Affinity Similarity-Based Contrastive Loss for Unsupervised Visual Representation Learning

Zheng Jiang, Zhou Zhou, Quan Zhou, *Senior Member, IEEE*, Yongan Guo, *Senior Member, IEEE*, Jing Yang, *Member, IEEE*, and Weihua Ou, *Member, IEEE*

***Abstract*—Contrastive loss and its variants are very popular for visual representation learning in an unsupervised scenario, where positive and negative pairs are produced to train a feature encoder based on data augmentation. However, treating only the original image and its augmented versions as positive pairs may lead to problems, since two training images that share the same object category may be misclassified as negative pairs. To address this issue, this paper introduces a plug-in-play affinity similarity module (ASM) used in any contrastive learning framework for unsupervised visual representation learning. In this approach, positive and negative pairs are determined not only based on data augmentation, but also according to the similarity of their feature embeddings. The core idea is that two training samples with higher feature similarities are likely to belong to the same object category, suggesting that they should be classified as a positive pair with high probability, and vice versa for negative pairs. Accordingly, we have also developed an improved affinity similarity-based contrastive loss (ASCL) that leverages the newly generated positive and negative training pairs produced by the ASM. To demonstrate the effectiveness of our method, we extensively evaluate its performance on the ImageNet dataset, showing that it achieves satisfactory results.**

***Index Terms*—Contrastive learning, unsupervised visual learning, positive and negative pairs, affinity similarity module.**

Received 9 March 2025; accepted 11 May 2025. Date of publication 22 May 2025; date of current version 14 August 2025. This work was supported in part by the National Natural Science Foundation of China under Grant 61876093 and Grant 62476139; in part by the Natural Science Foundation of Jiangsu Province under Grant BK2024023; in part by the Intergovernmental Bilateral Innovation Cooperation Project through the Innovation Support Program of Jiangsu Province under Grant BZ2023018; in part by the Frontier Leading Technology Basic Research Program of Jiangsu Province under Grant BK20202001; in part by the High-Level Innovative Talents in Guizhou Province under Grant GCC[2023]033; and in part by the Natural Science Research Project of Department of Education of Guizhou Province under Grant QJJ[2024]009 and Grant QJJ[2023]011. *(Corresponding author: Quan Zhou.)*

Zheng Jiang, Zhou Zhou, and Yongan Guo are with the Jiangsu Key Laboratory of Intelligent Information Processing and Communication Technology, Nanjing University of Posts and Telecommunications, Nanjing 210003, China.

Quan Zhou is with the Jiangsu Key Laboratory of Intelligent Information Processing and Communication Technology, Nanjing University of Posts and Telecommunications, Nanjing 210003, China, also with the Institute for Advanced Ocean Research (Nantong), Southeast University, Nantong 226019, China, and also with the State Key Laboratory of Robotics, Shenyang Institute of Automation, Chinese Academy of Sciences, Shenyang 110004, China (e-mail: quan.zhou@njupt.edu.cn).

Jing Yang is with the State Key Laboratory of Public Big Data, Guizhou University, Guiyang 550025, China (e-mail: jyang23@gzu.edu.cn).

Weihua Ou is with the School of Big Data and Computer Science, Guizhou Normal University, Guiyang 550001, China (e-mail: ouweihuahust@gznu.edu.cn).

Digital Object Identifier 10.1109/TCE.2025.3572120

I. Introduction

UNSUPERVISED representation learning has demonstrated its versatility in consumer electronics and manufacturing, from enabling efficient non-intrusive load monitoring [1] to anomaly detection [2] and wafer map design [3]. As a prominent paradigm of unsupervised representation learning, contrastive learning [4] aims to learn the most discriminative feature representations by maximizing inter-class distances while minimizing intra-class distances. In unsupervised settings, where annotated labels for training samples are not available, data augmentation [5], [6] is widely employed within the contrastive learning framework to construct sample pairs, referred to as positive and negative training pairs used to supervise contrast loss. Specifically, training images are typically augmented using various low-level image processing techniques, such as rotation, flipping, or cropping, etc. Finally, a contrast loss function is trained using these pre-defined training positive and negative pairs.

The most representative approaches for unsupervised visual learning using contrast losses include BYOL [7], Simsiam [8], and MoCo families [9], [10]. In InstDisc [11], a large-scale memory bank is introduced for storing image features, with the goal of expanding the pool of sample pairs. SimCLR [12] enhances representation learning [13], [14], [15] through the use of larger batch sizes, longer training durations, more advanced data augmentations, and an additional linear classifier. MoCo [9] employs a momentum encoder to maintain consistency in target features over time. BYOL [7] features an online network that predicts its own features, bootstrapping the learning process without relying on a target network. Different from MoCo [9] and BYOL [7], SimSiam [8] takes a more streamlined approach by simplifying the architecture and learning directly from the similarity between the query image and its augmented version, without the need for a momentum encoder. Although these advanced unsupervised training frameworks have achieved impressive results for visual representation learning, they inherently suffer from following limitations: as shown in Fig. 1(a), only treating images and its augmented ones as positive pairs may potentially lead to the problems that those training images sharing same object category are misclassified as negative pairs, eventually resulting in learning poor representation of visual data.

To address this problem, this paper designs a novel training pair production strategy for unsupervised visual learning. Intuitively, as illustrated in Fig. 1(b), if two training samples

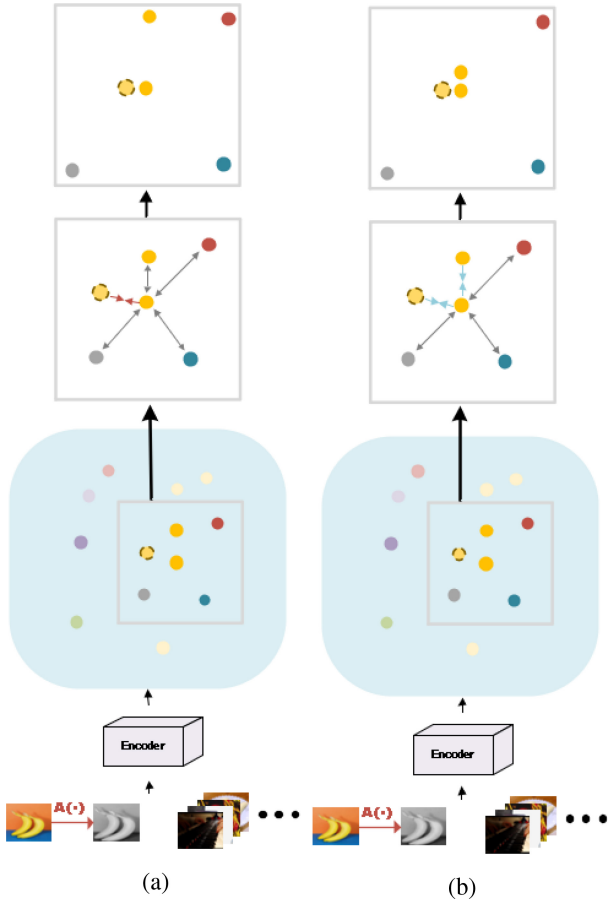

Fig. 1. The difference between traditional contrast learning (a) and ours (b). An augmented image produced by specific affine operator $\mathcal{A}(\cdot)$, together with other training images, is fed into an encoder to produce feature embeddings. Samples with same class are represented by same colors, while others are not. In (a), as there are no supervised signals, the training images and its augmented one are treated as a positive pair, while others are considered as negative pairs. As red arrows shown in (a), using such training pairs will push away the images that potentially have same object class. In contrast, as shown in (b), we measure the similarities of feature embeddings that provide strong evidence to decide whether two training samples belong to same class or not. Thus those samples locate in feature space as closed as possible will be treated as positive pairs with high probabilities, which is benefit for contrast learning, as blue arrows illustrated in (b). (Best viewed in color).

belong to same object category, they tend to have similar feature representation and vice versa. Therefore, we design an affinity similarity module (ASM) to produce positive and negative pairs, where a simple attention mechanism is employed to measure how similar of two training samples are. Based on the output of the ASM, we also design an improved affinity similarity-based contrastive loss (ASCL) supervised from produced positive and negative training pairs. Instead of typically using only one positive pair in traditional contrast loss [16], ASCL is able to explore multiple positive pairs, which enhances the learning of tighter feature representations within the same object category, ultimately maximizing inter-class distance. Moreover, the calculated similarities are also adopted as weight coefficients in ASCL to further enhance representation capabilities.

