A Real-Time Memory Updating Strategy for Unsupervised Person Re-Identification

Junhui Yin, Xinyu Zhang, Zhanyu Ma, *Senior Member, IEEE*, Jun Guo, and Yifan Liu, *Member, IEEE*

***Abstract*— Recently, clustering-based methods have been the dominant solution for unsupervised person re-identification (ReID). Memory-based contrastive learning is widely used for its effectiveness in unsupervised representation learning. However, we find that the inaccurate cluster proxies and the momentum updating strategy do harm to the contrastive learning system. In this paper, we propose a real-time memory updating strategy (RTMem) to update the cluster centroid with a randomly sampled instance feature in the current mini-batch without momentum. Compared to the method that calculates the mean feature vectors as the cluster centroid and updating it with momentum, RTMem enables the features to be up-to-date for each cluster. Based on RTMem, we propose two contrastive losses, *i.e.,* sample-to-instance and sample-to-cluster, to align the relationships between samples to each cluster and to all outliers not belonging to any other clusters. On the one hand, sample-to-instance loss explores the sample relationships of the whole dataset to enhance the capability of density-based clustering algorithm, which relies on similarity measurement for the instance-level images. On the other hand, with pseudo-labels generated by the density-based clustering algorithm, sample-to-cluster loss enforces the sample to be close to its cluster proxy while being far from other proxies. With the simple RTMem contrastive learning strategy, the performance of the corresponding baseline is improved by 9.3% on Market-1501 dataset. Our method consistently outperforms state-of-the-art unsupervised learning person ReID methods on three benchmark datasets. Code is made available at:https://github.com/PRIS-CV/RTMem.**

***Index Terms*— Real-time memory updating, unsupervised person ReID, contrastive learning, memory bank.**

I. Introduction

PERSON re-identification (ReID) targets at retrieving the person of interest under different camera views. It is widely used in large-scale security systems in the real world. Although great progress has been made by supervised ReID methods [1], [2], [3], [4], [5], reliance on cross-camera identity labels limits the scalability of ReID systems to real-world applications. Recently, unsupervised person ReID methods [6], [7], [8], [9] have drawn much more attention for their potential on the unlimited scalability.

Most existing unsupervised person ReID methods can be divided into two categories. One kind of methods is the unsupervised domain adaptation (UDA). These methods [9], [10], [11] first learn abundant knowledge from the labeled source-domain dataset, and then transfer the learned knowledge to the unlabeled target-domain dataset. UDA methods, however, heavily depend on the scale and quality of the source-domain dataset. Another category is the fully unsupervised learning (USL). USL methods [6], [7], [8], [12] usually generate pseudo labels from the unlabeled dataset by a clustering algorithm and train ReID model with these pseudo labels. USL is more challenging but owns more flexibility as it does not require any identity annotation.

During the model training, most pioneering USL works [9], [13], [14], [15], [16] employs the mean feature of all instances belonging to its pseudo identity as the cluster centroid. They assume that the data of each cluster is distributed in a high-dimensional spherical distribution, which follows the same assumption in the conventional K-means. In contrast, lots of methods [16], [17], [18] find that the density-based clustering algorithm, such as DBSCAN, can achieve higher accuracy in USL ReID task. It is because that the data points actually distribute in a manifold which may not always follow spherical clusters. Despite the employment of a proper clustering algorithm, there still exist conflicts between the clustering manner and the training process in current SOTA methods [15], [16], [17], [19]. First, it is inaccurate to utilize the mean feature as the proxy point of the cluster centroid. As shown in Figure 1 (d), if the real data distribute in a manifold instead of the spherical cluster, the proxy point of the mean feature may not fall in its own cluster and even in other clusters. Second, most existing methods update the proxy point with a momentum updating scheme. This scheme follows the same assumption of K-means on the spherical distribution in a cluster, which is not consistent with the assumption of DBSCAN. It thus results in the sub-optimal performance.

To overcome existing conflicts, we propose a novel real-time memory updating strategy for unsupervised person ReID. The main idea is to directly *replace* the feature stored in the memory bank with a random feature sampled from the current mini-batch without the momentum update. We call this strategy as the real-time memory updating strategy (RTMem). An instance-level and a cluster-level memory bank

Manuscript received 26 August 2022; revised 6 March 2023; accepted 14 March 2023. Date of publication 14 April 2023; date of current version 21 April 2023. This work was supported in part by Beijing Natural Science Foundation Project No. Z200002, in part by National Natural Science Foundation of China (NSFC) No. U19B2036 and 62225601, in part by the Program for Youth Innovative Research Team of BUPT No. 2023QNTD02, and in part by the BUPT Excellent Ph.D. Students Foundation under Grant CX2021205. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Yi Yang. *(Corresponding author: Zhanyu Ma.)*

Junhui Yin, Zhanyu Ma, and Jun Guo are with the Pattern Recognition and Intelligent System Laboratory, School of Artificial Intelligence, Beijing University of Posts and Telecommunications, Beijing 100876, China (e-mail: mazhanyu@bupt.edu.cn).

Xinyu Zhang is with Baidu VIS, Beijing 100193, China.

Yifan Liu is with the School of Computer Science, The University of Adelaide, Adelaide, SA 5005, Australia.

