Instance Similarity Learning for Unsupervised Feature Representation

Ziwei Wang[1,2,3], Yunsong Wang[1], Ziyi Wu[1], Jiwen Lu[1,2,3]*, Jie Zhou[1,2,3]
[1] Department of Automation, Tsinghua University, China
[2] State Key Lab of Intelligent Technologies and Systems, China
[3] Beijing National Research Center for Information Science and Technology, China
{wang-zw18, wangys16}@mails.tsinghua.edu.cn; dazitu616@gmail.com;
{lujiwen,jzhou}@tsinghua.edu.cn

Abstract

In this paper, we propose an instance similarity learning (ISL) method for unsupervised feature representation. Conventional methods assign close instance pairs in the feature space with high similarity, which usually leads to wrong pairwise relationship for large neighborhoods because the Euclidean distance fails to depict the true semantic similarity on the feature manifold. On the contrary, our method mines the feature manifold in an unsupervised manner, through which the semantic similarity among instances is learned in order to obtain discriminative representations. Specifically, we employ the Generative Adversarial Networks (GAN) to mine the underlying feature manifold, where the generated features are applied as the proxies to progressively explore the feature manifold so that the semantic similarity among instances is acquired as reliable pseudo supervision. Extensive experiments on image classification demonstrate the superiority of our method compared with the state-of-the-art methods. The code is available at https://github.com/ZiweiWangTHU/ISL.git.

1. Introduction

Deep neural networks have achieved the state-of-the-art performance in various vision applications such as face recognition [7, 45, 34], object detection [41, 33, 30], image retrieval [14, 42, 32] and many others. However, most successful deep neural networks are trained with strong supervision, which requires a large amount of labeled data with expensive annotation cost and strictly limits the deployment of deep models. Hence, it is desirable to train deep neural networks with only the unlabeled data while achieving comparable performance with supervised learning.

To enable deep neural networks to learn from the unlabeled data, unsupervised learning methods have been widely studied recently. The clustering methods [24, 47, 3] shown in the first column of Figure 1 provide pseudo labels to train the networks according to the cluster indexes, which are error-prone due to the complex inter-class boundaries. The instance specificity analysis methods [46, 2, 38, 16, 21] depicted in the second column of Figure 1 regard every single sample as an independent class to avoid clustering. However, the offered supervision is ambiguous and results in weak class discrimination. Meanwhile, designing pretext tasks with self-supervised learning [8, 51, 44] shares the same limitations of instance specificity analysis methods due to the discrepancy between the auxiliary supervision and the target task. In order to mitigates the disadvantages of clustering and instance specificity analysis, neighbor-

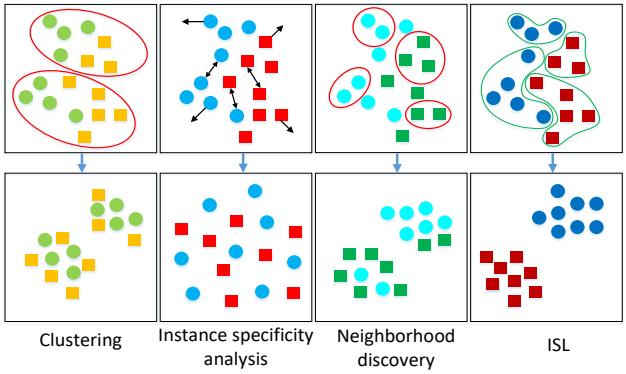


Figure 1. The difference among clustering methods, instance specificity analysis methods, neighborhood discovery methods and our method. The clustering methods are error-prone because of the complicated inter-class boundaries, and the instance specificity analysis methods are weakly discriminative due to the ambiguous supervision that treats each sample as an independent class. Meanwhile, the neighborhood discovery methods regard the instances close to the anchor as similar samples and fails to depict the true semantic similarity in large neighborhoods on the feature manifold. On the contrary, we mine the feature manifold and learn the instance-to-instance relationship with reliable semantic similarity, so that informative features can be obtained.

*Corresponding author

hood discovery methods [22, 54, 23] have been proposed, which explore the local neighbors progressively with class consistency maximization by mining instance-to-instance correlation. They simply assign high similarity to pairs that have short Euclidean distance in the feature space. While the representations lie in the implicit feature manifold that is continuous in the Euclidean space, the Euclidean distance only reveals the true semantic similarity in extremely small neighborhoods and fails to provide the informative pseudo supervision for large neighborhoods due to the inconsistency with the distance measured on the feature manifold. As a result, the feature discriminality is still limited as shown in the third column of Figure 1.

In this paper, we present an ISL method to learn the semantic similarity among instances for unsupervised feature representation. Unlike the conventional methods that assign high similarity to close pairs according to the Euclidean distance in the feature space, our method mines the feature manifold in an unsupervised manner and learns the semantic similarity among different samples, so that the reliable instance-to-instance relationship in large neighborhoods is applied to supervise the representation learning models as demonstrated in the last column of Figure 1. More specifically, we employ the Generative Adversarial Network (GAN) [13] to mine the underlying feature manifold, and Figure 2 depicts the overall pipeline of the proposed method. The generator yields the proxy feature that mines positives for each anchor instance based on the sampled triplet, and the discriminator predicts the confidence score that the generated proxy is semantically similar with the mined pseudo positive samples. Since the Euclidean distance reveals the sample similarity in small neighborhoods, the instances near the proxy feature with high confidence score are added to the positive sample set of the given anchor. In order to explore richer instance-wise relation and exploit the semantics of the mined positive sample set simultaneously, the generated proxy is enforced to be similar with negative instances and the mined pseudo positive samples during the training process of GANs. With the reliable pseudo supervision, we employ the contrastive loss with hard positive enhancement to learn discriminative features. Extensive experiments on CIFAR-10 [28], CIFAR-100 [28], SVHN [36] and ImageNet [6] datasets for image classification demonstrate that the proposed ISL outperforms most of the existing unsupervised learning methods. Moreover, our ISL can be integrated with state-of-the-art unsupervised features to further enhance the performance.

2. Related Work

Unsupervised learning has aroused extensive interests because it enables models to be trained by vast unlabeled data and saves expensive annotation cost. Existing methods can be divided into five categories: clustering, instance specificity analysis, neighborhood discovery, self-supervised learning and generative models.