Overall, there are at least two advantages using ASM. Firstly, both the proposed ASM and ASCL are plug-in-play, making them compatible with any unsupervised visual representation learning framework without the need for additional operations. Secondly, since a simple attention module is used to determine whether two training samples should be classified as positive pairs or not, ASM is computationally cheap and easy to be implemented. We have extensively evaluated our method on ImageNet dataset [17] for image classification tasks, and the experimental results show that the proposed ASM and ASCL achieves new state-of-the-art performance. In nutshell, the major contributions of our paper are three-fold:

- A plug-and-play module, referred to as ASM, is designed to produce positive and negative pairs according to feature similarities. This module can be integrated into any unsupervised visual representation learning framework, whether it employs Convolutional Neural Networks (CNNs) [18], [19] or Transformers [20], [21], [22] as feature encoders.
- Correspondingly, an ASCL is proposed under the supervision of new produced positive and negative training pairs. Unlike previous contrast loss functions [16], ASCL is able to learn more discriminative feature representations by investigating multiple positive pairs.
- We test our approach on ImageNet dataset [17] for image classification tasks. The exhausted experimental results show that our method obtains state-of-the-art performance, achieving 71.7% using ResNet-50, and 74% using ViT-B backbones in terms of top-1 accuracy.

The remainder of this paper is organized as follows. Section II briefly reviews the related works. The details of ASM and ASCL is introduced in Section III. Section IV shows the experimental settings, ablation studies, and results. Finally, the concluding remarks and future work are given in Section V.

II. Related Work

A. Unsupervised Learning

Unsupervised learning often designs pretext tasks that leverage unlabeled data to train models in learning meaningful feature representation [24], [25], [26]. These tasks are not the ultimate objectives but act as a means to facilitate the acquisition of features that can be effectively generalized across various tasks. In the realm of unsupervised learning, a diverse range of pretext tasks exists, including but not limited to autoencoder tasks [27], [28], [29], clustering tasks [25], [30], and generative tasks [31], [32], [33]. The first category trains a neural network to reconstruct its input data, thereby learning a compressed and robust representation through an encoder-decoder architecture. The second category focuses on categorizing data into distinct clusters, promoting intra-cluster similarity and inter-cluster dissimilarity. In contrast, generative tasks [31], [32], [33] aiming to create new data instances that are consistent with the statistical properties of the training data, using models that capture the underlying data distribution. In comparison to these traditional unsupervised learning methods, our approach is rooted in contrastive learning [8], [34], [35], which operates on the principles of intra-class aggregation and inter-class dispersion. This enhances the model's ability to discern subtle differences within categories as well as distinctions between them.

B. Contrastive Learning

Contrastive learning [4], as one of the most prominent approach to unsupervised learning, aims to train models by identifying the similarities and differences between positive and negative pairs [36]. A common strategy for generating standard positive pairs involves applying data augmentation to create multiple views of each image, treating all views except the current one as negatives. The widely used contrastive loss function [16] encourages the cohesion of positive pairs in the embedding space while enforcing the separation of negative pairs. InstDisc [11] introduces a large-scale memory bank for storing image features, thereby increasing the number of available negative pairs. MoCo [9] frames contrastive learning as a dictionary learning problem, maintaining an online queue for storing negative pairs and utilizing a momentum encoder with stop-gradient operations and momentum updates [11], [15]. SimCLR [12] leverages larger batch sizes, extended training durations, and more sophisticated data augmentations. SwAV [26] combines contrastive learning and clustering methods, clustering similar objects around a central point while pushing dissimilar objects toward other cluster centers. Conversely, BYOL [7] eliminates the need for negative pairs, learning exclusively from positive pairs by pulling the feature representations of two views of an image as closely together as possible. SimSiam [8] further simplifies BYOL [7], demonstrating that satisfactory results can be achieved even without the need for negative pairs, large batch size, and momentum encoders. No matter what contrastive learning strategy is used, the loss function must be defined based on multi-view image pairs produced through data augmentation. Our method diverges from traditional contrastive learning by employing affinity similarity as a heuristic for selecting positive pairs, rather than relying solely on data augmentation.

C. Visual Attention

Attention mechanisms empower models to concentrate on the most salient aspects of input data, thereby enhancing both model performance and efficiency. The core is the dynamic allocation of weights by models to emphasize important information across various inputs. In unsupervised learning, Masked Autoencoders (MAE) [24] introduces an innovative attention mechanism within its autoencoder framework, emphasizing the model's ability to focus on key features in the data. The ITPN [27] leverages a pyramid of attention mechanisms for robust feature extraction and efficient processing in vision tasks, representing a significant advancement in unsupervised learning. Incorporating attention mechanisms into contrastive learning can enhance the model's ability to focus on key information in the input data, thereby improving the quality of feature representation. However, there has been a noticeable lack of studies combining attention and contrastive learning in recent years. MoCov3 [10] distinguishes itself in the field of contrastive learning by adopting the Vision Transformer (ViT) as its backbone, which inherently incorporates attention to effectively capture and model spatial hierarchies in visual data, leading to enhanced feature representations. CLIP [37] harnesses visual attention via ViT to adeptly align image features with text prompts during contrastive learning, significantly enhancing its capability for zero-shot classification. While previous works have predominantly focused on using attention to strengthen backbone networks in contrastive learning, we shift the focus from intra-image to inter-image attention, extracting affinity similarity between images through visual attention, and heuristically selecting positive pairs while weighting negative pairs.