Digital Object Identifier 10.1109/TIP.2023.3266166

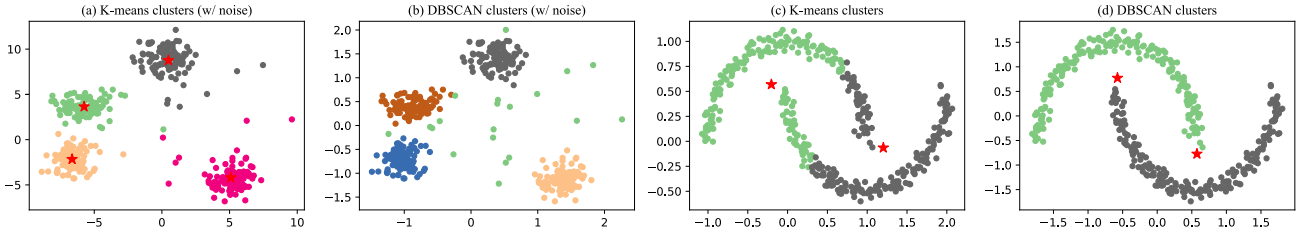


Fig. 1. The K-means algorithm classifies data points into spherical clusters based on the Euclidean distance. However, it does not work well on noisy and non-convex shaped data points (*e.g.*, toroidal-shaped data). On the contrary, the density-based DBSCAN algorithm is more satisfactory on different shaped points, with less influence by noises and outliers (*i.e.*, green dots in (b)). Besides, when the clustering algorithm produces non-spherical clusters as shown in (c) and (d), the mean feature can not well represent a cluster centroid. Note that, rounded dots represent data points, while pentagrams mean the mean feature for the cluster.

are built to calculate two contrastive losses for the model optimization, *i.e.*, sample-to-instance and sample-to-cluster losses. Considering the DBSCAN clustering manner that measures the instance-level similarities, our sample-to-instance contrastive loss treats anyone sample as an anchor and explores all its positive and negative samples stored in the instance-level memory bank during the training process. This allows our model to take full advantage of the global information, which is beneficial for overcoming the intra-cluster variations and learning to adapt clustering results. Further, our sample-to-cluster contrastive loss *randomly* picks one image feature in each mini-batch pseudo cluster as the proxy in the cluster-level memory and enforces the mini-batch sample to be close to its cluster proxy while being far from other proxies. It follows the character of the DBSCAN clustering algorithm, maintaining the original data manifold rather than the hypothetical spherical clusters in the previous works. Clustering and instance-level contrasts are not redundant, but synergistically enhance each other. Our key idea follows an intuitive format that each image should be close to any or even all of the samples in the same pseudo cluster to which it belongs. On the one hand, sample-to-instance explores the sample relationships of the whole dataset to enhance the clustering ability of DBSCAN, which relies on similarity measurement for the instance-level images. On the other hand, with pseudo-labels generated by DBSCAN, sample-to-cluster enforces the sample to be close to its cluster proxy while being far from other proxies.

The overall framework diagram of our proposed method is illustrated in Figure 2. Our contributions can be summarized as follows:

- We point out the limitations of the current momentum-based memory updating strategy and analyze the conflicts between the data distribution assumption in the clustering methods and the model learning process.
- We present a real-time memory updating strategy (RTMem). Based on RTMem, the sample-to-instance and the sample-to-cluster contrastive losses are proposed to improve the representation ability of the features.
- The proposed RTMem contrastive learning framework achieves consistent improvements on all benchmark datasets compared to state-of-the-art unsupervised person ReID methods.

II. Related Work

A. Unsupervised Person ReID

In recent years, unsupervised methods for person ReID have been proposed and they can be divided into two main categories, including generative network based methods [11], [20], [21], [22], [23], [24] and clustering-based methods [7], [8], [12], [13], [25], [26], [27], [28], [29], [30], [31]. Generative networks based methods leveraged GANs to learn domain-invariant information from cross domain style-transferred images [20], [21] or disentangle feature space into id-related/unrelated components [11], [23]. In clustering-based methods, their labels are obtained from feature similarity computation [28], [29], [30] or clustering features [7], [8], [12], [13], [25], [26], [27]. Clustering-based methods gradually becomes a mainstream learning paradigm to achieve state-of-the-art performance. Early works [7], [9], [14] mostly heavily relied on the K-means algorithm to generate pseudo labels. Due to its poor clustering capability, ReID models often need to be pre-trained on labeled source domain before clustering image features on the unlabeled target domain. The major challenge behind it is how to alleviate the effect brought by pseudo-label noise. Several works [13], [26], [32], [33] leveraged DBSCAN algorithm [34] to discard noisy labels automatically and generate high-quality clustering results based on data distribution. These methods also utilized self-similarity grouping [26], progressive augmentation strategies [13], multi-feature fusion with adaptive graph learning [35], and style-translated images [32] to further enhance the discriminative ability of the model. As a milestone, SpCL [16] treated each cluster and outlier as a single class while performing contrastive learning based on a hybrid memory containing cross-dataset features. Recently, a series of works retained the contrastive learning but further considered identity centroids for each camera [36], [37], a cluster consistency [15] or reliable pseudo labels generation [17], [38].

B. Joint Clustering and Feature Learning

Clustering algorithm is an effective technique to group unlabeled data into different clusters by clustering image features. K-means [39] and DBSCAN [34] are two traditional clustering methods. Recently, various unsupervised learning methods [40], [41], [42] have shown great potential in jointly optimizing feature learning and image clustering. Closer to our work, DeepCluster [42] adopted deep network model as a prior knowledge to iterative K-means clustering and representation learning. Along this direction, PCL [43] found the distribution of prototype via K-means to reformulate network training as Expectation-Maximization framework. In ReID context, considering that K-means is sensitive to initial cluster centroids and outliers, some methods [7], [9], [14]