Clustering: Clustering methods [3, 47, 24, 48] employ cluster indexes as pseudo labels to train the end-to-end unsupervised learning model. Caron *et al*. [3] jointly learned the network parameters and the cluster assignment of features, where k-means was applied for iterative data grouping. Furthermore, Yang *et al*. [48] applied stacked autoencoders [43] to provide stronger supervision by minimizing the image reconstruction loss despite of the cluster assignment. However, the clustering methods are error-prone as they fail to represent the highly complex class boundaries.

Instance specificity analysis: Instance specificity analysis [46, 2, 38, 1, 16, 21, 4, 50, 18, 49] methods consider every single instance as an independent class, and only take the sample and its transformed instance as positive pairs with the assumption that the instance semantic similarity is automatically discovered with the instance-wise supervision. Wu *et al*. [46] proposed the noise-contrastive estimation (NCE) to approximate the full softmax distribution in order to reduce the complexity of the instance-wise classifier, and utilized a memory bank to store the instance feature. He *et al*. [16] built dynamic dictionary on-the-fly that facilitated largescale contrastive learning. Chen *et al*. [4] composed various data augmentation techniques with an extra non-linear transformation to learn discriminative unsupervised features. However, the learned class boundaries are ambiguous in instance specificity analysis methods as they may push away samples with the same class label and increase the intra-class variance.

Neighborhood discovery: The neighborhood discovery methods [22, 23, 54] mitigate the drawbacks of the above two kinds of methods by progressively mining instance-to-instance correlation with class consistency maximization. Huang *et al*. [22] iteratively enlarged the neighborhood for each instance by comparing its cosine similarity with different samples in the curriculum learning setting, and treated all neighbors as positive instances. Zhuang *et al*. [54] presented a metric for local aggregation, where similar samples were encouraged to move together and vice versa. Nevertheless, existing neighborhood discovery methods simply assign the similarity based on the Euclidean distance of their features to train the representation learning model, which fails to demonstrate the semantic similarity on the underlying feature manifold for large neighborhoods.

Self-supervised learning: self-supervised learning methods [8, 51, 44, 37, 39, 25, 35, 11] usually design pretext tasks to provide the hand-crafted auxiliary supervision with human priors, where the assumption is that the semantics learned via the auxiliary supervision can be transferred to the downstream tasks such as image classification and object detection. Doersch *et al*. [8] and Noroozi *et al*. [37] sampled patches on a image and designed the jigsaw puz-

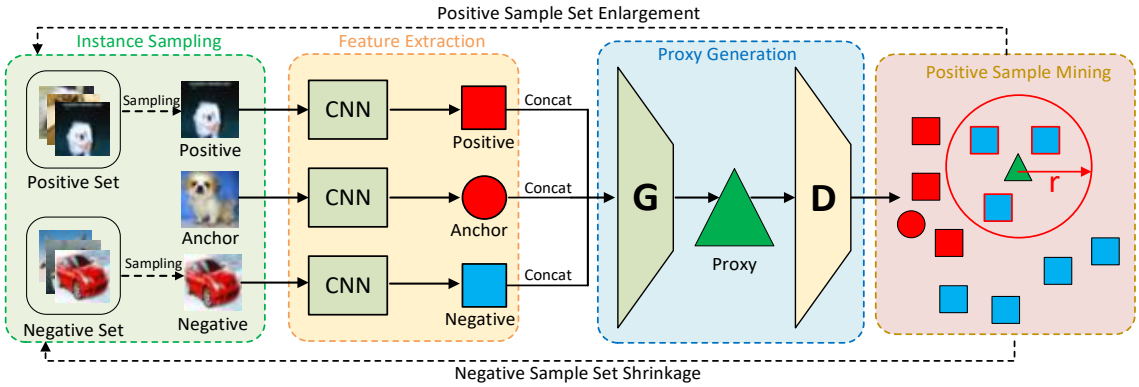


Figure 2. The pipeline of the instance similarity learning. For a given anchor, we first sample triplets from the mined positive set and the negative set, and then obtain the features via the convolutional neural networks. After concatenating the features of the anchor, the positive and the negative samples, we generate the proxy for feature manifold mining by the generator. The instances in the neighborhood of the proxy are removed from the negative set and added to the positive set if the proxy is semantically similar to the anchor, where the semantics similarity is predicted by the discriminator.

zles, where the networks were designed to predict the relative position of two patches. Pathak *et al*. [39] used the context-based pixel prediction as the pretext task, and the masked contents in an image should be generated by the context encoders with reconstruction and adversarial loss. However, the self-supervised learning methods share the same limitations with the instance specificity analysis methods in unsupervised learning due to the large discrepancy between the pretext tasks and the downstream applications.

Generative Models: Generative models [43, 31, 27, 40, 20, 13, 10] including RBM [20], AutoEncoders [27] and GAN [13] have been widely studied recently since it is able to learn the data distribution by reconstructing the input samples without supervision. Radford *et al*. [40] and Donahue *et al*. [9] applied the GANs to extract representations that generated samples semantically similar to the input. Learning representations directly with generative models leads to weak class discriminality due to the difference between the reconstruction and downstream tasks.

3. Approach

In this section, we first introduce feature manifold mining via GANs, and then present the instance semantic similarity learning on the mined feature manifold. Finally, we propose effective training objective with the learned semantic similarity to obtain discriminative representations.

3.1. Feature Manifold Mining

Let $X = \{\boldsymbol{x}_1, \boldsymbol{x}_2, ..., \boldsymbol{x}_N\}$ and $F = \{\boldsymbol{f}_1, \boldsymbol{f}_2, ..., \boldsymbol{f}_N\}$ be the input images and their features respectively, where N is the number of instances. $S \in \{0, 1\}^{N \times N}$ is the similarity matrix, where the element in the i_{th} row and j_{th} column s_{ij} equals to one if $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ are semantically similar (positive) and zero otherwise (negative). Conventional unsupervised methods treat pairs with short Euclidean distance in the feature space as similar ones. However, the Euclidean distance only reveals the similarity in extremely small neighborhoods and usually fails to depict the true semantic similarity in large neighborhoods due to the mismatch between the geodesic distance on the feature manifold and the Euclidean distance. As a result, samples with dissimilar semantics are regarded as similar pairs for pseudo supervision to train the representation model and vice versa, which leads to uninformative features in unsupervised learning. Because the implicit feature manifold changes during the training process of the feature extraction model, we employ GANs to dynamically mine the feature manifold according to the feature distribution.

