. Connection with InfoNCE and Stability

In our method, we employ online hard negative pair mining (HNPM) to add negative image pairs in the training and use a normalized L_2 distance in the loss function. We will demonstrate that minimizing the loss of our method is equivalent to minimizing the InfoNCE loss [34]. To simplify the analysis, we temporarily remove the online hard negative pair mining mechanism in our method in the derivation of connection with InfoNCE.

The InfoNCE loss [34] can be written as

$$\mathcal{L}_{NCE} = -\mathbb{E}_{I_i \sim \mathcal{D}}[\log \frac{f_k(U_i, U'_i)}{\sum_{I_j \in \mathcal{D}} f_k(U_j, U'_i)}]. \quad (8)$$

where U_i, U_i' are calculated from the teacher sub-network and the student sub-network, $f_k(\cdot,\cdot)$ models the mutual information between the encoded representations in the InfoNCE and can use similarity loss as a surrogate loss to approximate the mutual information.

We define the similarity loss as the reciprocal of normalized L_2 distance of the encoded representations. The InfoNCE loss can then be defined as

$$\begin{aligned}\mathcal{L}_{NCE} &\triangleq \mathbb{E}_{I_i\sim\mathcal{D}}[\log \frac{\text{DisSim}(U_i,U_i')}{\sum_{I_j\in\mathcal{D}}\text{DisSim}(U_j,U_i')}] \\ &= \mathbb{E}_{I_i\sim\mathcal{D}}[\log(\frac{U_i}{\|U_i\|_\infty}-\frac{U_i'}{\|U_i'\|_\infty})^2] \\ &\quad - \mathbb{E}_{I_i\sim\mathcal{D}}[\log\big(\sum_{I_j\in\mathcal{D}}(\frac{U_j}{\|U_j\|_\infty}-\frac{U_i'}{\|U_i'\|_\infty})^2)].\end{aligned} \tag{9}$$

The second part of the derived loss in Eq. (9) is the same as our negative pair loss in Eq. (4) if we temporarily neglect our hard negative sample pair mining for each batch. Minimizing the first part of Eq. (9) is equivalent with minimizing $\mathbb{E}_{I_i\sim\mathcal{D}}[(\frac{U_i}{\|U_i\|_\infty}-\frac{U_i'}{\|U_i'\|_\infty})^2]$, which is the positive pair loss in the Eq. (2). From the above derivation, we conclude, with the proper relaxation and assumption, minimizing our loss is equivalent to minimizing the InfoNCE loss.

Next, we demonstrate that hard negative pair mining (HNPM) leads to stable training. Without the trade-off factors α_1 and α_2, the loss can be written as

$$\begin{aligned}\mathcal{L} &= \mathbb{E}_{I_i\sim\mathcal{D}}[(\frac{U_i}{\|U_i\|_\infty}-\frac{U_i'}{\|U_i'\|_\infty})^2] \\ &\quad - \mathbb{E}_{I_i\sim\mathcal{D}}[\log\big(\sum_{I_j\in\tilde{\mathcal{B}}_i}(\frac{U_j}{\|U_j\|_\infty}-\frac{U_i'}{\|U_i'\|_\infty})^2)].\end{aligned} \tag{10}$$

Without loss of generality, we remove the normalization constraint and denote $\frac{U_i}{\|U_i\|_\infty}$ as U_i.

$$\mathcal{L} = \mathbb{E}_{I_i\sim\mathcal{D}}\big[(U_i-U_i')^2 - \log\big(\sum_{I_j\in\tilde{\mathcal{B}}_i}(U_j-U_i')^2\big)\big]. \tag{11}$$

The hard negative pair mining (HNPM) always explores negative pairs with L_2 distance smaller than 1, which guarantees $(U_j-U_i')^2$ is bounded to be smaller than 1. We use M to denote the upper bound of negative pair loss.

$$|\mathcal{L}| \le \mathbb{E}_{I_i\sim\mathcal{D}}\big[(U_i-U_i')^2\big] + M. \tag{12}$$