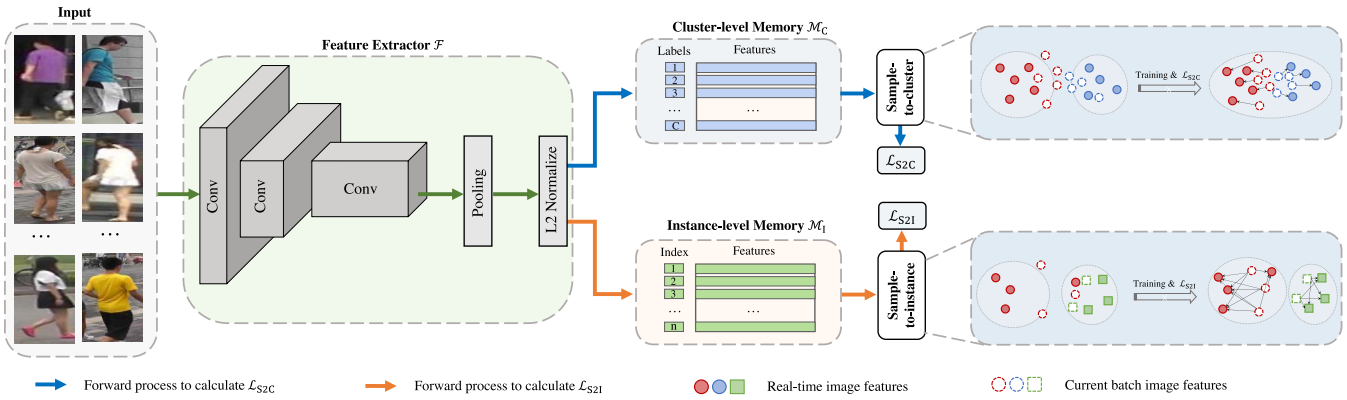


Fig. 2. Overview of the overall training process. The pipeline consists of three main components: feature extractor $\mathcal{F}$, instance-level memory bank $\mathcal{M}_I$, and cluster-level memory bank $\mathcal{M}_C$. During training, we feed the mini-batch training images into $\mathcal{F}$ to obtain their up-to-date feature representations. Then, these newly extracted features are used to conduct the sample-to-instance and the sample-to-cluster contrastive learning based on $\mathcal{M}_I$ and $\mathcal{M}_C$. During the back-propagation, we directly replace memory features to the coming-in features at the corresponding position. The details are illustrated in Section III.

designed special mechanisms to avoid those situations, *e.g.* initializing ReID models with labeled source dataset and refining pseudo labels in a mutual learning strategy. Recently, a series of works [15], [16], [33], [36] used the DBSCAN algorithm to produce pseudo-labels and integrated it with contrastive learning to refine features. Reference [44] generate support samples from actual samples and their neighboring clusters to discover underlying information and reveal the accurate clusters.

Their success can largely be ascribed to the use of DBSCAN, which can handle clusters of different sizes or shapes and is less affected by noise and outliers. Different from these approaches, our work comprehensively considers feature update strategy and enforces these features over the entire dataset to fit DBSCAN algorithm, rather than simply using pseudo labels for hardest mining. For example, ClusterContrast [15] used cluster centroid as cluster proxy and update it with one (hardest) instance feature vector from the current mini-batch in a moving averaging manner. Inter-instance Contrastive Encoding (ICE) [19] leverages inter-instance pairwise similarity scores as one-hot hard pseudo labels for hard instance contrast, which aims at reducing intra-class variance. However, ICE, an unsupervised learning paradigm, rarely ensure the hardest mining ability with noisy labels and incorrect case of hardest mining can mislead the ReID model. As shown in Table III, randomly picking the proxy surpasses the hardest proxy strategy with large margins (e.g. 2.3% for mAP on Market). Different from the above works, our method only requires to randomly select one image feature from each cluster (without momentum updating) and compare it with current mini-batch image features in a contrastive learning manner. Our key intuition is that the training data of the same class can follow an unbalanced distribution, *i.e.* the shape of data distribution is not the hypothetical spherical distribution in the previous works [16], [19] and there are no cluster centers and hardest (easiest) pairs in feature space.

The non-parametric memory bank has been presented to address various tasks, including unsupervised contrastive learning [45], [46], metric learning [47], few-shot learning [48], [49], face recognition [50], [51] and unsupervised ReID [16], [27]. In these computer vision tasks, non-parametric memory allows sample features to be stored directly in the feature bank and updated at each training iteration. References [27], [28], [45] stored the whole dataset and treat each image instance as a different class, which is distinguished with a classifier. Recently, several works [47], [51] identify the phenomenon of slow feature drift and directly use the current mini-bath feature to update the embeddings of instances, without additional computational cost (*i.e.* moving averaging). Inspired by these approaches, we have developed a new memory-based framework for unsupervised ReID. However, our work differs from these memory-based variants in: (1) All the above methods take a memory bank as instance-level storage to memorize the past features, while we bridge feature learning and clustering features not only at instance-level but also at cluster-level. (2) [27], [28], [45] discard clustering and treat each instance as a single class, while we make full use of DBSCAN clustering and inject memorized features into both sample-to-cluster and sample-to-instance contrast. (3) Although XBM [47] and VPL [51] and our work all use the slow feature drift, these methods train network models with human-annotated labels in a supervised manner. In this paper, the training of our method can be viewed as an unsupervised process and it can achieve better performance over the previous unsupervised methods, even surpassing supervised methods on several benchmarks. (4) [45] uses real-time memory updating to store features and conduct instance-level non-parametric classification, but our method addresses the inconsistency between clustering algorithm and momentum-based contrastive learning from sample-to-cluster and sample-to-instance contrast. In addition, our method does not require a feature smoothing term [45] to stabilize the training process and instead relies on real-time sample-to-cluster contrast to ensure the discriminative ability of the approach. This strategy allows the features to be more up-to-date and provides more accurate cluster proxies.