In order to evaluate feature distribution, we sample the triplets $\{\boldsymbol{f}_i^a, \boldsymbol{f}_i^p, \boldsymbol{f}_i^n\}$ in the feature space according to the similarity matrix S, where $\boldsymbol{f}_i^a$, $\boldsymbol{f}_i^p$ and $\boldsymbol{f}_i^n$ are the features of the anchor, the positive sample and the negative sample in the i_{th} triplet respectively. For initialization, S is set to be the identity matrix at the beginning of training, which means that all instances are only semantically similar with themselves. The proxy generator G generates the proxy feature $\boldsymbol{f}_i^g$ for the i_{th} triplet that is used to explore the feature manifold by mining positives for the given anchor and modify the similarity matrix dynamically. Aiming to explore richer instance-wise relation and exploit the semantics of the mined positive sample set simultaneously, we expect the proxy feature $\boldsymbol{f}_i^g$ to have the two following properties:

(1) The proxy feature should be semantically similar with negative samples in the triplet. At the beginning of training process, the positive sample in the triplet is identical with the anchor, where the rich instance-wise relation is not explored for discriminative representation learning. In order to enlarge the positive sample set for more informative supervision, enforcing the proxy to be semantically similar to negatives enables active

feature manifold exploration.

(2) The proxy feature should also be semantically similar to positive samples with the goal of exploiting the semantics from mined positive sets, so that the feature manifold is learned with high precision.

We employ a discriminator D to measure the semantic similarity between the proxy and the positives or negatives. D should accurately classify the real triplet $\mathcal{T}_r = \{\boldsymbol{f}_i, \boldsymbol{f}_i^p, \boldsymbol{f}_i^n\}$ sampled from the mined sets and synthetic triplet $\mathcal{T}_s^n = \{\boldsymbol{f}_i, \boldsymbol{f}_i^p, \boldsymbol{f}_i^g\}$ with the generated proxy as the negative. Meanwhile, the real triplet should also be distinguished by D from the synthetic triplet $\mathcal{T}_s^p = \{\boldsymbol{f}_i, \boldsymbol{f}_i^g, \boldsymbol{f}_i^n\}$ with the generated proxy as the positive. Following the adversarial loss in [13], we design the following objective to train the generator and discriminator, and obtain the proxy feature similar to both positive and negative samples:

$$\min_G \max_D \mathcal{L}_{gan} = \log D(\mathcal{T}_r) + \log(1 - D(\mathcal{T}_s^p)) + \alpha \log(1 - D(\mathcal{T}_s^n)) \quad (1)$$

where $\boldsymbol{f}_i^g$ in $\mathcal{T}_s^n$ and $\mathcal{T}_s^p$ is generated by G based on the real triplet $\mathcal{T}_r$ and is denoted as $\boldsymbol{f}_i^g = G(\mathcal{T}_r)$. $D(\mathcal{T})$ represents the confidence score that the input triplet $\mathcal{T}$ is real, which is predicted by the discriminator. α is a hyperparameter that balances the hardness of the generated proxy feature to be recognized as positive samples. When α increases, the generated proxy $\boldsymbol{f}_i^g$ is forced to be more similar to the negative sample and is harder to be recognized as the positive instance, which means the proxy explores the feature manifold more aggressively. When finishing the training of GANs, the generator G learns the underlying feature manifold and is able to generate the reliable proxy to enlarge the positive sample set for the given anchor.

3.2. Instance Similarity Learning

In this section, we first briefly introduce the hand-crafted instance similarity assignment in conventional methods that utilize the Euclidean distance among features to measure the similarity, and then detail the instance similarity learning with the mined feature manifold in our method. In conventional methods [22, 23, 54], the neighborhood $\mathcal{N}(\boldsymbol{x})$ is identified by k-nearest neighbors for a given anchor $\boldsymbol{x}$ in the following form:

$$\mathcal{N}(\boldsymbol{x}) = \{\boldsymbol{x}_i | d(\boldsymbol{x}_i, \boldsymbol{x}) \text{ is ranked the bottom } k \text{ for all } i\}$$

where $d(\boldsymbol{x}, \boldsymbol{y})$ means the distance between two feature vectors $\boldsymbol{x}$ and $\boldsymbol{y}$, and the Euclidean distance is usually applied. k is a hyperparameter that decides the size of the neighborhood, and instances in the neighborhood are all treated as similar samples. Since the Euclidean distance can only reveal the true semantic similarity in extremely small neighborhoods and fails to provide informative pseudo supervision in large neighborhoods, k is usually limited to be very small and the class discrimination is weak due to the constrained size of the positive sets.

Our method employs the generated proxy $\boldsymbol{f}_i^g$ to mine the semantically similar instances with the anchor feature $\boldsymbol{f}_i$ in order to enlarge the positive sample set $\mathcal{P}_i$, where $\mathcal{P}_i$ is initialized with the anchor itself. Since the generator G learns the underlying feature manifold according to the feature distribution, the generated proxy $\boldsymbol{f}_i^g$ is utilized to mine semantically similar instances for the given anchor and move the semantically similar samples from the negative set to enlarge the positive one. Because the confidence score of the synthetic triplet $D(\mathcal{T}_s^p)$ evaluates the sematic similarity between the generated feature and the mined positives, it represents the reliability of the proxy for positive sample set enlargement. When the confidence score $D(\mathcal{T}_s^p)$ of the generated proxy is high, the proxy mines the reliable region in which the instances are removed from the negative sample set and added to the positive one. We employ the following strategy to enlarge the positive sample set $\mathcal{P}_i$ for a given anchor $\boldsymbol{f}_i$ with the instance $\boldsymbol{f}_j$:

$$\boldsymbol{f}_j = \{\boldsymbol{f}_j \big| ||\boldsymbol{f}_i^g - \boldsymbol{f}_j||_F < r,\ D(\mathcal{T}_s^p) > h\} \quad (2)$$

where $||\cdot||_F$ means the Frobenius norm and r is a hyperparameter to control the size of the region for positive sample set enlargement. h is the threshold to trigger positive sample addition. Since the feature manifold is continuous in the feature space, the Euclidean distance can reveal the semantic similarity in extremely small neighborhoods. As a result, instances in the small hyperspherical neighborhoods of the proxy can be treated as semantically similar samples with the proxy feature, which share consistent semantics with the anchor for positive sample set enlargement.