Algorithm 1 Our Training Algorithm

Input: T: total number of training epochs, B: total number of training batches, $\boldsymbol{X_b}$: training images in b^{th} batch, f_T: teacher networks, f_S: student networks, σ: momentum coefficient, η: training epoch parameter, δ: judgment threshold
Output: $\boldsymbol{\theta}_t$: teacher network parameters, $\boldsymbol{\theta}_s$: student network parameters
Randomly initialize $\boldsymbol{\theta}_t$ and $\boldsymbol{\theta}_s$
for each $t \in [1, T]$ **do**
  for each $b \in [1, B]$ **do**
    Initialize loss: $\mathcal{L}_b \leftarrow 0$
    Apply batch augmentations: $\boldsymbol{X}_b^+ \leftarrow \mathcal{A}(\boldsymbol{X}_b)$
    Precompute features:
      $\boldsymbol{Q}_b \leftarrow f_T(\boldsymbol{X}_b, \boldsymbol{\theta}_t)$, $\boldsymbol{Q}_b^+ \leftarrow f_T(\boldsymbol{X}_b^+, \boldsymbol{\theta}_t)$
      $\boldsymbol{K}_b \leftarrow f_S(\boldsymbol{X}_b, \boldsymbol{\theta}_s)$, $\boldsymbol{K}_b^+ \leftarrow f_S(\boldsymbol{X}_b^+, \boldsymbol{\theta}_s)$
    for each $i \in [1, N]$ **do**
      Compute similarity for $\boldsymbol{x}_i$:
        $\boldsymbol{A}_{i,j} \leftarrow \frac{\boldsymbol{q}_i \cdot \boldsymbol{k}_j^+}{\|\boldsymbol{q}_i\| \|\boldsymbol{k}_j^+\|}, \forall j \in [1, N]$ \\ Eq (1)
      Generate mask $\boldsymbol{M}_{i,j}$ via threshold δ \\ Eq (2)
      Build $\tilde{\boldsymbol{P}}_i$, $\tilde{\boldsymbol{N}}_i$ for $\boldsymbol{x}_i$ \\ Eq (3)
      if $t < \eta$ **then**
        Compute $\mathcal{L}_i \leftarrow \mathcal{L}^{tra}$ \\ Eq (11)
      else
        Compute $\mathcal{L}_i \leftarrow \mathcal{L}^{ASCL}$ \\ Eq (9)
      end if
      Accumulate loss: $\mathcal{L}_b \leftarrow \mathcal{L}_b + \mathcal{L}_i$
    end for
    Normalize loss: $\mathcal{L}_b \leftarrow \mathcal{L}_b / N$
    Update $\boldsymbol{\theta}_t$ via SGD: $\nabla_{\boldsymbol{\theta}_t} \mathcal{L}_b$
    Update $\boldsymbol{\theta}_s$ via EMA: $\boldsymbol{\theta}_s \leftarrow (1 - \sigma) \times \boldsymbol{\theta}_t + \sigma \times \boldsymbol{\theta}_s$
  end for
end for
return $\boldsymbol{\theta}_t$, $\boldsymbol{\theta}_s$

III. Our Method

In this section, we first outline the feature extraction component of our framework, namely teacher-student network. Then, we proceed to the details of ASM. Finally, we elaborate on the details of ASCL, and summarize the whole training process in Algorithm 1. The complete pipeline of our contrastive learning framework is depicted in Fig. 2.

A. Teacher-Student Network

Regarding the details of the network architecture, our design follows the MoCo v3 framework [10], implementing

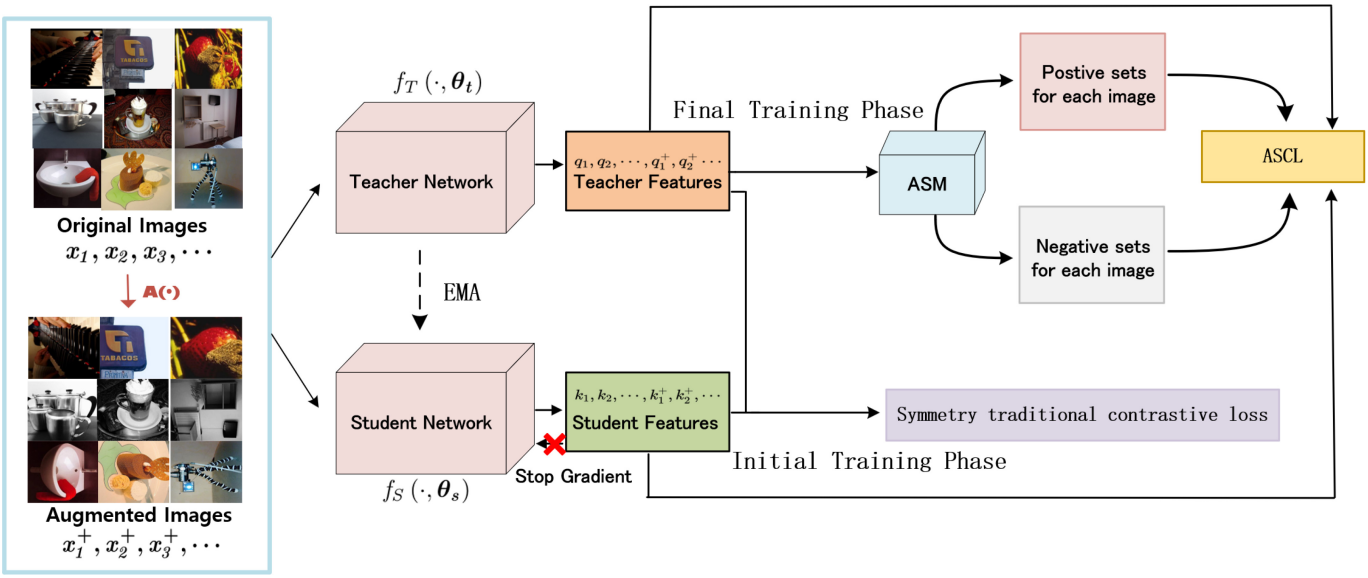

Fig. 2. **The contrastive learning framework with ASM and ASCL.** Original images and their augmented counterparts are fed into the teacher-student network, yielding distinct feature representations from both networks. The student network is learned via exponential moving average (EMA) [23] from the teacher network. During the initial training phase, we employ symmetric traditional contrastive loss [16] to train feature encoder of teacher network. This critical baseline enables reliable affinity-based pair discrimination for subsequent modules. The final phase implements two key components: ASM and ASCL. ASM dynamically identifies positive and negative relationships within the training batch by calculating feature affinity scores. ASCL is used to enhance inter-class distinction and intra-class compactness based on the constructed training pairs.

a teacher-student dual-branch structure. The teacher branch consists of a backbone encoder, a projection head, and a prediction head, while the student branch retains only the projection head along with a momentum-updated backbone to maintain representation consistency. We utilize either ResNet-50 [19] or Vision Transformer (ViT) [20] as the shared backbone across both the teacher and student networks.

Teacher-student network strives to learn the underlying structure and derive meaningful feature representations from a large unlabeled dataset. Let $f_T(\cdot, \boldsymbol{\theta}_t)$ and $f_S(\cdot, \boldsymbol{\theta}_s)$ be the teacher network and the student network, respectively, where $\boldsymbol{\theta}_t$ and $\boldsymbol{\theta}_s$ denote model parameters for each network. Let $\boldsymbol{X} = \{\boldsymbol{x}_1, \boldsymbol{x}_2, \boldsymbol{x}_3, \ldots, \boldsymbol{x}_N\}$ be an image batch, where each component $\boldsymbol{x}_i$ denotes an individual image. For i^{th} image $\boldsymbol{x}_i$, we first produce its augmented image $\boldsymbol{x}_i^+$ using $\mathcal{A}(\cdot)$ by randomly applying one type of augmentation operation, such as random horizontal flipping, cropping, color jittering, and Gaussian blurring. Both the original image $\boldsymbol{x}_i$ and its augmented one $\boldsymbol{x}_i^+$ serve as inputs to the teacher network $f_T(\cdot, \boldsymbol{\theta}_t)$ and the student network $f_S(\cdot, \boldsymbol{\theta}_s)$, producing output features $\boldsymbol{q}_i = f_T(\boldsymbol{x}_i, \boldsymbol{\theta}_t)$, $\boldsymbol{q}_i^+ = f_T(\boldsymbol{x}_i^+, \boldsymbol{\theta}_t)$, as well as $\boldsymbol{k}_i = f_S(\boldsymbol{x}_i, \boldsymbol{\theta}_t)$, $\boldsymbol{k}_i^+ = f_S(\boldsymbol{x}_i^+, \boldsymbol{\theta}_t)$. By collecting all these features, we obtain four feature sets: $\boldsymbol{Q} = (\boldsymbol{q}_1, \boldsymbol{q}_2, \ldots, \boldsymbol{q}_N)$, $\boldsymbol{Q}^+ = (\boldsymbol{q}_1^+, \boldsymbol{q}_2^+, \ldots, \boldsymbol{q}_N^+)$, as well as $\boldsymbol{K} = (\boldsymbol{k}_1, \boldsymbol{k}_2, \ldots, \boldsymbol{k}_N)$, $\boldsymbol{K}^+ = (\boldsymbol{k}_1^+, \boldsymbol{k}_2^+, \ldots, \boldsymbol{k}_N^+)$, each of which is fed into ASM and ASCL introduced in Section III-B and Section III-C, ready to construct positive and negative training pairs.