Next, we can further prove that Eq. (12) can be optimized stably, and the first part of Eq. (12), *i.e.*, the loss of positive pairs, can be decreased consecutively by escaping undesirable equilibria. If the model stacks into an undesirable equilibrium solution, the feature representation of the teacher sub-network can be denoted as $\mathbb{E}[U_i'|U_i]$ from the update rule in Eq. (6). The loss of positive pairs $\mathcal{L}_P$ can be derived as

$$\begin{aligned}\mathcal{L}_P &= \mathbb{E}_{I_i\sim\mathcal{D}}\big[(U_i-U_i')^2\big] \\ &= \mathbb{E}_{I_i\sim\mathcal{D}}\big[(\mathbb{E}[U_i'|U_i]-U_i')^2\big] = \mathbb{E}_{I_i\sim\mathcal{D}}[Var(U_i'|U_i)].\end{aligned} \tag{13}$$

Let Z denote an additional variability induced by stochasticities in the training dynamics. We always have a solution leading to a lower loss during the training, which escapes the current equilibrium, because

$$Var(U_i'|U_i,Z) \le Var(U_i'|U_i). \tag{14}$$

From the above derivation, the learning is stable with the benefit of hard negative pair mining and student sub-network updating rule.

3.6. Implementation Details

Because of our advanced learning strategy, we do not use any pre-trained model as the backbone of our implementation. To generate multi-view representations, we employ data augmentation to model various variations in different views.

We use residual networks as the student sub-network $\mathcal{S}(\cdot,\theta_S)$ and teacher sub-network $\mathcal{T}(\cdot,\theta_T)$. The two coefficients of the loss in Eq. (7), α_1 is set to 0.8 and α_2 is set to 0.1. We employ the gradient clipping strategy in the back-propagation where we set the maximum norm of gradient clipping as 1.0. The Adam optimizer [21] is used to minimize the loss in Eq. (7). The batch size is 160. The learning rate is set as 0.1 and we use a cosine annealing schedule [29] for the learning rate with the maximum number of iterations as 100. The smoothing coefficient τ in the update of the student sub-network in Eq. (6) is set as 0.5, which will be discussed in section 4.4.

We employ data augmentation for teacher sub-network on-the-fly during training. We first apply color jittering with the brightness of 0.8, the contrast of 0.8, a saturation of 0.8, and a hue of 0.2 to random 80% training images in each batch. Then we convert random 20% images to grayscale and horizontally flip 50% images. After that, we smooth random 10% images with a random Gaussian kernel of size 3×3 and standard deviation of 1.5×1.5. Finally, we crop each image with a random crop size of scale range $[0.8, 1.0]$. We use the mean of $[0.485, 0.456, 0.406]$ and the standard deviation of $[0.229, 0.224, 0.225]$ to normalize RGB channels of each image.

4. Experiments

We conduct experiments to validate the performance of the proposed method on the ILSVRC-2012 dataset [39]. We compare our method with other self-supervised learning (SSL) approaches with the same encoder based on linear

Method	Top-1	Top-5
CPCv2 [16]	63.8	85.3
CMC [44]	66.2	87.0
SimCLR [5]	69.3	89.0
MoCov2 [7]	71.1	N/A
SimCLRv2 [6]	71.7	N/A
InfoMin Aug. [45]	73.0	91.1
BYOL [11]	74.3	91.6
SwAV [3]	71.8	N/A
SimSiam [8]	71.3	N/A
Ours	**77.1**	**93.7**

Table 1. The accuracy comparison of self-supervised learning (SSL) approaches with the ResNet-50 encoder based on linear evaluation on the ImageNet dataset. The boldface denotes the best accuracy.

Method	Dep.	Wid.	Top-1	Top-5
CMC [44]	50	2×	70.6	89.7
SimCLRv2 [6]	50	2×	75.6	N/A
BYOL [11]	50	2×	77.4	93.6
Ours	50	2×	**79.4**	**94.5**
SimCL [5]R	50	4×	76.5	93.2
BYOL [11]	50	4×	78.6	94.2
Ours	50	4×	**80.3**	**95.1**
BYOL [11]	200	2×	79.6	94.8
Ours	200	2×	**81.9**	**96.4**

Table 2. The accuracy (%) comparison of SSL methods with other ResNet encoders based on the linear evaluation on the ImageNet dataset.

evaluation, semi-supervised learning, and transfer learning. We also conduct a systematical ablation study to validate each component of our method.