Because the generated proxy $\boldsymbol{f}_i^g$ is influenced by the sampled real triplet $\mathcal{T}_r$ input to the generator G, we sample the real triplets of the given anchor for multiple times to gain more information about the distribution of the positives and negatives. We denote the optimal proxy as $\boldsymbol{f}_i^{g*}$ with the definition in the following:

$$\boldsymbol{f}_i^{g*} = \arg\max_{\boldsymbol{f}_i^g} D(\mathcal{T}_s^p) \quad (3)$$

We utilize the optimal proxy among all generated proxy features to enlarge the positive sample set via (2). The pseudo supervision provided by instance similarity learning is informative as it sets the instances with short geodesic distance on the mined feature manifold to be positive, and maximizing the similarity between their features can significantly enhance the feature informativeness on downstream tasks such as image classification and object detection.

3.3. Learning Representations with the Mined Instance Similarity

The learned instance similarity can provide effective supervision for unsupervised feature representation, where the

semantic similar pairs should be constrained to be close in the feature space and vice versa. Following non-parametric loss in [46], we illustrate the similarity by the probability distributions p_{ij} that two samples $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ come from the same class:

$$p_{ij} = \frac{\exp(\boldsymbol{f}_i^T \boldsymbol{f}_j / \tau)}{\sum_{k=1}^{N} \exp(\boldsymbol{f}_i^T \boldsymbol{f}_k / \tau)} \tag{4}$$

where τ is the hyperparameter for the temperature that controls the concentration of the distribution [19]. Since we argue that all semantically similar instances in the positive sample set $\mathcal{P}_i$ for a given anchor share the same class label, we propose the following objective that maximizes the log-likelihood of the probability that all instances in the positive sample set come from the same class:

$$\mathcal{L}_1 = -\sum_{i=1}^{N} \log\Big(\sum_{\boldsymbol{f}_k \in \mathcal{P}_i} p_{ik}\Big) \tag{5}$$

The objective is to encourage the label consistency between the anchor and all of its positive samples, so that the more informative pseudo supervision for representation learning is provided.

As demonstrated in [23], the less semantically similar instances in the positive sample set can be overwhelmed during training because of the small quantity. However, the hard positives provide large gradients and contribute significantly to the training process [52, 53, 15]. As a result, we apply the hard positive enhancement (HPE) strategy demonstrated in [23] to further enhance the performance. We define the positive sample $\boldsymbol{f}_j$ with smallest p_{ij} w.r.t. the anchor $\boldsymbol{f}_i$ as the hard positive. For initialized positive sample set, the feature of a randomly transformed variant of the anchor image $\boldsymbol{x}_i$ is regarded as hard positive. Denoting the hard positive of the anchor $\boldsymbol{f}_i$ as $\boldsymbol{f}_i^{hard}$, we employ the following loss to integrate the hard positive enhancement strategy with our method:

$$\mathcal{L}_2 = \sum_{i=1}^{N} \sum_{k=1}^{N} p_{ik} \log \frac{p_{ik}}{p_{ik}^{hard}} \tag{6}$$

where N is the number of samples in the dataset, and p_{ik}^{hard} demonstrates the probability that the instance $\boldsymbol{f}_k$ and the hard positive $\boldsymbol{f}_i^{hard}$ of the anchor $\boldsymbol{f}_i$ comes from the same class. The loss for hard positive enhancement significantly magnifies the influence of hard positives during training, which leads to discriminative boundaries among classes in the feature space. The overall loss for our ISL is written as follows:

$$\mathcal{L} = \mathcal{L}_1 + \lambda \mathcal{L}_2 \tag{7}$$

where λ is a hyperparameter that balances the importance of two loss terms. For fair comparison with the state-of-the-art methods, we conducted experiments in the settings with and without the hard positive enhancement strategy. Following [46], we maintain an offline memory bank to avoid intractable loss computations for all the instances by storing feature vectors in the memory. We initialize the memory bank with random vectors and update the memory features $\hat{\boldsymbol{f}}_i$ by mixing the memory features and the learned up-to-date features $\boldsymbol{f}_i$:

$$\hat{\boldsymbol{f}}_i = \eta \boldsymbol{f}_i + (1 - \eta) \hat{\boldsymbol{f}}_i \tag{8}$$

where $\eta \in [0, 1]$ is a hyperparameter that illustrates the importance of up-to-date features during the process of memory update.

4. Experiments

In this section, we first describe the datasets and our implementation details briefly. Then we demonstrate our intuitive logic by toy examples, and conducted the ablation study to investigate the impact of different components in the presented instance similarity learning. Finally, we compare our ISL with the state-of-the-art unsupervised feature learning methods on image classification. The implementation details and the results on other tasks such as object detection and transfer learning are shown in the supplementary material.

4.1. Datasets and Implementation Details

We first detail the datasets that we carried out experiments on: The CIFAR-10 dataset consists of $60,000$ images from 10 classes with $50,000$ images for training and $10,000$ for evaluation. The CIFAR-100 dataset has the same data split with CIFAR-10, and the only difference is the images consist of 100 classes with 600 images for each. The Street View House Numbers (SVHN) dataset contains 10 classes of digit images with $73,257$ of them for training and $26,032$ of them for evaluation. The ImageNet dataset consists of about 1.2 million and $50k$ images from $1,000$ classes for training and validation respectively.