B. Affinity Similarity Module (ASM)

The ASM leverages the similarities among various image features to heuristically select positive and negative images within an unsupervised learning framework, facilitating the construction of positive and negative training pairs. As illustrated in Fig. 3, the key idea of ASM lies in using image features to produce a mask matrix, which is then employed to create positive and negative sets. Specifically, given the features $\boldsymbol{q_i} \in \boldsymbol{Q}$ encoded by the teacher network and $\boldsymbol{k_j^+} \in \boldsymbol{K^+}$ encoded by the student network, respectively, we compute the attention map $\boldsymbol{A}_{i,j}$ to measure feature similarities, with each element indicating which training images are most likely to belong to the same category within a training batch:

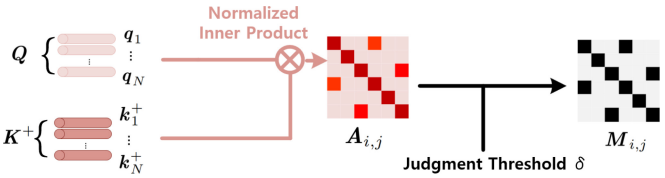

Fig. 3. **Generation of mask matrix $\boldsymbol{M}_{i,j}$.** The color intensity in $\boldsymbol{A}_{i,j}$ represents the pairwise similarity between i and j, with darker red indicating higher similarity. In $\boldsymbol{M}_{i,j}$, white denotes 0 and black denotes 1, representing the binary mask values.

$$\boldsymbol{A}_{i,j} = \frac{\boldsymbol{q}_i \cdot \boldsymbol{k}_j^+}{\|\boldsymbol{q}_i\| \|\boldsymbol{k}_j^+\|}, \quad \text{for } i, j \in [1, N] \tag{1}$$

Then we apply a hard decision to determine whether a pair of samples belongs to a same class, based on the principle that the corresponding element $\boldsymbol{A}_{i,j}$ exceeds a predefined judgment threshold δ:

$$\boldsymbol{M}_{i,j} = \begin{cases} 1, \text{ if } \boldsymbol{A}_{i,j} \geq \delta \\ 0, \text{ otherwise} \end{cases} \tag{2}$$

For the i^{th} image in the batch, we generate positive sets $\tilde{\boldsymbol{P}}_i$ and negative sets $\tilde{\boldsymbol{N}}_i$ based on the value of $\boldsymbol{M}_{i,j}$:

$$\tilde{\boldsymbol{P}}_i = \left\{ \boldsymbol{k}_j^+ \mid j \in \{1, \ldots, N\}, \boldsymbol{M}_{i,j} = 1 \right\}$$
$$\tilde{\boldsymbol{N}}_i = \left\{ \boldsymbol{k}_j^+ \mid j \in \{1, \ldots, N\}, \boldsymbol{M}_{i,j} = 0 \right\} \tag{3}$$

where $|\tilde{\boldsymbol{P}}_i| + |\tilde{\boldsymbol{N}}_i| = N$. Selecting any sample features $\boldsymbol{k}_j^+$ from either the positive or negative set and pairing them with anchor features $\boldsymbol{q}_i$, we are able to create positive and negative training pairs that are ready to be fed into ASCL in the following section.

C. Affinity Similarity-Based Contrastive Loss (ASCL)

Before introducing ASCL, we first provide a brief review of the traditional contrastive loss function [16], serving as the foundation for our ASCL. In the context of unsupervised learning, the goal of contrastive learning is to maximize inter-class distinctions while minimizing intra-class compactness. According to this principle, the traditional contrastive loss is formally defined as follows:

$$\mathcal{L}\left(\boldsymbol{q}_i, \boldsymbol{k}_j^+\right) = -\sum_{i=1}^{N} \log \frac{\exp\left(\frac{\boldsymbol{q}_i \cdot \boldsymbol{k}_j^+}{\gamma}\right)}{\sum_{j=1}^{N} \exp\left(\frac{\boldsymbol{q}_i \cdot \boldsymbol{k}_j^+}{\gamma}\right)} \tag{4}$$

where γ is the temperature coefficient that regulates the distribution of similarities, thereby contributing to the stabilization of the training process.

The loss function defined in Eq. (4) only takes into account one positive training pair, treating all other samples as negative pairs. However, this assumption is not always true when there are multiple samples from the same class in a training batch. Incorrectly labeling those samples that should be positive pairs as negative ones will inevitably result in inconsistent representation learning. Thanks to the ASM introduced in Section III-B, for i^{th} anchor feature $\boldsymbol{q}_i$, we are allowed to explore multiple positive training pairs indicated by $\tilde{\boldsymbol{P}}_i$. Therefore, the contrastive loss defined in Eq. (4) can be rewritten as:

$$\mathcal{L}\left(\boldsymbol{q}_i, \boldsymbol{k}_j^+\right) = -\sum_{i=1}^{N} \log \frac{\sum_{\boldsymbol{k}_j^+ \in \tilde{\boldsymbol{P}}_i} \exp\left(\frac{\boldsymbol{q}_i \cdot \boldsymbol{k}_j^+}{\gamma}\right)}{\sum_{\boldsymbol{k}_j^+ \in \tilde{\boldsymbol{P}}_i} \exp\left(\frac{\boldsymbol{q}_i \cdot \boldsymbol{k}_j^+}{\gamma}\right) + \sum_{\boldsymbol{k}_j^+ \in \tilde{\boldsymbol{N}}_i} \exp\left(\frac{\boldsymbol{q}_i \cdot \boldsymbol{k}_j^+}{\gamma}\right)} \tag{5}$$

In order to further learning discriminative feature representation, we also introduce a negative pair weighting strategy into our loss function. Intuitively, traditional contrastive loss defined in Eq. (4) encourages negative training pairs to have low similarity in feature space while penalizing those with higher similarity. Following this paradigm, we assign greater weights to these similarities, thereby facilitating the process of pushing away these challenging negative samples. Let $\lambda_{i,j}$ be the weight of one specific negative pair consisting of image features $\boldsymbol{q}_i$ and $\boldsymbol{k}_j^+ \in \tilde{\boldsymbol{N}}_i$, then our weighting mechanism is defined as:

$$\lambda_{i,j} = \begin{cases} 1, & \text{if } K \times \boldsymbol{A}_{i,j} \leq 1 \\ K \times \boldsymbol{A}_{i,j}, & \text{otherwise} \end{cases} \tag{6}$$

where K is a constant coefficient, and $\boldsymbol{A}_{i,j}$ is defined in Eq. (1). Then, our final contrastive loss is defined as:

$$\mathcal{L} = -\sum_{i=1}^{N} \log \frac{\sum_{\boldsymbol{k}_j^+ \in \tilde{\boldsymbol{P}}_i} \exp\left(\frac{\boldsymbol{q}_i \cdot \boldsymbol{k}_j^+}{\gamma}\right)}{\sum_{\boldsymbol{k}_j^+ \in \tilde{\boldsymbol{P}}_i} \exp\left(\frac{\boldsymbol{q}_i \cdot \boldsymbol{k}_j^+}{\gamma}\right) + \sum_{\boldsymbol{k}_j^+ \in \tilde{\boldsymbol{N}}_i} \lambda_{i,j} \exp\left(\frac{\boldsymbol{q}_i \cdot \boldsymbol{k}_j^+}{\gamma}\right)} \tag{7}$$

where independent variables $\boldsymbol{q}_i$ and $\boldsymbol{k}_i^+$ are omitted for notation simplification. As exhibited in Eq. (7), when two samples in a negative pair have low similarities, our ASCL degenerates to traditional contrastive loss. Otherwise, we assign a big punish wights if they have very high similarities.