4.1. Linear Evaluation

The linear evaluation can be used to evaluate the accuracy of self-supervised learning (SSL) by freezing the SSL model and training a separate linear classifier after the SSL model [11, 22, 50]. We compare our method with previous state-of-the-art approaches with the ResNet-50 encoder and other ResNet encoders on ImageNet in Table 1 and Table 2, respectively. The top-1 and top-5 accuracy are listed. With the standard ResNet-50 encoder [15], our method obtains 77.1% top-1 accuracy and 93.7% top-5 accuracy, which outperform previous ResNet-50-based state-of-the-art top-1 and top-5 results by 2.8% and 2.1%, respectively. Most surprisingly, our method achieves 0.6% better accuracy than the accuracy, 76.5%, of the supervised baseline from SimCLR [5].

Table 2 reports the accuracy of self-supervised learning methods using deeper and wider ResNet encoders based on the linear evaluation. Our method with ResNet-200 (2×) obtains 81.9% top-1 and 96.4% top-5 accuracy which increases previous ResNet-based best top-1 and top-5 accuracy by 2.3% and 1.6%, respectively. With ResNet-50 (2×) and ResNet-50 (4×) encoders, our method also achieves better accuracy than those of CMC [44], SimCLRv2 [6] and BYOL [11] with the same encoder.

4.2. Semi-Supervised Learning

Semi-supervised learning can also be used to evaluate the accuracy of self-supervised learning (SSL) by fine-tuning representation with a small subset of the training set [11]. In this experiment, we use the fixed data splits of 1% and 10% of the training set in ImageNet, which are the same as [11]. We also use the top-1 and top-5 accuracy as the evaluation metric for semi-supervised learning. The comparison using the ResNet-50 encoder and deeper and wider ResNet encoders are listed in Table 3 and Table 4, respectively. Our method achieves 80.2% top-5 accuracy based on a ResNet-50 encoder which improves the previous ResNet-based best result by 1.8% using only 1% training labels in Table 3. Using 10% training labels, our method achieves 73.4% and 92.5% for the top-1 and top-5 accuracy, which improves the previous ResNet-based best top-1 and top-5 accuracy by 4.6% and 3.5%.

The result with ResNet of various depths, widths, and selective kernel convolution [26] configurations are listed in Table 4. Our method achieves the best top-1 and top-5 accuracy for all the experimental configurations. Specifically, based on ResNet-50 (2×) encoder, our method achieves 65.7% and 78.6% top-1 accuracy using 1% training labels and 10% training labels, which improves the previous best top-1 accuracy by 3.5% and 5.1%. Based on ResNet-200 (2×), our method obtains 76.5% and 80.7% top-1 accuracy using 1% training labels and 10% training labels, which improves the accuracy of BYOL [11] by 5.3% and 3.0%.

4.3. Transfer Learning

Transfer learning is another widely used task to evaluate the accuracy of self-supervised learning (SSL) methods. Transfer learning can be used to evaluate the generalization ability of the learned SSL model. Practically, both linear evaluation, *i.e.*, only training the last classification layer, and fine-tuning the whole network based on the target dataset can be employed for the evaluation of transfer learning. The comparison of transfer learning with linear evaluation and fine-tuning are listed in Table 5 and Table 6.

For the linear evaluation of the transfer learning task, our method achieves better accuracy than previous ResNet-based state-of-the-art approaches on six out of seven widely used transfer learning datasets in Table 5. We provide the

Method	Top-1 (1%)	Top-5 (1%)	Top-1 (10%)	Top-5 (10%)
SimCLR [5]	48.3	75.5	65.6	87.8
SimCLRv2 [6]	**57.9**	N/A	68.4	N/A
BYOL [11]	53.2	78.4	68.8	89.0
Ours	56.7	**80.2** (1.8↑)	**73.4** (4.6↑)	**92.5** (3.5↑)

Table 3. The accuracy (%) comparison of SSL methods with the ResNet-50 encoder based on semi-supervised learning on ImageNet dataset.