We employed the top-1 accuracy to evaluate ISL on image classification. Following the experiment settings in [46], we tested two classifiers including Linear Classifier (LC) and Weighted kNN to evaluate the features extracted in different layers. We applied a fully-connected layer as the LC, which was trained by the cross-entropy loss. The weighted kNN classifier infers the class label for the feature $\boldsymbol{f}$ by the votes of the top-k neighbors. For each neighbor $\boldsymbol{f}_i$, the weight is assigned to be $\exp(\boldsymbol{f}_i^T \boldsymbol{f} / \tau)$. We set $k = 200$ and $\tau = 0.07$ in our experiments. We trained our ISL with the architectures of the AlexNet [29], ResNet18 and ResNet50 [17].

We iteratively trained GANs that mined the feature manifold, learned the semantic similarity among instances and optimized the backbone that extracted unsupervised features of images with 4 rounds in total. In the training of

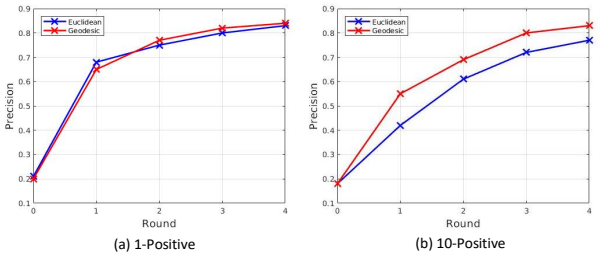


Figure 3. The average precision of mined positive samples w.r.t. different rounds during training for positive sample set size of 1 and 10.

GANs, we sampled five triplets for a given anchor in order to decrease the discrepancy between the sampled feature distribution and the real feature distribution so that the feature manifold could be mined precisely. We leveraged three fully-connected layers as the generator and another three-layer fully-connected networks as the discriminator. In each round, we trained GANs until the loss of the generator converged. The hyperparameter α was set to 1. We used the Adam optimizer [26] with fixed learning rate $1e$-4 to train both the generator and discriminator.

In the training of the backbone networks, the number of training epochs in each round was 200, 200, 100 and 100 for experiments on CIFAR-10, CIFAR-100, SVHN and ImageNet respectively. Following [46], we adopted the SGD optimizer with momentum at 0.9. The learning rate was initially set to 0.03 and decayed twice by multiplying 0.1 at the 75% and 90% epoch of the total epochs. We used a batchsize of 256 for ImageNet and 128 for others. The feature was normalized and the length was fixed to 128 in most experiments. The hyperparameter η, τ and λ were set as 0.5, 0.07 and 0.5 respectively.

For instance similarity learning that enlarges the positive sample set, we sampled five triplets for a given anchor to generate the optimal proxy feature with enhanced reliability. The hyperparameters h and r were set to 0.5 and 1.

4.2. Performance Analysis

In this section, we first demonstrate the intuitive logic of our instance similarity learning by toy examples, and show the influence of different components in the proposed techniques by ablation study.

4.2.1 Toy Examples

While Euclidean distance fails to reveal the true semantic similarity for samples in large neighborhoods, the thought of the presented ISL is learning the instance similarity in the feature manifold to provide informative pseudo supervision for unsupervised feature representation. We conducted simple experiments on CIFAR-10 with AlexNet to show our thoughts with intuition.

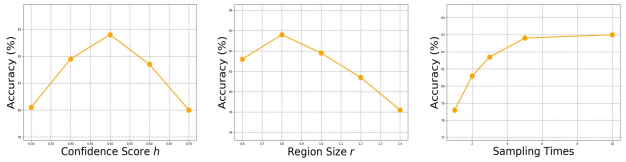
We show the average precision of the positive sample sets across all anchors, where the precision is defined as the ratio of mined pseudo positives from the anchor class. Figure 3 demonstrates the precision of mined pseudo positives across anchors w.r.t. different epochs during training, where the positive sample set size was 1 and 10. The geodesic distance applied in our ISL is compared with the Euclidean distance leveraged in conventional neighborhood discovery methods, and the latter chose the closest samples to be positive. Both distance measure achieves similar precision for the positive sample set in size of 1, while geodesic distance significantly surpasses Euclidean distance for the positive sample set in size of 10 since the former reveals the true semantic similarity in large neighborhoods.

4.2.2 Ablation Study

Leveraging the Euclidean distance among features as the supervision only reveals the semantic similarity in extremely small neighborhoods and fails to provide informative pseudo supervision for representation learning. On the contrary, our instance similarity learning illustrates geodesic distance on the mined feature manifold that demonstrates the reliable instance-to-instance relationship. In order to investigate the effectiveness of the proposed instance similarity learning and the impact of the critical hyperparameters, we conducted ablation study w.r.t. the confidence score threshold h in instance similarity learning, the region size r for positive sample set enlargement and the sampling times to generate each proxy for positive sample set enlargement. We adopted the AlexNet architecture as the backbone and trained our ISL on the CIFAR-10 dataset in the ablation study. The kNN classification accuracy is reported for evaluation, which is shown in Fig. 4.

Performance w.r.t. the confidence score threshold h: In instance similarity learning, the generated proxy is applied to enlarge the positive sample set using the surrounding instances when the confidence score is larger than the threshold h. Increasing h reduces the mined positives for the given anchor because the proxy is required to be more confident in positive sample set enlargement and vice versa. The impact of h on the performance is illustrated in Fig. 4(a), where medium threshold achieves the best performance. The low threshold is not able to guarantee the reliability of the generated proxy and the high threshold fails to provide sufficient proxies for positive sample set enlargement, where both of them degrade the accuracy.

Performance w.r.t. the region size r: In positive sample set enlargement, instances whose Euclidean distance from the proxy is less than r are assigned to be the positives for the given anchor. Larger r represents that more instances are added to the positive sample set for each generated proxy, and assumes that the Euclidean distance can better approximate the geodesic distance on the feature manifold in larger neighborhoods. Fig. 4(b) demonstrates the performance versus different r, and medium r enlarges the pos-



(a) (b) (c)

Figure 4. Classification accuracy on the CIFAR-10 dataset w.r.t. (a) the confidence score threshold h in instance similarity learning and (b) the region size r and (c) the sampling times to generate each proxy for positive sample set enlargement.

itive sample set with sufficient reliable instances. Large r adds unreliable instances to the positive sample set because the Euclidean distance cannot reveal the true semantic similarity in large neighborhoods. On the contrary, insufficient instances are added to the positive sample set for small r, so that the samples with similar semantics are pushed away and the class boundaries of features become ambiguous.