Eq. (7) defines the contrastive loss solely between the original images and their augmented counterparts. However, augmented images also provide valuable information that can improve the capability of loss function to learn better feature representations. Therefore, we propose a symmetric contrastive loss based on Eq. (7), which encourages a more comprehensive exploration of the feature space:

$$\mathcal{L}^{sym} = -\sum_{i=1}^{N} \log \frac{\sum_{\boldsymbol{k}_j \in \tilde{\boldsymbol{P}}_i} \exp\left(\frac{\boldsymbol{q}_i^+ \cdot \boldsymbol{k}_j}{\gamma}\right)}{\sum_{\boldsymbol{k}_j \in \tilde{\boldsymbol{P}}_i} \exp\left(\frac{\boldsymbol{q}_i^+ \cdot \boldsymbol{k}_j}{\gamma}\right) + \sum_{\boldsymbol{k}_j \in \tilde{\boldsymbol{N}}_i} \lambda_{i,j} \exp\left(\frac{\boldsymbol{q}_i^+ \cdot \boldsymbol{k}_j}{\gamma}\right)} \tag{8}$$

By combining contrastive loss $\mathcal{L}$ and its symmetric counterpart $\mathcal{L}^{sym}$, we can leverage information from both the original-to-augmented and augmented-to-original directions. As a result, these two loss functions have to be equally combined, producing our final ASCL:

$$\mathcal{L}^{ASCL} = \frac{1}{2}\left(\mathcal{L} + \mathcal{L}^{sym}\right) \tag{9}$$

During the training process, both our $\mathcal{L}^{ASCL}$ and traditional contrastive loss will be used to learn feature representations. To ensure a fairness, we thus improve a symmetry version of traditional contrastive loss:

$$\mathcal{L}\left(\boldsymbol{q}_i^+, \boldsymbol{k}_j\right) = -\sum_{i=1}^{N} \log \frac{\exp\left(\frac{\boldsymbol{q}_i^+ \cdot \boldsymbol{k}_j}{\gamma}\right)}{\sum_{j=1}^{N} \exp\left(\frac{\boldsymbol{q}_i^+ \cdot \boldsymbol{k}_j}{\gamma}\right)} \tag{10}$$

Similarly, the final traditional loss $\mathcal{L}^{tra}$ is defined as:

$$\mathcal{L}^{tra} = \frac{1}{2}\left[\mathcal{L}\left(\boldsymbol{q}_i, \boldsymbol{k}_j^+\right) + \mathcal{L}^{sym}\left(\boldsymbol{q}_i^+, \boldsymbol{k}_j\right)\right] \tag{11}$$

Here, $\mathcal{L}^{ASCL}$ and $\mathcal{L}^{tra}$ are both used to supervise the teacher $f_T(\cdot, \boldsymbol{\theta}_t)$ and student backbone $f_S(\cdot, \boldsymbol{\theta}_s)$, respectively.

D. Training Algorithm

This section presents the entire training algorithm under the supervision of $\mathcal{L}^{ASCL}$ and $\mathcal{L}^{tra}$. Following [9], [10], we adopt different parameter updating strategies for teacher network $f_T(\cdot, \boldsymbol{\theta}_t)$ and student network $f_S(\cdot, \boldsymbol{\theta}_s)$. More specifically, we use statistic gradient descent (SGD) [38] to optimize $\boldsymbol{\theta}_t$ as

usual, where the updated gradients are propagated backward from the network output to the input. In contrast, an exponential moving average (EMA) [9] strategy is employed to update $\boldsymbol{\theta}_s$, where the gradients are not allowed to be back-propagated. This process is formally defined as:

$$\boldsymbol{\theta}_s \leftarrow (1-\sigma) \times \boldsymbol{\theta}_t + \sigma \times \boldsymbol{\theta}_s \tag{12}$$

where σ denotes the momentum smoothing coefficient, which controls the update rate of $\boldsymbol{\theta}_s$.

Given the limited representation capability of the encoders at the beginning of the training phase, it is not advisable to apply our proposed $\mathcal{L}^{ASCL}$ directly to supervise the entire training process. To address this issue, we first use the traditional loss $\mathcal{L}^{tra}$, which utilizes both the original images and their augmented counterparts. Once the image features are sufficiently initialized through the encoder backbones, we then replace $\mathcal{L}^{tra}$ with $\mathcal{L}^{ASCL}$, which is able to explore multiple positive training pairs to enhance feature representation learning. It is important to note that introducing $\mathcal{L}^{ASCL}$ too early might result in incorrect labeling of positive and negative pairs based on feature similarity. Conversely, introducing $\mathcal{L}^{ASCL}$ too late could lead to optimizing $\boldsymbol{\theta}_t$ and $\boldsymbol{\theta}_s$ primarily using $\mathcal{L}^{tra}$. Therefore, a natural question arises: when is it appropriate to implement this replacement during the training phase? To address this, we employ a training epoch parameter η that controls the switch between two loss functions, where η is determined by ablation studies. In general, the whole training algorithm is summarized in Algorithm 1. After the teacher network $f_T(\cdot, \boldsymbol{\theta}_t)$ and the student network $f_S(\cdot, \boldsymbol{\theta}_s)$ have been effectively pre-trained, only $f_T(\cdot, \boldsymbol{\theta}_t)$ is paired with head of downstream tasks to perform fine-tuning and inference.

IV. Experiments

A. Dataset and Evaluation Metrics

1) ImageNet: ImageNet [17] serves as a foundational large-scale benchmark in computer vision, driving significant advancements in visual representation learning. It comprises over 14 million human-annotated images systematically organized into 1,000 hierarchical categories spanning diverse visual concepts. Our experimental implementation follows the standard split configuration [19], [39]: 1.2 million training samples for model optimization, 50,000 validation images for fine-tuning, and the rest 100,000 images reserved for final performance evaluation.

2) Evaluation Metrics: Top-1 accuracy [39] serves as our primary performance benchmark in classification tasks. This metric is defined as the percentage of samples where the model's single highest-confidence prediction matches the true label, offering an unambiguous assessment of classification effectiveness.

B. Implementation Details

1) Training Settings: To assess the quality of unsupervised feature representations learned through our self-supervised framework, we employ the standard linear probing protocol [9], [12]. This methodology involves three stages: (1) self-supervised pretraining of the encoder on unlabeled data exclusively, (2) freezing the encoder's parameter weights while discarding all projection heads, and (3) training a linear classification layer at the top of frozen backbone using labeled samples. The resultant classification accuracy serves as an effective quantitative proxy for representation quality, isolating the encoder's discriminative capability from subsequent task-specific adaptations.

Concretely, our implementation employs full-scratch training on ImageNet [17] using 8 NVIDIA RTX 3090 GPUs, with batch size optimized for hardware scalability. Departing from conventional pretraining paradigms, our architecture implements a novel teacher-student framework without inherited backbone weights. The training process is conducted through 200 epochs of stochastic gradient descent [38], where the initialized learning rate is set as 4.0, together with weight decay and momentum are set as 1×10^{-6} and 0.9, respectively. For ASM and ASCL hyperparameters, we adopt the optimal values through extensive experiments, with the sample pair judgment threshold parameter δ of 0.8 and the phase of introducing ASM and ASCL η of 150 (detailed ablation studies can be found in Section IV-D). In order to produce multi-view data with diversity, we employ an asymmetric augmentation strategy to capture various transformation strategies for teacher and student branches, respectively; more specifically, for the teacher network, the input images are augmented using random cropping and resizing to 224 $\times$ 224, color jittering (with brightness of 0.4, contrast of 0.4, saturation of 0.2, and hue of 0.1, each with a probability of 0.8), grayscale conversion with a probability of 0.2, and Gaussian blur filtering with a radius ranging from 0.1 to 2.0, along with random horizontal flipping. For the student network, solarization with a probability of 0.2 is additionally employed alongside these operations.