III. Methodology

A. Preliminary and Revisiting

Let $\mathcal{X} = \{x_1, x_2, \ldots, x_n\}$ denote an unlabeled training set, where x_i is the i-th unlabeled image and n is the total number of person images. The aim of the unsupervised person ReID is

to train a robust model $\mathcal{F} = f(\theta; x)$ to project an image to a specific embedding feature $\boldsymbol{f} \in \mathbb{R}^d$. d is the feature dimension. Most of methods [15], [16], [17], [19], [33] perform the two-step iterative strategy: i) generate pseudo labels $\mathcal{Y} = \{y_i, y_2, \ldots, y_n\}$ of all training images via offline clustering operations, such as Kmeans [39] and DBSCAN [34]. Here, $y_i \in \{1, 2, \ldots, C\}$ and C is the number of pseudo labels. ii) optimize the model with the obtained pseudo-labeled dataset $\mathcal{X}' = \{(x_1, y_1), (x_1, y_2), \ldots, (x_n, y_n)\}$.

1) Revisiting Memory-Based Methods: During the model optimization, current state-of-state-art methods [15], [16], [17], [18], [33], [36] adopt the non-parametric InfoNCE [52] as the loss function. Despite the various variations of InfoNCE in different approaches, the unified formulation is defined by:

$$\mathcal{L} = -log \frac{exp(\boldsymbol{f}_i^T \cdot \boldsymbol{m}_i/\tau)}{\sum_{j=1}^{K} exp(\boldsymbol{f}_j^T \cdot \boldsymbol{m}_j/\tau)}. \quad (1)$$

where τ is a temperature that controls the concentration of distribution [53]. $\boldsymbol{f}_i$ is L2-normalized, *i.e.*, $\boldsymbol{f}_i \leftarrow \boldsymbol{f}_i/||\boldsymbol{f}_i||_2$. $\boldsymbol{m}_i$ is a specific proxy entry that the i-th sample belongs to. Specifically, $\boldsymbol{m}$ is picked from a memory bank $\mathcal{M}$. There are several types of $\mathcal{M}$: i) If $\mathcal{M}$ stores the feature of each image sample [16], $\mathcal{M} \in \mathbb{R}^{d \times n}$ and $\boldsymbol{m}_i$ is the i-th entry of $\mathcal{M}$. ii) If $\mathcal{M}$ saves the centroid of each cluster [15], [19], $\mathcal{M} \in \mathbb{R}^{d \times C}$ and $\boldsymbol{m}_i$ is the y_i-th entry of $\mathcal{M}$. iii) If camera IDs are available, $\mathcal{M}$ can maintain the camera-based centroid of each cluster [19], [36]. For each cluster c, $\mathcal{M}[c]$ is separated into $\mathcal{A} = \{a_1, a_2, \cdots a_c\}$ entries, where a_j represents the j-th camera of the cluster c and the total camera number is a_c. In this way, $\boldsymbol{m}_i$ is the j-th camera entry of $\mathcal{M}[y_i]$.

It is important to update the memory bank $\mathcal{M}$ during the back-propagation. Most of previous approaches apply the momentum updating strategy following [27], which is denoted as:

$$\boldsymbol{m}_i \longleftarrow \alpha \boldsymbol{m}_i + (1 - \alpha)\boldsymbol{f}_i, \quad (2)$$

where the hyper-parameter $\alpha \in [0, 1]$ controls the updating rate. The larger α will maintain more previous information, while the smaller α will focus more on current features. Therefore, the former works usually adjust α carefully to achieve the best result. In this paper, we denote the centroid-based memory (the 2rd type of $\mathcal{M}$) with the above momentum updating strategy as our baseline $\mathcal{L}_{\text{Base}}$.

B. Real-Time Memory Updating Strategy

To the best of our knowledge, there is no prior work to explore the negative impact of the momentum updating strategy. In other words, we still do not know *whether the momentum updating is optimal or not*, especially under the DBSCAN clustering method. In this paper, we first analyze this issue from a new perspective, and then propose a novel *real-time* memory updating strategy.

1) A New Perspective: Inconsistency Between the Clustering Algorithm and the Momentum Memory Updating Strategy: Previous centroid-based methods [15], [16], [19], [36] generally represent a cluster with its centroid, *i.e.*, the mean feature of all samples belonging to this cluster. The centroid-based memory is then stored in $\mathcal{M}$ as shown in Section III-A.1. The canonical way to update $\mathcal{M}$ is the *momentum updating*

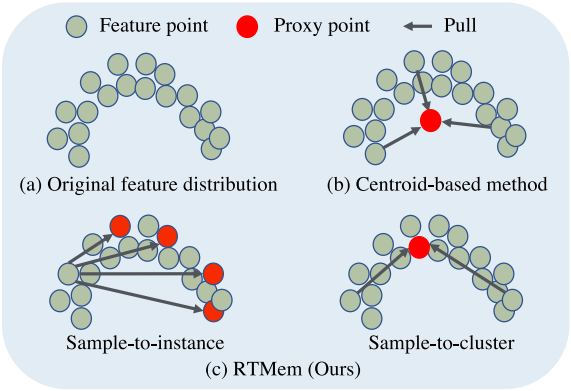


Fig. 3. Examples of the learning process. All points belong to the same pseudo cluster. (a) The original feature distribution. (b) The centroid-based method: the mean feature is used as the proxy point of the cluster centroid. All features are pulled to the proxy point. (c) Our RTMem: The sample-to-instance memory stores instance features as proxy points. RTMem encourage a specific feature to be close to all proxies with the same pseudo label (sample-to-instance memory) and the randomly sampled proxy in the same cluster (sample-to-cluster memory). For simplicity, we only present a few proxy points to illustrate our method.