Performance w.r.t. the sampling times to generate the proxy: The generator G generates the proxy feature according to the anchor, the distribution of its positive samples and negative samples. In order to provide accurate information about the distribution, we sampled the triplets for multiple times so that the more reliable proxy could be generated. The performance w.r.t. different sampling times is illustrated in Fig. 4(c), where the classification accuracy increases when the triplets are sampled for more times. However, the improvements become very incremental when the sampling time is larger than five, while the computational cost during the training stage increases significantly. To balance the efficiency and the effectiveness, we sampled five triplets to generate reliable proxies in most experiments.

4.3. Comparison with the State-of-the-art Methods

In this section, we compare the proposed ISL with the state-of-the-art unsupervised representation learning methods including the clustering method DeepCluster[3], the instance specificity analysis methods Instance [46], MoCo-v1 [16] and MoCo-v2 [5], self-supervised methods RotNet [12] and the neighborhood discovery methods AND [22], LA [54], PAD [23]. Meanwhile, the baselines of random features are provided for reference. We demonstrate the top-1 accuracy on CIFAR-10, CIFAR-100, SVHN and ImageNet.

For the experiments on CIFAR-10, CIFAR-100 and SVHN, we utilized AlexNet, ResNet18 and ResNet50 as the backbone network to evaluate the proposed ISL. We tested two classification models, the weighted kNN with the FC features and the linear classifier using the Conv5 features. Table 1 demonstrates the results. All the unsupervised learning methods outperform the random features by a sizable margin, which clearly shows the effectiveness. Except for PAD, other existing methods did not apply hard positive enhancement (HPE) strategy in unsupervised representation learning. As the hard positive enhancement (HPE) strategy also increases the accuracy of the learned representation, we also tested our ISL without HPE on the three datasets to evaluate the benefit brought only by the instance similarity learning, which is denoted as ISL w/o HPE in Table 1. Compared with existing unsupervised features, our ISL archives higher accuracy on all three datasets with the two classification models in most cases.

Table 1. Classification accuracy (%) on CIFAR-10, CIFAR-100 and SVHN, where the architecture of AlexNet, ResNet18 and ResNet50 were applied as the backbone networks. The results of two classification models are reported: the weighted kNN with the FC features and the linear classifier using the Conv5 features. ISL w/o HPE means our method without hard positive enhancement.

	Dataset	CIFAR10	CIFAR100	SVHN
Architecture	Classifier/Feat.	Weighted kNN / FC		
AlexNet	Random	34.5	12.1	56.8
	DeepCluster	62.3	22.7	84.9
	RotNet	72.5	32.1	77.5
	Instance	60.3	32.7	79.8
	AND	74.8	41.5	90.9
	ISL w/o HPE	**81.1**	**49.2**	**91.0**
	PAD	81.5	48.7	91.2
	ISL	**82.8**	**50.3**	**91.8**
ResNet18	Instance	80.8	40.1	92.6
	AND	86.3	48.1	93.1
	ISL w/o HPE	**87.0**	**52.1**	**93.9**
	ISL	**87.8**	**54.7**	**94.2**
ResNet50	Instance	81.8	42.3	92.9
	AND	87.6	49.0	93.2
	ISL w/o HPE	**88.3**	**56.7**	**94.0**
	ISL	**88.9**	**58.1**	**94.5**
Architecture	Classifier/Feat.	Linear Classifier / conv5		
AlexNet	Random	67.3	32.7	79.2
	DeepCluster	77.9	41.9	92.0
	RotNet	**84.1**	57.4	92.3
	Instance	70.1	39.4	89.3
	AND	77.6	47.9	**93.7**
	ISL w/o HPE	83.5	**58.5**	93.3
	PAD	84.7	58.6	93.2
	ISL	**85.8**	**60.1**	**93.9**
ResNet18	Instance	84.1	48.9	94.0
	AND	88.9	57.4	94.3
	ISL w/o HPE	**89.2**	**61.1**	**94.4**
	ISL	**90.7**	**63.5**	**94.5**
ResNet50	Instance	85.0	50.1	94.4
	AND	90.2	58.5	94.9
	ISL w/o HPE	**91.0**	**63.0**	**94.9**
	ISL	**91.5**	**65.9**	**95.2**

For experiments on ImageNet, AlexNet, ResNet18 and ResNet50 were applied as the backbone in our ISL. Despite of the kNN classification model with FC features was used for evaluation, the features from the Conv1 to Conv5 layers were also utilized to test our model as shown in Table 2. The methods with the marker $*$ set the feature dimension as $2,048$. Due to the local aggregation metric that automatically pushes away dissimilar samples and pulls together

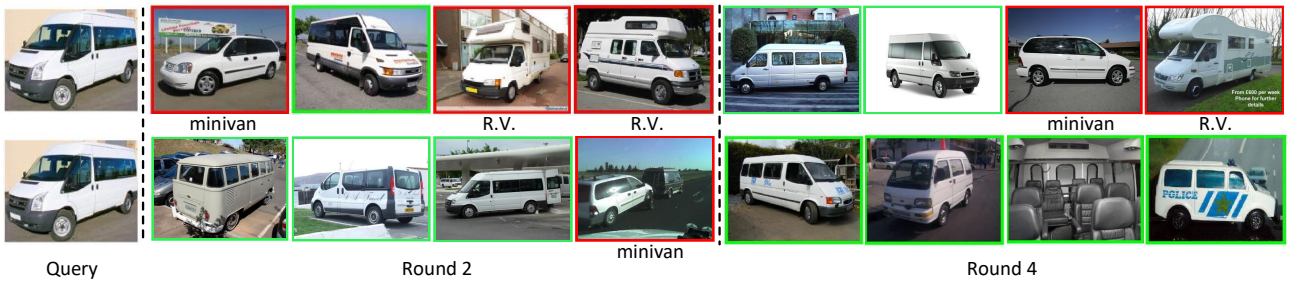


Figure 5. An example of positive sample mining via LA (top row) and our ISL (bottom row) in different rounds during training. The query image is from the minibus class. The images with green boxes represent the positives mined correctly and those with red boxes mean the images from other classes. More examples are visualized in the supplementary material.