2) Loss Settings: As illustrated in Algorithm 1, the entire training phase is implemented by Eqs. (11) and (9), where contrastive loss facilitates the learning of initialized feature representations, followed by ASCL that explores multiple positive training pairs. Following [10], we adopt the temperature coefficient γ of 1.0 in contrast loss. In Algorithm 1, this process is controlled by hyperparameter η, which is set as 150 in our experiment. The empirical study on the impact of different η on model performance is presented in Section IV-D.

C. Comparisons With State-of-the-Art Contrastive Learning Methods

In Table I, we compare our method with previous state-of-the-art unsupervised contrastive learning methods on the ImageNet dataset and present the top-1 accuracy results. Employing a standard ResNet-50 as a benchmark, our proposed methodology achieves a top-1 accuracy of 71.7%, which represents a significant enhancement of 1.1% over the previously established state-of-the-art results in this domain. By transitioning the backbone to ViT-B/16, our method achieves a top-1 accuracy of 73.0%, marking an increment of 2.3%. The above results demonstrate that our method achieves superior performance in representation learning, verifying its effectiveness.

TABLE I
COMPARISON OF TOP-1 CLASSIFICATION ACCURACY ON THE IMAGENET DATASET

Method	Backbone	Top-1(%)
SimCLR [12]	Resnet-50	66.5
MoCov2 [9]	Resnet-50	67.5
SWAV [26]	Resnet-50	69.1
SimSiam [8]	Resnet-50	70.0
BYOL [7]	Resnet-50	70.6
MoCov3 [10]	Resnet-50	71.0
OUR	Resnet-50	71.7
OUR	ViT-B/16	74.0

TABLE II
ABLATION STUDIES FOR COMPONENTS OF ASM AND ASCL ON THE IMAGENET DATASET

ASM	ASCL	Top-1(%)
		71.050
	✓	71.418 +0.368↑
✓		71.480 +0.420↑
✓	✓	71.712 +0.662↑

TABLE III
ABLATION STUDIES FOR THE SAMPLE PAIR JUDGMENT THRESHOLD PARAMETER δ ON THE IMAGENET DATASET

δ	0.6	0.7	0.8	0.9
Top-1(%)	71.474	71.552	**71.712**	71.458

TABLE IV
ABLATION STUDIES FOR TRAINING EPOCH PARAMETER η ON THE IMAGENET DATASET

η	50	100	150	175
Top-1(%)	71.154	71.464	**71.712**	71.522

TABLE V
ABLATION STUDIES FOR THE NEGATIVE PAIR WEIGHTING HYPERPARAMETER K ON THE IMAGENET DATASET

K	10	20	30
Top-1(%)	71.684	**71.712**	71.492

D. Ablation Study

To understand the underlying behavior of ASM and ASCL, this section reports the results of a series of ablation studies.

1) Ablation Studies for Components of ASM and ASCL: Table II presents ablation studies quantifying the contributions of ASM and ASCL. We begin with a baseline that employs traditional contrastive loss without incorporating ASM and ASCL, achieving a top-1 classification accuracy of 71.05%. To assess the contribution of the negative pair weighting strategy, we directly apply it within the traditional contrastive loss, resulting in the improved performance from 71.05% to 71.42%. Utilizing the proposed ASM yields an additional improvement of 0.42% over the baseline, demonstrating the effectiveness of employing multiple positive training pairs. Finally, when ASM and ASCL are combined, we achieve the best performance (71.712%) in terms of top-1 accuracy. The results reported in Table II demonstrate that each of these components consistently enhances classification performance, with their combination yielding the most significant benefit.

2) Ablation Studies for the Sample Pair Judgment Threshold Parameter δ: The selection of positive and negative pairs by ASM is related to the sample pair judgment threshold parameter δ. In pursuit of a deeper understanding of ASM, we undertake a series of experiments that systematically vary the hyperparameter δ. In Table III, we incrementally assign values to δ in the sequence of 0.6, 0.7, 0.8, and 0.9, and subsequently assess the impact of each on the model's predictive accuracy when evaluated against the ImageNet dataset. The results demonstrate that when δ is set to 0.8, the model achieves the optimal performance with a top-1 accuracy of 71.712%. We observe that both overly strict (δ=0.9) and overly loose (δ=0.6) threshold settings lead to performance degradation, with accuracies of 71.458% and 71.474% respectively. This suggests that a moderate threshold strikes the best balance between positive pair selection stringency and model performance.

3) Ablation Studies for Training Epoch Parameter η: The training epoch parameter of introducing ASM and ASCL with η is significant. In Table IV, we sequentially select the values 50, 100, 150, and 175 for the hyperparameter η and meticulously assess their top-1 accuracies on the ImageNet dataset. When η is set to 150, we achieve the optimal top-1 accuracy 71.712%. Values of η that are either excessively large or small can result in a decline in model performance, as evidenced by the lower accuracies of 71.154% when introduced too early (η=50) and 71.522% when introduced too late (η=175). This indicates that the timing of introducing ASM and ASCL is crucial for allowing the encoder to develop sufficient feature representation capabilities while maintaining optimal model convergence.

4) Ablation Studies for the Negative Pair Weighting Hyperparameter K: The hyperparameter K significantly influences the weighting degree of high-similarity negative pairs. In Table V, we meticulously assign the parameter K the values of 10, 20, and 30 in succession, conducting a comprehensive assessment of the resultant top-1 accuracy on the ImageNet dataset. Upon setting K to 20, we attain the optimal top-1 accuracy 71.712%. A smaller value (K=10) yields a slightly lower accuracy of 71.684%, while a larger value (K=30) leads to more significant performance degradation with an accuracy of 71.492%. This suggests that a moderate value of K provides the most effective weighting for high-similarity negative pairs.

5) Ablation Studies for Computation Overhead by Introducing ASM and ASCL: To evaluate the computation overhead of our proposed ASM and ASCL framework, we first establish a baseline under the unsupervised learning framework of MoCov3 [10], and then sequentially integrate the ASM and ASCL into this baseline. As shown in Table VI, we compare the computation time per epoch across

TABLE VI
ABLATION STUDY OF COMPUTATION OVERHEAD BY INTRODUCING ASM AND ASCL UNDER THE FRAMEWORK OF MOCOV3 [10]

baseline	ASM	ASCL	Computational costs (min)
✓			21.62
✓	✓		21.66 +0.04↑
✓	✓	✓	21.74 +0.12↑

TABLE VII
COMPARISON OF TOP-1 CLASSIFICATION ACCURACY OF INTEGRATING ASM AND ASCL INTO SIMCLR AND MOCOV2 ON THE IMAGENET DATASET

Method	Top-1(%)
SimCLR [12]	66.512
SimCLR(with OUR)	66.857+0.345↑
MoCov2 [9]	67.548
MoCov2(with OUR)	67.932+0.384↑

different methods. When integrated with MoCov3 framework, our ASM component introduces minimal overhead, adding only 0.04 minutes per epoch. With both ASM and ASCL components integrated, the additional computation time is merely 0.12 minutes compared to the baseline, which takes 21.62 minutes per epoch. These results show that our framework provides substantial performance improvements while bringing almost negligible additional computation.