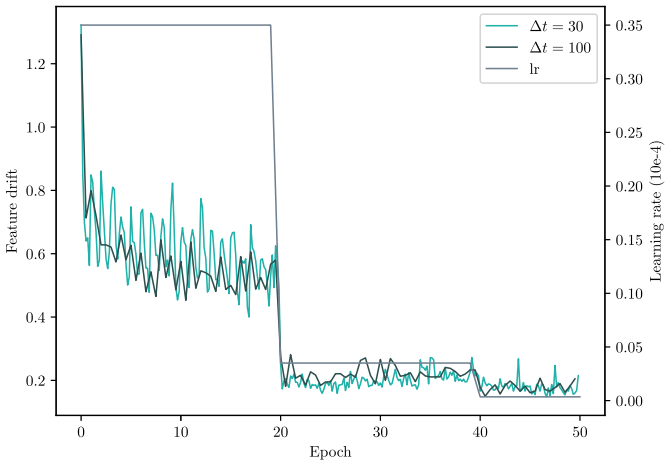


Fig. 4. The slow drift phenomena observed in the feature updating process. After one epoch, the Euclidean distance between embedding features at the present and the previous iterations is relatively small. Even though there is a large variation of the feature representations due to the decrease of the learning rate, the re-initialization of the memorized features at each epoch can maintain the slow drift of the feature change.

as Eq. (2). It is based on the assumption that the data follows a perfect spherical distribution. Based on the assumption, it is natural to utilize K-means [39] as the clustering method since K-means also estimates a proxy as the cluster center. However, most of the prior approaches apply the density-based DBSCAN [34] algorithm as the clustering method and find that DBSCAN outperforms K-means by a large margin. The underlying reason is that the data points distribute in a manifold that the clustering manner of DBSCAN is more appropriate. Despite the proper clustering algorithm to generate good pseudo labels, these approaches still use the momentum updating strategy on $\mathcal{M}$. Actually, the manifold requires Euclidean space in a local area. That is to say that the mean feature of a cluster may be not in the real distribution (as shown in Figure 1). It thus results in sub-optimal performance due to the inconsistency between the clustering algorithm and the momentum-based memory updating strategy.

2) Real-Time Memory Updating Strategy (RTMem): To address the above problem, we propose to update the memory bank with the timely feature embedding. This behavior is named as the *real-time* memory updating strategy. In detail, we simply replace the memory entry $\boldsymbol{m}_i$ with the current feature $\boldsymbol{f}_i$:

$$\boldsymbol{m}_i \leftarrow \boldsymbol{f}_i \tag{3}$$

Interestingly, our real-time updating strategy in Eq. (3) is a special situation of Eq. (2) when $\alpha = 0$. In this case, we can preserve the original feature distribution with the sample feature rather than the fake distribution with the mean feature. Meanwhile, it is reasonable to directly replace the memory entry with the feature embedding. As shown in Figure 4, the feature drift (formulated with the Euclidean distance between the features of the preceding iteration and those of the present iteration) gradually becomes low. It demonstrates the stability and effectiveness of the real-time updating strategy. Besides, there is no need to carefully tune the hyper-parameter α, making our method more efficient. The proposed updating strategy enables cluster learnable to provide accurate cluster proxies for sample-to-cluster contrast. Some existing works [54], [55] have a similar idea of the learnable queries or prototype. For example, [54] uses a set of learnable queries to interact with both video and textual representations. [55] uses the learnable verb prototypes to guide noun classification with information flow between verb and noun.

C. Optimization

Based on the proposed RTMem, we further propose two losses for the model optimization, *i.e.*, the sample-to-instance loss and the sample-to-cluster loss.

1) Sample-to-Instance Loss With RTMem: Considering the DBSCAN clustering manner that measures the instance-level similarities, it is necessary to explore the sample relationships during the training process. To this end, we first construct a memory bank $\mathcal{M}_\mathrm{I}$, storing the features of all images in the training set. Here, $\mathcal{M}_\mathrm{I}[i]$ is the i-th entry of $\mathcal{M}_\mathrm{I}$, representing the i-th image. To ensure the real-time memory updating on the instance feature, we directly *replace* $\mathcal{M}_\mathrm{I}[i]$ with the timely feature $\boldsymbol{f}_i$ in every iteration, where $\boldsymbol{f}_i$ is in the current mini-batch $\mathcal{B}$:

$$\mathcal{M}_\mathrm{I}[i] \leftarrow \boldsymbol{f}_i. \tag{4}$$

The goal of the model optimization is to pull intra-cluster samples together while push inter-cluster samples far away from each other. To this end, given a specific feature $\boldsymbol{f}_i$, we propose a *sample-to-instance* loss function, which is formulated as:

$$\mathcal{L}_\mathrm{S2I} = -log\frac{\sum_{s\in\mathcal{S}}\exp(\boldsymbol{f}_i^T\cdot\boldsymbol{m}_s/\tau)}{\sum_{j=1}^{n}\exp(\boldsymbol{f}_i^T\cdot\boldsymbol{m}_j/\tau)}, \tag{5}$$

where $\boldsymbol{m}_s = \mathcal{M}_\mathrm{I}[s]$, $s \in \mathcal{S}$. $\mathcal{S}$ is a set that consists of all positive samples, sharing the same pseudo label with $\boldsymbol{f}_i$. Note that the features in $\mathcal{M}_\mathrm{I}$ are real-time updated via Eq. (4). In this way, the sample-to-instance loss $\mathcal{L}_\mathrm{S2I}$ can focus on the up-to-date features, instead of the fake and obsolete momentum-updated features [16], [56]. Meanwhile, we take all samples into consideration at once. It is different from [56] that [56] only considers the hardest positive samples. We argue that our $\mathcal{L}_\mathrm{S2I}$ can take full advantages of the global information that is beneficial for overcoming the intra-cluster variations.