Table 2. Comparison of top-1 accuracy (%) on ImageNet with architectures of AlexNet, ResNet18 and ResNet50. The results of two classification models are reported: the weighted kNN with the FC features and the linear classifier using the Conv1-Conv5 features.

Classifier	Linear Classifier					kNN
Feature	conv1	conv2	conv3	conv4	conv5	FC
AlexNet						
Random	11.6	17.1	16.9	16.3	14.1	3.5
DeepCluster	13.4	32.3	**41.0**	39.6	38.2	26.8
RotNet	**18.8**	31.7	38.7	38.2	36.5	9.2
Instance	16.8	26.5	31.8	34.1	35.6	31.3
AND	15.6	27.0	35.9	39.7	37.9	31.3
PAD	-	-	-	-	38.6	35.1
LA	18.7	**32.7**	38.1	42.3	42.4	38.1
ISL	17.3	29.0	38.4	**43.3**	**43.5**	**38.9**
ResNet18						
DeepCluster	**16.4**	17.2	28.7	44.3	49.1	–
Instance	16.0	**19.9**	29.8	39.0	44.5	41.0
LA	9.1	18.7	**34.8**	48.4	52.8	45.0
ISL	15.3	19.1	32.7	**49.1**	**54.0**	**46.1**
ResNet50						
DeepCluster	18.9	27.3	36.7	52.4	44.2	–
LA	10.2	23.3	39.3	49.0	60.2	49.4
ISL	**17.3**	24.2	38.5	52.5	61.2	**50.2**
MoCo-v1*	15.7	22.9	40.6	50.8	60.6	37.7
MoCo-v2*	14.9	**28.4**	41.7	**52.9**	67.5	38.5
MoCo-v2+ISL*	13.2	27.1	**41.9**	51.7	**68.6**	40.1

similar instances, LA obtained the state-of-the-art performance among neighborhood discovery methods. However, LA ignored the mismatch between the Euclidean distance of sample pairs and the geodesic distance among instances that revealed the semantics. On the contrary, our ISL mines the feature manifold via GANs, and learns the instance similarity through the generated proxy to supervise the representation learning. Our method achieves the best performance among all existing neighborhood discovery methods when applying the high-level Conv4 and Conv5 features in the linear classifier and the FC features in kNN. MoCo-v1 [16] verified that building large and consistent dictionary on-the-fly via momentum contrast could facilitate effective largescale contrastive learning, and SimCLR [4] validated that an extra MLP projection head and more data augmentation benefited the contrastive learning. In order to further enhance the performance of our ISL, we integrated the proposed method with MoCo-v2 [5] that combined the techniques from both MoCo-v1 and SimCLR. The accuracy of Moco-v2 was obtained by rerunning the officially released code. Since our ISL employ the neighborhood discovery via the geodesic distance on the mined feature manifold, the feature discriminativeness is further strengthened by the informative pseudo supervision in contrastive learning. Figure 5 visualizes an example of positive sample mining via LA and our ISL in different rounds during training. LA treats instances with similar appearance including colors and shapes as positive samples and fails to distinguish the fine-grained difference among various classes. On the contrary, our method mines the feature manifold to assign similarity among instances and successfully finds the semantically similar samples even with different appearance.

5. Conclusion

In this paper, we have presented an instance similarity learning (ISL) method for unsupervised feature representation. The proposed ISL mines the feature manifold by GANs and learns the semantic similarity among instances by exploring the mined feature manifold, through which informative pseudo supervision is provided to learn discriminative features. Extensive experiments demonstrate the superiority of the proposed method compared with the state-of-the-art unsupervised features.

Acknowledgements

This work was supported in part by the National Key Research and Development Program of China under Grant 2017YFA0700802, in part by the National Natural Science Foundation of China under Grant 61822603, Grant U1813218, and Grant U1713214, in part by a grant from the Beijing Academy of Artificial Intelligence (BAAI), and in part by a grant from the Institute for Guo Qiang, Tsinghua University.

References

[1] Philip Bachman, R Devon Hjelm, and William Buchwalter. Learning representations by maximizing mutual information across views. In *NIPS*, pages 15509–15519, 2019.

[2] Piotr Bojanowski and Armand Joulin. Unsupervised learning by predicting noise. In *ICML*, pages 517–526, 2017.

[3] Mathilde Caron, Piotr Bojanowski, Armand Joulin, and Matthijs Douze. Deep clustering for unsupervised learning of visual features. In *ECCV*, pages 132–149, 2018.

[4] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. *arXiv preprint arXiv:2002.05709*, 2020.

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

[6] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *CVPR*, pages 248–255, 2009.

[7] Jiankang Deng, Jia Guo, Niannan Xue, and Stefanos Zafeiriou. Arcface: Additive angular margin loss for deep face recognition. In *CVPR*, pages 4690–4699, 2019.

[8] Carl Doersch, Abhinav Gupta, and Alexei A Efros. Unsupervised visual representation learning by context prediction. In *ICCV*, pages 1422–1430, 2015.

[9] Jeff Donahue, Philipp Krähenbühl, and Trevor Darrell. Adversarial feature learning. *arXiv preprint arXiv:1605.09782*, 2016.

[10] Jeff Donahue and Karen Simonyan. Large scale adversarial representation learning. In *NIPS*, pages 10542–10552, 2019.

[11] Zeyu Feng, Chang Xu, and Dacheng Tao. Self-supervised representation learning by rotation feature decoupling. In *CVPR*, pages 10364–10374, 2019.

[12] Spyros Gidaris, Praveer Singh, and Nikos Komodakis. Unsupervised representation learning by predicting image rotations. *arXiv preprint arXiv:1803.07728*, 2018.

[13] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *NIPS*, pages 2672–2680, 2014.

[14] Albert Gordo, Jon Almazán, Jerome Revaud, and Diane Larlus. Deep image retrieval: Learning global representations for image search. In *ECCV*, pages 241–257, 2016.

[15] Ben Harwood, BG Kumar, Gustavo Carneiro, Ian Reid, Tom Drummond, et al. Smart mining for deep metric learning. In *ICCV*, pages 2821–2829, 2017.

[16] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. *arXiv preprint arXiv:1911.05722*, 2019.