Method	Dep.	Wid.	SK	Para.	Top-1	Top-5	Top-1 (10%)	Top-5 (10%)
SimCLR [5]	50	2×	✗	94M	58.5	83.0	71.7	91.2
BYOL [11]	50	2×	✗	94M	62.2	84.1	73.5	91.7
Ours	50	2×	✗	94M	**65.7**	**86.2**	**78.6** (5.1↑)	**93.2** (1.5↑)
SimCLR [5]	50	4×	✗	375M	63.0	85.8	74.4	92.6
BYOL [11]	50	4×	✗	375M	69.1	87.9	75.7	92.5
Ours	50	4×	✗	375M	**70.3**	**89.9**	**78.9** (3.2↑)	**95.5** (2.9↑)
BYOL [11]	200	2×	✗	250M	71.2	87.9	77.7	92.5
Ours	200	2×	✗	250M	**76.5**	**90.3**	**80.7** (3.0↑)	**95.4** (2.9↑)
SimCLRv2 distilled [6]	50	1×	✗	N/A	73.9	91.5	77.5	93.4
SimCLRv2 distilled [6]	50	2×	✓	N/A	75.9	93.0	80.2	95.0
SimCLRv2 self-distilled [6]	152	3×	✓	N/A	76.6	93.4	80.9	95.5
Ours	152	3×	✓	N/A	**77.6**	**94.2**	**81.3**	**95.7**

Table 4. The accuracy (%) comparison of SSL approaches with other ResNet encoders including selective kernel convolution (SK) [26] based on semi-supervised learning on the ImageNet dataset.

accuracy improvement in Table 5, and our method improves 2.3%, 3.5%, 4.5%, 2.7%, 4.9% and 0.9% on Flood101, SUN397, Cars, Pets, VOC 2007 and Flowers datasets respectively. On average, the transfer learning accuracy of our method is 2.5% higher than the previous best results based on the linear evaluation. For the transfer learning with a fine-tuning task, our method achieves the best accuracy on four out of seven tasks in Table 6.

4.4. Ablation Study

Coefficient τ in the update of student sub-network We investigate the accuracy of our method with linear evaluation based on the ResNet-50 encoder with respective to the smoothing coefficient τ of the exponential moving average (EMA) in Table 7. The bigger the τ is, the smaller update the student sub-network performs. When the τ is 0, it means that we copy the weights of the teacher sub-network to update the student sub-network completely in each step. When the τ is 1, it means that the student sub-network is never updated. We find that at a moving average coefficient value of 0.5, we obtain the best top-1 accuracy, 77.1%, based on the linear evaluation. Neither the moving average coefficient τ of 0 nor 1 generates good performance.

Hard negative pair mining (HNPM) We conduct an ablation study on hard negative pair mining (HNPM) based on a linear evaluation task using the ResNet-200 (2×) encoder on the ImageNet dataset. Training with all negative pairs, *i.e.*, without HNPM, is denoted as "w/o HNPM + block student gradient", and our method is trained with HNPM, which is denoted as "w/ HNPM + block student

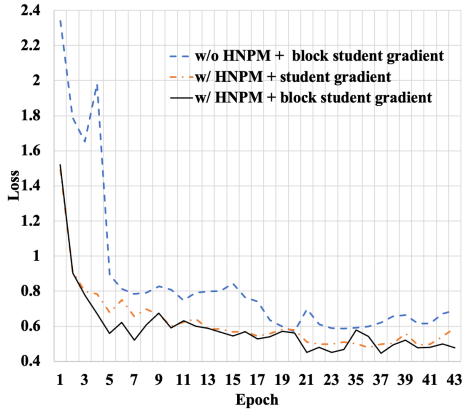

Figure 2. The loss comparison $w.r.t.$ different epochs for ablation study of hard negative pair mining (HNPM) and blocking gradient in the student sub-network based on linear evaluation with the ResNet-200 (2×) encoder on ImageNet.

Method	Food101	CIFAR-10	SUN397	Cars	Pets	VOC 2007	Flowers
BYOL [11]	75.3	91.3	60.6	67.8	90.4	82.5	96.1
SimCLR [5]	68.4	90.6	58.8	50.3	83.6	80.5	91.2
Supervised-IN [5]	72.3	**93.6**	61.9	66.7	91.5	82.8	94.7
Ours	**77.6**	92.4	**65.4**	**72.3**	**94.2**	**87.7**	**97.0**

Table 5. The transfer learning accuracy (%) comparison of SSL approaches with ResNet-50 encoder based on linear evaluation on ImageNet.

Method	Food101	CIFAR-10	SUN397	Cars	Pets	VOC 2007	Flowers
BYOL [11]	88.5	97.8	63.7	91.6	91.7	**85.4**	97.0
SimCLR [5]	88.2	97.7	63.5	91.3	89.2	84.1	97.0
Supervised-IN [5]	88.3	97.5	**64.3**	**92.1**	92.1	85.0	**97.6**
Ours	**89.1**	**98.0**	64.1	**92.1**	**92.8**	85.3	97.5

Table 6. The transfer learning accuracy (%) comparison of SSL approaches with the ResNet-50 encoder based on fine-tuning on ImageNet.