Algorithm 1 Our RTMem Algorithm on Unsupervised Person ReID

Require: Unlabeled data $\mathcal{X}$, network $\mathcal{F}$, the instance-based memory bank $\mathcal{M}_\mathrm{I}$ initialized with features of all training images, the cluster-based memory bank $\mathcal{M}_\mathrm{C}$ initialized with image features, the temperature τ, the iteration number N_iters, the total training epochs N_epochs.
Ensure: A powerful $\mathcal{F}$.

```
for epoch in [1, N_epochs] do
    Cluster X into C pseudo clusters;
    Obtain the pseudo-labeled dataset X' = {(x_1, y_1), (x_1, y_2), ..., (x_n, y_n)};
    Initial the centroid-based memory bank M_C in Eq.(6);
    for iter in [1, N_iters] do
        Sample mini-batch images and extract the sample features f;
        Calculate L_S2I and L_S2C according to Eq. (5) and Eq. (7);
        Update the network F by back-propagation;
        Update M_I and M_C according to Eq. (4) and Eq. (6);
    end for
end for
```

2) Sample-to-Cluster Loss With RTMem: The previous work [15] finds that the centroid-based memory with the contrastive loss is effective on the unsupervised ReID. However, they still utilize the momentum updating on the memory bank to keep the stability of the training process. As analyses in Section III-B, there exists the inconsistency on the momentum updating scheme and the DBSCAN clustering manner. Thus, we update the centroid-based memory by the proposed real-time updating strategy. Specifically, we build a centroid-based memory bank $\mathcal{M}_\mathrm{C}$ as in [15]. Instead of the momentum update, for a specific mini-batch $\mathcal{B}$, we *randomly* pick one sample in each mini-batch pseudo cluster. For a specific $\boldsymbol{f}_i$ with the pseudo label y_i, we directly *replace* the y_i-th entry in $\mathcal{M}_\mathrm{C}$ to $\boldsymbol{f}_i$ in each iteration. The formulation is as follows:

$$\mathcal{M}_\mathrm{C}[y_i] \leftarrow \boldsymbol{f}_i. \tag{6}$$

In this way, the randomly sampled instance can be considered as the newly assigned proxy of the cluster centroid as illustrated in Figure 3. Here, we also initial $\mathcal{M}_\mathrm{C}$ in each epoch with the randomly picked sample in each cluster. Then we utilize the non-parameter InfoNCE [52] loss function on $\mathcal{M}_\mathrm{C}$ to enforce the sample to be close to its cluster proxy while being far from other proxies. We name this loss as the *sample-to-cluster* loss function, denoted as:

$$\mathcal{L}_\mathrm{S2C} = -\log\frac{\exp(\boldsymbol{f}_i^T\cdot\mathcal{M}_\mathrm{C}[y_i]/\tau)}{\sum_{j=1}^{C}\exp(\boldsymbol{f}_i^T\cdot\mathcal{M}_\mathrm{C}[j]/\tau)}. \tag{7}$$

Different from [15], our $\mathcal{L}_\mathrm{S2C}$ directly utilize the up-to-date centroid proxy for the model optimization. It follows the characteristic of the DBSCAN clustering algorithm, maintaining the original data manifold rather than the hypothetical spherical distribution in the previous works [16], [19].

3) Overall: In each iteration, we joint optimize the sample-to-instance loss and the sample-to-cluster loss. The overall optimization is:

$$\mathcal{L} = \mathcal{L}_\mathrm{S2C} + \lambda\mathcal{L}_\mathrm{S2I}, \tag{8}$$

where λ controls the degree of two loss functions. In summary, the proposed RTMem based optimization does not rely on any strict assumption of the data distribution, but follows an intuitive format that each image should be close to any or even all of the samples in the same pseudo cluster to which it belongs. The training details of our RTMem are provided in Algorithm 1.

IV. Experiment

Datasets: We evaluate our proposed RTMem on three large-scale ReID benchmarks, *i.e.*, Market-1501 [57], MSMT17 [20], and VeRi [58].

Market-1501 [57] contains 32,668 annotated images of 1,501 identities captured by 6 disjoint cameras. Among them, 12,936 images of 751 identities are used for training and the remaining images of 705 identities are for testing.

MSMT17 [20] is the most challenging ReID dataset, which is composed of 126,441 bounding boxes of 4,101 identities. The training set has 32,621 images with 1,041 identities and the testing set has 93,820 testing images with 3,060 identities, collected by fifteen cameras.

VeRi [58] consists of over 40,000 bounding boxes of 619 vehicles in real-world traffic scene, where each vehicle is captured by 2∼18 cameras and thus contains different viewpoints, illuminations, and resolutions.

Evaluation Protocols: In all experiments, we adopt mean average precision(mAP) and cumulative matching characteristic (CMC) to evaluate the performance of our methods on three benchmark datasets. No post-processing technique (*e.g.* re-ranking [59] and multi-query fusion [57]) is adopted during testing stage.