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

[18] Olivier J Hénaff, Aravind Srinivas, Jeffrey De Fauw, Ali Razavi, Carl Doersch, SM Eslami, and Aaron van den Oord. Data-efficient image recognition with contrastive predictive coding. *arXiv preprint arXiv:1905.09272*, 2019.

[19] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015.

[20] Geoffrey E Hinton, Simon Osindero, and Yee-Whye Teh. A fast learning algorithm for deep belief nets. *Neural computation*, 18(7):1527–1554, 2006.

[21] R Devon Hjelm, Alex Fedorov, Samuel Lavoie-Marchildon, Karan Grewal, Phil Bachman, Adam Trischler, and Yoshua Bengio. Learning deep representations by mutual information estimation and maximization. *arXiv preprint arXiv:1808.06670*, 2018.

[22] Jiabo Huang, Qi Dong, Shaogang Gong, and Xiatian Zhu. Unsupervised deep learning by neighbourhood discovery. *arXiv preprint arXiv:1904.11567*, 2019.

[23] Jiabo Huang, Qi Dong, Shaogang Gong, and Xiatian Zhu. Unsupervised deep learning via affinity diffusion. In *AAAI*, pages 11029–11036, 2020.

[24] Xu Ji, Joao F Henriques, and Andrea Vedaldi. Invariant information distillation for unsupervised image segmentation and clustering. *arXiv preprint arXiv:1807.06653*, 2018.

[25] Dahun Kim, Donghyeon Cho, and In So Kweon. Self-supervised video representation learning with space-time cubic puzzles. In *AAAI*, volume 33, pages 8545–8552, 2019.

[26] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[27] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.

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

[29] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *NIPS*, pages 1097–1105, 2012.

[30] Hei Law and Jia Deng. Cornernet: Detecting objects as paired keypoints. In *ECCV*, pages 734–750, 2018.

[31] Honglak Lee, Roger Grosse, Rajesh Ranganath, and Andrew Y Ng. Convolutional deep belief networks for scalable unsupervised learning of hierarchical representations. In *ICML*, pages 609–616, 2009.

[32] Haomiao Liu, Ruiping Wang, Shiguang Shan, and Xilin Chen. Deep supervised hashing for fast image retrieval. In *CVPR*, pages 2064–2072, 2016.

[33] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, and Alexander C Berg. Ssd: Single shot multibox detector. In *ECCV*, pages 21–37, 2016.

[34] Weiyang Liu, Yandong Wen, Zhiding Yu, Ming Li, Bhiksha Raj, and Le Song. Sphereface: Deep hypersphere embedding for face recognition. In *CVPR*, pages 212–220, 2017.

[35] Ishan Misra and Laurens van der Maaten. Self-supervised learning of pretext-invariant representations. In *CVPR*, pages 6707–6717, 2020.

[36] Yuval Netzer, Tao Wang, Adam Coates, Alessandro Bissacco, Bo Wu, and Andrew Y Ng. Reading digits in natural images with unsupervised feature learning. 2011.

[37] Mehdi Noroozi and Paolo Favaro. Unsupervised learning of visual representations by solving jigsaw puzzles. In *ECCV*, pages 69–84, 2016.

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

[39] Deepak Pathak, Philipp Krahenbuhl, Jeff Donahue, Trevor Darrell, and Alexei A Efros. Context encoders: Feature learning by inpainting. In *CVPR*, pages 2536–2544, 2016.

[40] Alec Radford, Luke Metz, and Soumith Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks. *arXiv preprint arXiv:1511.06434*, 2015.

[41] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *NIPS*, pages 91–99, 2015.

[42] Jerome Revaud, Jon Almazán, Rafael S Rezende, and Cesar Roberto de Souza. Learning with average precision: Training image retrieval with a listwise loss. In *ICCV*, pages 5107–5116, 2019.

[43] Pascal Vincent, Hugo Larochelle, Isabelle Lajoie, Yoshua Bengio, and Pierre-Antoine Manzagol. Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion. *JMLR*, 11(Dec):3371–3408, 2010.

[44] Xiaolong Wang and Abhinav Gupta. Unsupervised learning of visual representations using videos. In *ICCV*, pages 2794–2802, 2015.

[45] Yandong Wen, Kaipeng Zhang, Zhifeng Li, and Yu Qiao. A discriminative feature learning approach for deep face recognition. In *ECCV*, pages 499–515, 2016.

[46] Zhirong Wu, Yuanjun Xiong, Stella X Yu, and Dahua Lin. Unsupervised feature learning via non-parametric instance discrimination. In *CVPR*, pages 3733–3742, 2018.

[47] Junyuan Xie, Ross Girshick, and Ali Farhadi. Unsupervised deep embedding for clustering analysis. In *ICML*, pages 478–487, 2016.

[48] Bo Yang, Xiao Fu, Nicholas D Sidiropoulos, and Mingyi Hong. Towards k-means-friendly spaces: Simultaneous deep learning and clustering. In *ICML*, pages 3861–3870, 2017.

[49] Mang Ye, Xu Zhang, Pong C Yuen, and Shih-Fu Chang. Unsupervised embedding learning via invariant and spreading instance feature. In *CVPR*, pages 6210–6219, 2019.

[50] Liheng Zhang, Guo-Jun Qi, Liqiang Wang, and Jiebo Luo. Aet vs. aed: Unsupervised representation learning by auto-encoding transformations rather than data. In *CVPR*, pages 2547–2555, 2019.

[51] Richard Zhang, Phillip Isola, and Alexei A Efros. Split-brain autoencoders: Unsupervised learning by cross-channel prediction. In *CVPR*, pages 1058–1067, 2017.

[52] Yiru Zhao, Zhongming Jin, Guo-jun Qi, Hongtao Lu, and Xian-sheng Hua. An adversarial approach to hard triplet generation. In *ECCV*, pages 501–517, 2018.

[53] Wenzhao Zheng, Zhaodong Chen, Jiwen Lu, and Jie Zhou. Hardness-aware deep metric learning. In *CVPR*, pages 72–81, 2019.

[54] Chengxu Zhuang, Alex Lin Zhai, and Daniel Yamins. Local aggregation for unsupervised learning of visual embeddings. In *ICCV*, pages 6002–6012, 2019.