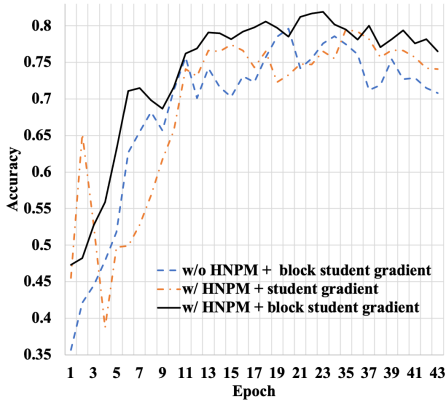

Figure 3. The accuracy comparison $w.r.t.$ different epochs for ablation study of hard negative pair mining (HNPM) and blocking gradient in the student sub-network based on linear evaluation with the ResNet-200 (2×) encoder on the ImageNet dataset.

τ	1.0	0.999	0.5	0.0
Top-1 (%)	24	73.4	**77.1**	49.1

Table 7. The effect of the smoothing coefficient τ in the exponential moving average with ResNet-50 encoder based on linear evaluation on the ImageNet dataset.

gradient". The loss comparison $w.r.t.$ the training epochs for the two methods is visualized in Fig. 2, and the accuracy comparison $w.r.t.$ the training epochs for the two methods are shown in Fig. 3. With hard negative pair mining, the training of our method is much more stable and it achieves lower loss and higher accuracy than that without hard negative pair mining, which validates the conclusion in section 3.5.

Blocking gradient of student sub-network We also conduct an ablation study on the blocking gradient of student sub-network in Fig. 2. Training without blocking the gradient of the student sub-network is denoted as "w/ HNPM + student gradient". Our method achieves lower loss and higher accuracy than that with gradient updating of student sub-network, which draws the same conclusion with SimSiam [8].

5. Conclusion

In this work, we introduce a self-supervised learning framework in a student-teacher network with contrastive loss. To increase the training efficiency, we add the hard negative image pairs into the contrastive self-supervised learning paradigm, named HNSSL. To stabilize the training and avoid a collapsing solution, we block the gradient of the student sub-network and update the parameters of the student sub-network using an exponential moving average. We also conduct an ablation study to validate the effectiveness of each component. Extensive experiments demonstrate that our method achieves better performance than previous state-of-the-art approaches based on linear evaluation, semi-supervised learning, and transfer learning on the ImageNet dataset in the HNSSL when compared based on the same backbone network. In the future, applying various advanced augmentations and the latest image and video encoding Transformer architectures, such as video Transformer [49], audio Transformer [55], multimodal Transformers [51, 56], etc., to the HNSSL framework is an interesting direction to explore.

References

[1] Aayush Bansal, Shugao Ma, Deva Ramanan, and Yaser Sheikh. Recycle-gan: Unsupervised video retargeting. In *ECCV*, pages 119–135, 2018. 1

[2] Mathilde Caron, Piotr Bojanowski, Armand Joulin, and Matthijs Douze. Deep clustering for unsupervised learning of visual features. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 132–149, 2018. 2

[3] Mathilde Caron, Ishan Misra, Julien Mairal, Priya Goyal, Piotr Bojanowski, and Armand Joulin. Unsupervised learning of visual features by contrasting cluster assignments. *Advances in neural information processing systems*, 33:9912–9924, 2020. 6

[4] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 9650–9660, 2021. 1

[5] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pages 1597–1607. PMLR, 2020. 1, 3, 6, 7, 8

[6] Ting Chen, Simon Kornblith, Kevin Swersky, Mohammad Norouzi, and Geoffrey E Hinton. Big self-supervised models are strong semi-supervised learners. *Advances in neural information processing systems*, 33:22243–22255, 2020. 1, 2, 6, 7

[7] Xinlei Chen et al. Improved baselines with momentum contrastive learning. *arXiv preprint arXiv:2003.04297*, 2020. 6

[8] 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. 2, 4, 6, 8

[9] Xinlei Chen, Saining Xie, and Kaiming He. An empirical study of training self-supervised vision transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9640–9649, 2021. 3