E. "Plug-and-Play" Validation

ASM and ASCL embody the principles of plug-and-play interoperability. We demonstrate the plug-and-play nature of our method by showcasing its adaptability and universality across diverse unsupervised contrastive learning frameworks. To validate this, we integrate ASM and ASCL into several recent unsupervised contrastive learning frameworks. Our performance comparisons, detailed in Table VII, underscore the module's seamless integration and its capacity to augment various learning strategies without necessitating extensive modifications.Specifically, when integrated into SimCLR, our method improves the top-1 accuracy from 66.512% to 66.857%, achieving a gain of 0.345%. Similarly, when applied to MoCov2, the accuracy increases from 67.548% to 67.932%, showing an improvement of 0.384%. These consistent improvements across different frameworks demonstrate the generalizability of our proposed modules.

V. CONCLUSION AND FUTURE WORK

This paper propose an ASCL for unsupervised visual representation learning. In contrast to previous unsupervised contrastive learning methods that utilize the judgement of positive and negative pairs, our approach introduces a plug-in-play ASM. This module classifies similar pairs as positive and dissimilar pairs as negative, based on the similarity of the image sample features. This effectively prevents the issue of misclassifying multiple images of the same class in the training set as negative pairs during the training process. Additionally, We also design an improved ASCL that applies stronger penalties to negative pairs with higher similarities. Extensive experiments demonstrate our method obtain impressive results in linear evaluation. In future work, we plan to extend our similarity-based weighting strategy to GNNs [40], [41], [42], [43] for enhanced representation discriminability and incorporate complementary techniques including adversarial learning [44], adaptive feature fusion [45], and multi-scale contrastive enhancement [46] to strengthen our framework.

REFERENCES

[1] Q. Liu, K. M. Kamoto, X. Liu, M. Sun, and N. Linge, "Low-complexity non-intrusive load monitoring using unsupervised learning and generalized appliance models," *IEEE Trans. Consum. Electron.*, vol. 65, no. 1, pp. 28–37, Feb. 2019.

[2] B. Xu, J. Wang, Z. Zhao, H. Lin, and F. Xia, "Unsupervised anomaly detection on attributed networks with graph contrastive learning for consumer electronics security," *IEEE Trans. Consum. Electron.*, vol. 70, no. 1, pp. 1–10, Feb. 2024.

[3] Q. Xu, N. Yu, and H. Yu, "Unsupervised representation learning for large-scale wafer maps in micro-electronic manufacturing," *IEEE Trans. Consum. Electron.*, vol. 70, no. 1, pp. 1226–1235, Feb. 2024.

[4] R. Hadsell, S. Chopra, and Y. LeCun, "Dimensionality reduction by learning an invariant mapping," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, vol. 2, 2006, pp. 1735–1742.

[5] C. Shorten and T. M. Khoshgoftaar, "A survey on image data augmentation for deep learning," *J. Big Data*, vol. 6, no. 1, pp. 1–48, 2019.

[6] J. Zhang and K. Ma, "Rethinking the augmentation module in contrastive learning: Learning hierarchical augmentation invariance with expanded views," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2022, pp. 16650–16659.

[7] J.-B. Grill et al., "Bootstrap your own latent-a new approach to self-supervised learning," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 33, 2020, pp. 21271–21284.

[8] X. Chen and K. He, "Exploring simple Siamese representation learning," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 15750–15758.

[9] K. He, H. Fan, Y. Wu, S. Xie, and R. Girshick, "Momentum contrast for unsupervised visual representation learning," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 9729–9738.

[10] X. Chen, S. Xie, and K. He, "An empirical study of training self-supervised vision transformers," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2021, pp. 9640–9649.

[11] Z. Wu, Y. Xiong, S. X. Yu, and D. Lin, "Unsupervised feature learning via non-parametric instance discrimination," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 3733–3742.

[12] T. Chen, S. Kornblith, M. Norouzi, and G. Hinton, "A simple framework for contrastive learning of visual representations," in *Proc. Int. Conf. Mach. Learn.*, 2020, pp. 1597–1607.

[13] O. Henaff, "Data-efficient image recognition with contrastive predictive coding," in *Proc. Int. Conf. Mach. Learn.*, 2020, pp. 4182–4192.

[14] Y. Tian, D. Krishnan, and P. Isola, "Contrastive multiview coding," in *Proc. 16th Eur. Conf. Comput. Vis.*, 2020, pp. 776–794.

[15] I. Misra and L. V. D. Maaten, "Self-supervised learning of pretext-invariant representations," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 6707–6717.

[16] A. V. D. Oord, Y. Li, and O. Vinyals, "Representation learning with contrastive predictive coding," 2018, *arXiv:1807.03748*.

[17] O. Russakovsky et al., "ImageNet large scale visual recognition challenge," *Int. J. Comput. Vis.*, vol. 115, pp. 211–252, Dec. 2015.

[18] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 2014, *arXiv:1409.1556*.

[19] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 770–778.

[20] A. Dosovitskiy et al., "An image is worth 16×16 words: Transformers for image recognition at scale," 2020, *arXiv:2010.11929*.

[21] Z. Liu et al., "Swin transformer: Hierarchical vision transformer using shifted windows," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2021, pp. 10012–10022.

[22] X. Zhang et al., "HiViT: Hierarchical vision transformer meets masked image modeling," 2022, *arXiv:2205.14949*.
[23] A. Tarvainen and H. Valpola, "Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 30, 2017, pp. 1195–1204.
[24] K. He, X. Chen, S. Xie, Y. Li, P. Dollár, and R. Girshick, "Masked autoencoders are scalable vision learners," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2022, pp. 16000–16009.
[25] M. Caron, P. Bojanowski, A. Joulin, and M. Douze, "Deep clustering for unsupervised learning of visual features," in *Proc. 15th Eur. Conf. Comput. Vis.*, 2018, pp. 1–30.
[26] M. Caron, I. Misra, J. Mairal, P. Goyal, P. Bojanowski, and A. Joulin, "Unsupervised learning of visual features by contrasting cluster assignments," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 33, 2020, pp. 9912–9924.
[27] Y. Tian et al., "Integrally pre-trained transformer pyramid networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2023, pp. 18610–18620.
[28] K. Tian, Y. Jiang, Q. Diao, C. Lin, L. Wang, and Z. Yuan, "Designing bert for convolutional networks: Sparse and hierarchical masked modeling," 2023, *arXiv:2301.03580*.
[29] Y. Liu, S. Zhang, J. Chen, Z. Yu, K. Chen, and D. Lin, "Improving pixel-based MIM by reducing wasted modeling capability," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2023, pp. 5361–5372.
[30] M. Zaharia, M. Chowdhury, M. J. Franklin, S. Shenker, and I. Stoica, "Spark: Clust. Comput. with working sets," in *Proc. IEEE Int. Conf. Cloud Comput. Technol. Sci.*, 2010, p. 10.
[31] T. Salimans, I. Goodfellow, W. Zaremba, V. Cheung, A. Radford, and X. Chen, "Improved techniques for training GANs," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 29, 2016, pp. 2234–2242.
[32] M. Noroozi and P. Favaro, "Unsupervised learning of visual representations by solving jigsaw puzzles," in *Proc. Eur. Conf. Comput. Vis.*, 2016, pp. 69–84.
[33] S. Gidaris, P. Singh, and N. Komodakis, "Unsupervised representation learning by predicting image rotations," 2018, *arXiv:1803.07728*.
[34] J. Robinson, C.-Y. Chuang, S. Sra, and S. Jegelka, "Contrastive learning with hard negative samples," 2020, *arXiv:2010.04592*.
[35] Y. Kalantidis, M. B. Sariyildiz, N. Pion, P. Weinzaepfel, and D. Larlus, "Hard negative mixing for contrastive learning," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 33, 2020, pp. 21798–21809.
[36] R. D. Hjelm et al., "Learning deep representations by mutual information estimation and maximization," 2018, *arXiv:1808.06670*.
[37] A. Radford et al., "Learning transferable visual models from natural language supervision," in *Proc. Int. Conf. Mach. Learn.*, 2021, pp. 8748–8763.
[38] L. Bottou, "Large-scale machine learning with stochastic gradient descent," in *Proc. COMPSTAT*, 2010, pp. 177–186. pp. 177–186.
[39] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 25, 2012, pp. 1097–1105.
[40] M. Li et al., "Guest editorial: Deep neural networks for graphs: Theory, models, algorithms, and applications," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 35, no. 4, pp. 4367–4372, Apr. 2024.
[41] M. Li, S. Zhou, Y. Chen, C. Huang, and Y. Jiang, "EduCross: Dual adversarial bipartite hypergraph learning for cross-modal retrieval in multimodal educational slides," *Inf. Fusion*, vol. 109, Sep. 2024, Art. no. 102428.
[42] M. Li, X. Zhuang, L. Bai, and W. Ding, "Multimodal graph learning based on 3D haar semi-tight framelet for student engagement prediction," *Inf. Fusion*, vol. 105, May 2024, Art. no. 102224.
[43] M. Li, L. Zhang, L. Cui, L. Bai, Z. Li, and X. Wu, "BLoG: Bootstrapped graph representation learning with local and global regularization for recommendation," *Pattern Recognit.*, vol. 144, Dec. 2023, Art. no. 109874.
[44] S. Zhang, X. Zhang, S. Wan, W. Ren, L. Zhao, and L. Shen, "Generative adversarial and self-supervised dehazing network," *IEEE Trans. Ind. Informat.*, vol. 19, no. 12, pp. 13530–13540, Oct. 2023.
[45] S. Zhang et al., "Semantic-aware dehazing network with adaptive feature fusion," *IEEE Trans. Cybern.*, vol. 53, no. 1, pp. 454–467, Jan. 2023.
[46] Y. Liu, Z. Yan, T. Ye, A. Wu, and Y. Li, "Single nighttime image dehazing based on unified variational decomposition model and multi-scale contrast enhancement," *Eng. Appl. Artif. Intell.*, vol. 116, Nov. 2022, Art. no. 105373.