Implementation Details: For fairness, we follow the standard experiment settings as in [15], [16], [17], and [19]. Specifically, we adopt an ImageNet [60] pretrained ResNet-50 [61] as the default backbone network. We remove all sub-modules after the 4-th layer and add a global average pooling (GAP) layer or a generalized mean pooling (GEM) layer, followed by a batch normalization layer [62] to yield 2048-dimensional feature embeddings. Without the specification, we adopt GEM as the default pooling operation for all ablation studies. Note that we report both results of GAP and GEM in Table IV in the main paper. The maximum distance between two samples to be considered as the neighbors in DBSCAN [34] is set to 0.5 for Market-1501 and DukeMTMC-reID, and 0.7 for MSMT17. At the beginning of each epoch, DBSCAN is first utilized to generate pseudo labels for training images. During training, person images is resized to 256×128. The image augmentation is random flipping, padding with 10 pixels, random crop, and random erasing [63]. Each mini-batch is composed of 256 training images belonging to 16 pseudo classes and is sampled with the class-balanced manner [64]. The optimizer is Adam [65] and the weight decay is 5×10^{-4}. The learning rate is initialized to 3.5×10^{-4} and is decreased by 0.1 of its previous value every 20 epochs. The total epoch is 50. For sample-to-instance and sample-to-cluster contrastive loss, we set the temperature parameter τ as 0.05 and the balancing factor λ in Eq. (8) as 1.2 based on empirical experiments.

A. Ablation Study

To comprehensively understand that our RTMem can fully exploit the clustering results and fit feature distribution in a manifold for unsupervised ReID task, we conduct a qualitative ablation study to investigate different components of our methods.

1) Effectiveness of the Sample-to-Instance Loss $\mathcal{L}_{S2I}$: We train our baseline model with the mean feature vectors as the cluster centroid and update it with the momentum scheme. As shown in Table I, we observe the obvious improvement when adding $\mathcal{L}_{\text{S2I}}$ into baseline model, especially from 17.2% to 23.5% mAP on MSMT17. RTMem also achieves significant improvements when adding $\mathcal{L}_{\text{S2I}}$ into $\mathcal{L}_{\text{S2C}}$, *e.g.*, 6.6% mAP improvement on MSMT17. It demonstrates that the proxy centroid for the cluster representation in the baseline results in a sub-optimal status. On the contrary, our sample-to-instance loss with RTMem treats each cluster as a set of real-time features. Injecting the real-time features within cluster into the model could take full advantage of DBSCAN clustering to align all intra-cluster samples and overcome their variations, which in turn improves clustering.

2) Effectiveness of the Sample-to-Cluster Loss $\mathcal{L}_{S2C}$: The sample-to-cluster loss $\mathcal{L}_{\text{S2C}}$ is enforced on mini-batch features and proxy features of the real-time memory $\mathcal{M}_{\text{C}}$. It preserves the original data manifold, rather than the hypothetical spherical distribution in the previous works [16], [19]. With $\mathcal{L}_{\text{S2C}}$, our method can adaptively characterize the original data distributions and maintain as much benefits of DBSCAN clustering as possible to achieve the strong representation capability. Applying our RTMem on $\mathcal{M}_{\text{C}}$ can boost the baseline model by 9.3% and 14.7% mAP on Market1501 and MSMT17, respectively. It is observed that $\mathcal{L}_{\text{S2C}}$ combining $\mathcal{L}_{\text{S2I}}$ further improves its retrieval accuracy. Our RTMem generalizes better than the baseline model. We also re-implement XBM [47] on ReID datasets (i.e., Market-1501 and MSMT17) and obtain poor performance. The main reason is that cross-batch memory in XBM is used to collect sufficient sample pairs and mining hard negative examples heavily rely on human-annotated labels in a supervised manner, while our clustering-based method rarely ensures the quality of labels.

3) Clustering Methods v.s. *the Memory Updating Strategy:* Current mainstream methods (*e.g.*, SpCL [16]) typically moving average as feature update strategy and make each image feature converge to its corresponding cluster centroid at uniform space. However, the data points of same class actually distribute in a manifold, where the shape of data distribution is non-uniform and there are no cluster centroids in feature space. This is also why K-means is not a mainstream clustering algorithm due to its poor clustering ability for ReID data. In fact, as shown in Figure 1, K-means based clustering estimates a proxy as the cluster center during clustering process and thus K-means is suitable for the learning paradigm of converging to a center point and not consistent with the memory updating strategy.

To validate this point, we train our RTMem (*i.e.*, $\mathcal{L}_{\text{S2I}}$&$\mathcal{L}_{\text{S2C}}$ in Table II) and the centroid-based learning paradigm (*i.e.*, $\mathcal{L}_{\text{Base}}$ in Table II) with different clustering processes, including K-means and DBSCAN. Specifically, in order to preserve original feature distribution, we adopt the real-time memory updating strategy for sample-to-instance ($\mathcal{L}_{\text{S2I}}$) and

TABLE I

QUALITATIVE ABLATION STUDIES OF THE PROPOSED APPROACH ON MARKET1501 AND MSMT17. THE BASELINE SCHEME ($\mathcal{L}_{\text{BASE}}$) IS TRAINED WITH THE MEAN FEATURE VECTORS AS THE CLUSTER CENTROID AND UPDATING IT WITH MOMENTUM

$\mathcal{L}_{\text{Base}}$	RTMem		Market1501				MSMT17			
	$\mathcal{L}_{\text{S2I}}$	$\mathcal{L}_{\text{S2C}}$	mAP	R1	R5	R10	mAP	R1	R5	R10
✓	×	×	74.6	88.5	95.6	97.0	17.2	42.5	54.1	59.6
✓	✓	×	78.3	90.1	96.0	97.6	23.5	51.1	62.3	67.1
×	×	✓	83.9	92.9	97.1	98.2	31.9	59.8	71.1	75.5
×	✓	✓	86.5	94.3	97.9	98.5	38.5	63.3	75.4	79.6