[10] Basura Fernando et al. Self-supervised video representation learning with odd-one-out networks. In *CVPR*, 2017. 1

[11] Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Avila Pires, Zhaohan Guo, Mohammad Gheshlaghi Azar, et al. Bootstrap your own latent-a new approach to self-supervised learning. *Advances in neural information processing systems*, 33:21271–21284, 2020. 2, 3, 6, 7, 8

[12] Raia Hadsell, Sumit Chopra, and Yann LeCun. Dimensionality reduction by learning an invariant mapping. In *CVPR*, 2006. 2

[13] Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16000–16009, 2022. 1

[14] Kaiming He et al. Momentum contrast for unsupervised visual representation learning. In *CVPR*, 2020. 2, 3

[15] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, pages 770–778, 2016. 1, 6

[16] Olivier Henaff. Data-efficient image recognition with contrastive predictive coding. In *ICML*, pages 4182–4192. PMLR, 2020. 2, 6

[17] Yufang Huang, Wentao Zhu, Deyi Xiong, Yiye Zhang, Changjian Hu, and Feiyu Xu. Cycle-consistent adversarial autoencoders for unsupervised text style transfer. *arXiv preprint arXiv:2010.00735*, 2020. 1

[18] Longlong Jing and Yingli Tian. Self-supervised visual feature learning with deep neural networks: A survey. *IEEE TPAMI*, 2020. 1, 2

[19] Yannis Kalantidis, Mert Bulent Sariyildiz, Noe Pion, Philippe Weinzaepfel, and Diane Larlus. Hard negative mixing for contrastive learning. *Advances in Neural Information Processing Systems*, 33:21798–21809, 2020. 3

[20] Jacob Devlin Ming-Wei Chang Kenton and Lee Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of NAACL-HLT*, pages 4171–4186, 2019. 2

[21] Diederik P Kingma and Jimmy Ba. Adam: a method for stochastic optimization 3rd int. In *Conf. for Learning Representations, San*, 2014. 5

[22] Simon Kornblith, Jonathon Shlens, and Quoc V Le. Do better imagenet models transfer better? In *CVPR*, pages 2661–2671, 2019. 6

[23] Samuli Laine and Timo Aila. Temporal ensembling for semi-supervised learning. In *International Conference on Learning Representations*, 2017. 2

[24] Gustav Larsson, Michael Maire, and Gregory Shakhnarovich. Colorization as a proxy task for visual understanding. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6874–6883, 2017. 2

[25] Hsin-Ying Lee, Jia-Bin Huang, Maneesh Singh, and Ming-Hsuan Yang. Unsupervised representation learning by sorting sequences. In *ICCV*, 2017. 1

[26] Xiang Li et al. Selective kernel networks. In *CVPR*, 2019. 6, 7

[27] Zeming Li et al. Momentumˆ 2 teacher: Momentum teacher with momentum statistics for self-supervised learning. *arXiv preprint arXiv:2101.07525*, 2021. 3

[28] Tsung-Yi Lin et al. Focal loss for dense object detection. In *ICCV*, 2017. 3

[29] Ilya Loshchilov and Frank Hutter. Sgdr: Stochastic gradient descent with warm restarts. In *International Conference on Learning Representations*, 2016. 5

[30] Jiasen Lu, Dhruv Batra, Devi Parikh, and Stefan Lee. Vilbert: Pretraining task-agnostic visiolinguistic representations for vision-and-language tasks. *Advances in neural information processing systems*, 32, 2019. 2

[31] Michael Mathieu et al. Deep multi-scale video prediction beyond mean square error. In *ICLR*, 2016. 1

[32] Ishan Misra et al. Shuffle and learn: unsupervised learning using temporal order verification. In *ECCV*, 2016. 1

[33] Mehdi Noroozi and Paolo Favaro. Unsupervised learning of visual representations by solving jigsaw puzzles. In *ECCV*. Springer, 2016. 1

[34] Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018. 1, 3, 4

[35] Deepak Pathak et al. Context encoders: Feature learning by inpainting. In *CVPR*, 2016. 1, 2

[36] Deepak Pathak, Ross Girshick, Piotr Dollár, Trevor Darrell, and Bharath Hariharan. Learning features by watching objects move. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2701–2710, 2017. 2