Zheng Jiang received the B.S. degree in telecommunications engineering from the Nanjing University of Posts and Telecommunications, Nanjing, China, in 2022, where he is currently pursuing the M.S. degree in electronic information. His research interests include unsupervised representation learning in computer vision.

Zhou Zhou received the B.S. degree in electronic information engineering from Zhejiang Gongshang University, Hangzhou, China, in 2024. She is currently pursuing the M.S. degree in electronic information with the Nanjing University of Posts and Telecommunications. Her research interests include machine learning, object detection, and image generation.

Quan Zhou (Senior Member, IEEE) received the B.S. degree in electronics and information engineering from the China University of Geosciences, Wuhan, China, in 2002, and the M.S. and Ph.D. degrees in electronics and information engineering from the Huazhong University of Science and Technology, Wuhan, in 2006 and 2013, respectively.

He was a Visiting Scholar with Temple University, Philadelphia, PA, USA, from 2019 to 2020. He is currently a Full Professor at Nanjing University of Posts & Telecommunications, Nanjing, China. He also served as a Part-time Professor at Southeast University, Nanjing, China. He also held the position of Visiting Professor at the Shenyang Institute of Automation, Chinese Academy of Sciences, Shenyang, China, Temple University, Philadelphia, USA, and Kyushu Institute of Technology, Fukuoka, Japan He has authored or co-authored more than 100 related academic articles, including IEEE TRANSACTIONS ON IMAGE PROCESSING, IEEE TRANSACTIONS ON INTELLIGENT TRANSPORTATION SYSTEMS, IEEE TRANSACTIONS ON MULTIMEDIA, IEEE TRANSACTIONS ON MEDICAL IMAGING, IEEE TRANSACTIONS ON NEURAL NETWORKS AND LEARNING SYSTEMS, and *Pattern Recognition*. His research interests include deep learning, pattern recognition, and computer vision.

Dr. Zhou received the Most Influence Paper Award at IEEE ICIP2024 and the Best Paper Award at IEEE/SPIE ISAIR2024. He served as the Area Chair for the IEEE ICME2019, IEEE/SPIE ISAIR2019–2025, and PRCV2022–2025 and the Leading Guest Editor for IEEE TRANSACTIONS ON MULTIMEDIA, IEEE TRANSACTIONS ON FUZZY SYSTEMS, *Computers and Electrical Engineering*, and *Multimedia Tools and Applications*. He also serves as an Editor for *Pattern Recognition*.

Yongan Guo (Senior Member, IEEE) received the Ph.D. degree from the School of Communications and Information Engineering, Nanjing University of Posts and Telecommunications, China, where he is currently a Faculty Member with the Jiangsu Key Laboratory of Intelligent Information Processing and Communication Technology and the School of Communications and Information Engineering. His research focuses on deep reinforcement learning, edge computing, neural networks, and resource allocation in network environments, with significant contributions to graph convolutional networks and computational resource optimization.

Jing Yang (Member, IEEE) received the Ph.D. degree in mechanical and electronic engineering from Guizhou University, China, in 2020.

From August 2018 to September 2019, he was awarded a scholarship by the China Scholarship Council under the State Scholarship Fund to pursue his study with Oklahoma State University as a joint Ph.D. student with the School of Computer Science and Technology, where he joined Guoliang Fan's Group. From October 2022 to October 2023, he was a Visiting Scholar studying in the team of Prof. Guo Minyi (IEEE Fellow) with Shanghai Jiao Tong University. He is currently an Assistant Professor with the State Key Laboratory of Public Big Data, Guizhou University. He has published more than 50 peer-reviewed papers in the related area, including well-archived international journals, such as IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY, IEEE TRANSACTIONS ON GREEN COMMUNICATIONS AND NETWORKING, IEEE TRANSACTIONS ON INTELLIGENT TRANSPORTATION SYSTEMS, and IEEE INTERNET OF THINGS JOURNAL. His research interests include high-performance computing, task scheduling in various architectures, and open-domain visual learning. He serves more than 20 prestigious international journals and conferences, such as IEEE TRANSACTIONS ON COMPUTERS, IEEE TRANSACTIONS ON MOBILE COMPUTING, IEEE TRANSACTIONS ON INDUSTRIAL INFORMATICS, IEEE TRANSACTIONS ON INSTRUMENTATION AND MEASUREMENT, IEEE TRANSACTIONS ON SERVICES COMPUTING, IEEE TRANSACTIONS ON SEMICONDUCTOR MANUFACTURING, IEEE TRANSACTIONS ON ARTIFICIAL INTELLIGENCE, IEEE TRANSACTIONS ON AUTOMATION SCIENCE AND ENGINEERING, IEEE TRANSACTIONS ON INFORMATION THEORY, IEEE TRANSACTIONS ON CONSUMER ELECTRONICS, IEEE TRANSACTIONS ON MULTIMEDIA, and IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY.

Weihua Ou (Member, IEEE) received the Ph.D. degree in information and communication engineering from the Huazhong University of Science and Technology, Wuhan, China. He is currently a Full Professor with the School of Big data and Computer Science, Guizhou Normal University, Guiyang, China. His research results have published more 80 papers at prominent journals and conferences, such as IEEE TRANSACTIONS ON NEURAL NETWORKS AND LEARNING SYSTEMS, IEEE TRANSACTIONS ON MULTIMEDIA, IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY, PR, ICPR, and ICME. His research interests include graph learning, LLM, and multimodal machine learning. His publications have been cited in Google Scholar more than 1800 times, his H-index is 23.