[37] Rui Qian, Tianjian Meng, Boqing Gong, Ming-Hsuan Yang, Huisheng Wang, Serge Belongie, and Yin Cui. Spatiotemporal contrastive video representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6964–6974, 2021. 1

[38] Shaoqing Ren, Kaiming He, Ross B Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *NIPS*, 2015. 3

[39] Olga Russakovsky et al. Imagenet large scale visual recognition challenge. *IJCV*, 2015. 2, 5

[40] Shuwei Shao, Zhongcai Pei, Weihai Chen, Wentao Zhu, Xingming Wu, Dianmin Sun, and Baochang Zhang. Self-supervised monocular depth and ego-motion estimation in endoscopy: Appearance flow to the rescue. *Medical image analysis*, 77:102338, 2022. 1

[41] Liyue Shen, Wentao Zhu, Xiaosong Wang, Lei Xing, John M Pauly, Baris Turkbey, Stephanie Anne Harmon, Thomas Hogue Sanford, Sherif Mehralivand, Peter L Choyke, et al. Multi-domain image completion for random missing input data. *IEEE transactions on medical imaging*, 40(4):1113–1122, 2020. 1

[42] Minchul Shin. Semi-supervised learning with a teacher-student network for generalized attribute prediction. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XI 16*, pages 509–525. Springer, 2020. 3

[43] Antti Tarvainen and Harri Valpola. Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results. In *NeurIPS*, 2017. 3, 4

[44] Yonglong Tian, Dilip Krishnan, and Phillip Isola. Contrastive multiview coding. *arXiv preprint arXiv:1906.05849*, 2019. 1, 2, 6

[45] Yonglong Tian, Chen Sun, Ben Poole, Dilip Krishnan, Cordelia Schmid, and Phillip Isola. What makes for good views for contrastive learning? *Advances in neural information processing systems*, 33:6827–6839, 2020. 6

[46] Ashish Vaswani et al. Attention is all you need. In *NIPS*, 2017. 1

[47] Zhirong Wu, Yuanjun Xiong, Stella X Yu, and Dahua Lin. Unsupervised feature learning via non-parametric instance discrimination. In *CVPR*, 2018. 1, 3

[48] Dejing Xu et al. Self-supervised spatiotemporal learning via video clip order prediction. In *CVPR*, 2019. 1

[49] Xuefan Zha, Wentao Zhu, Lv Xun, Sen Yang, and Ji Liu. Shifted chunk transformer for spatio-temporal representational learning. In *Advances in Neural Information Processing Systems*, volume 34, pages 11384–11396, 2021. 8

[50] Richard Zhang et al. Colorful image colorization. In *ECCV*, 2016. 1, 6

[51] Wentao Zhu, Keval Doshi, Jingru Yi, Xiaohang Sun, Zhu Liu, Linda Liu, Hao Xiang, Mohamed Omar, and Ahmed Saad. Multiscale multimodal transformer for multimodal action recognition. 2022. 8

[52] Wentao Zhu, Yufang Huang, Xiufeng Xie, Wenxian Liu, Jincan Deng, Debing Zhang, Zhangyang Wang, and Ji Liu. Autoshot: A short video dataset and state-of-the-art shot boundary detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2023. 1

[53] Wentao Zhu, Yufang Huang, Daguang Xu, Zhen Qian, Wei Fan, and Xiaohui Xie. Test-time training for deformable multi-scale image registration. In *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pages 13618–13625. IEEE, 2021. 1

[54] Wentao Zhu, Andriy Myronenko, Ziyue Xu, Wenqi Li, Holger Roth, Yufang Huang, Fausto Milletari, and Daguang Xu. Neurreg: Neural registration and its application to image segmentation. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 3617–3626, 2020. 1

[55] Wentao Zhu and Mohamed Omar. Multiscale audio spectrogram transformer for efficient audio classification. In *International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 2023. 8

[56] Wentao Zhu, Mohamed Omar, Jingru Yi, Xiaohang Sun, Kevin Hsu, Burak Uzkent, Ashutosh Sanan, Linda Liu, Xiang Hao, et al. Avt: Audio-video transformer for multimodal action recognition. 2022. 8

[57] Chengxu Zhuang, Alex Lin Zhai, and Daniel Yamins. Local aggregation for unsupervised learning of visual embeddings. In *ICCV*, 2019. 2