TABLE II

COMPARISON WITH CENTROID-BASED APPROACHES UNDER DIFFERENT CLUSTERING METHODS. MAP AND R1 ARE REPORTED TO EVALUATE THE PERFORMANCE QUANTITATIVELY ON MARKET1501 (MARKET) AND MSMT17 (MSMT). $\mathcal{L}_{\text{BASE}}$ REPRESENTS THE MOMENTUM-BASED MEMORY UPDATING WITH THE CENTROID-BASED LEARNING. $\mathcal{L}_{\text{S2I}}\&\mathcal{L}_{\text{S2C}}$ DENOTES OUR RTMEM WITH THE SAMPLE-TO-CLUSTER AND SAMPLE-TO-INSTANCE CONTRASTIVE LEARNING. UNDER THE K-MEANS CLUSTERING, THE LEARNING PARADIGM OF CONVERGING TO A CENTER POINT IS SUPERIOR TO OURS, WHILE UNDER DBSCAN CLUSTERING, OUR METHOD PERFORMS SUPERIOR PERFORMANCE

K-means	DBSCAN	$\mathcal{L}_{\text{Base}}$	$\mathcal{L}_{\text{S2I}}\&\mathcal{L}_{\text{S2C}}$	Market				MSMT			
				mAP	R1	R5	R10	mAP	R1	R5	R10
✓	×	✓	×	34.5	61.3	78.3	84.6	1.3	4.2	7.7	9.9
✓	×	×	✓	16.3	37.0	54.9	63.0	1.0	3.1	7.5	10.3
×	✓	✓	×	74.6	88.5	95.6	97.0	12.0	30.6	40.9	45.6
×	✓	×	✓	86.5	94.3	97.9	98.5	38.5	63.3	75.4	79.6

TABLE III

QUALITATIVE ABLATION STUDIES OF THE PROPOSED APPROACH WITH DIFFERENT SAMPLE-TO-CLUSTER FEATURE CONTRASTIVE LOSS ON MARKET-1501

$\mathcal{L}_{\text{S2C}}$			$\mathcal{L}_{\text{S2I}}$	mAP	R1	R5	R10
Hardest	Easiest	Random					
✓	×	×	×	81.6	92.5	96.9	98.2
×	✓	×	×	76.2	89.8	95.3	96.8
×	×	✓	×	83.9	92.9	97.1	98.2
✓	×	×	✓	85.1	93.5	97.4	98.3
×	✓	×	✓	78.4	90.2	96	97.4
×	×	✓	✓	86.5	94.3	97.9	98.5

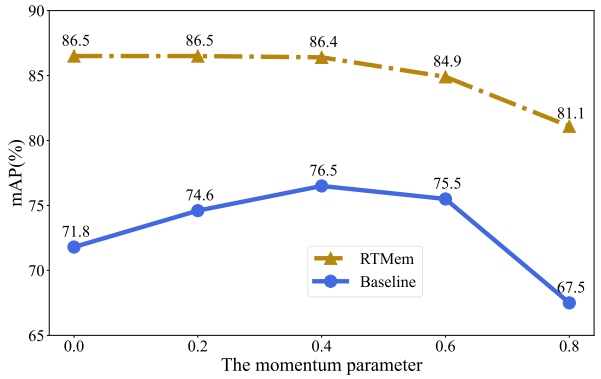


Fig. 5. Performance of our method and baseline with different momentum parameter on Market-1501.

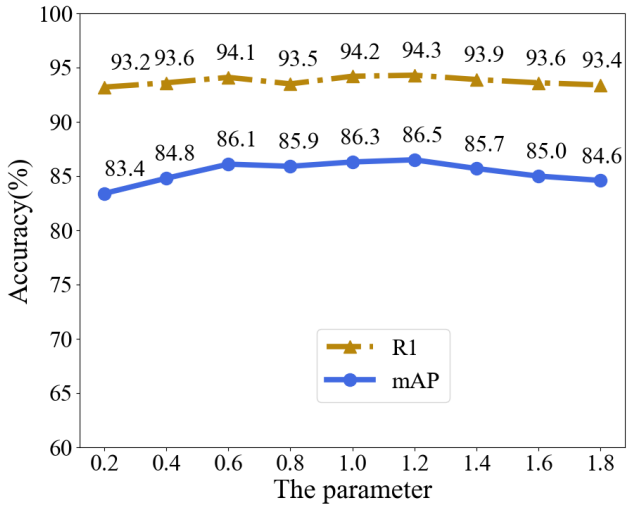
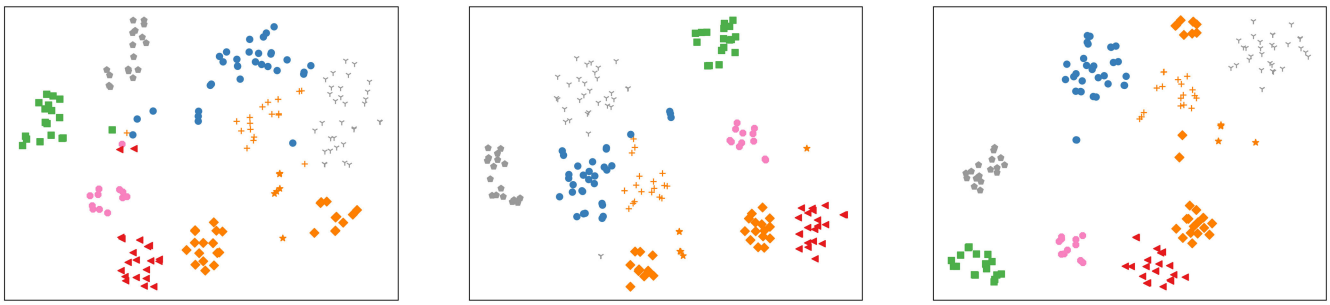


Fig. 6. Results of our method with different values of parameter λ on Market-1501.

sample-to-cluster ($\mathcal{L}_{\text{S2C}}$), in which each memorized feature is directly replaced with up-to-date feature in a random manner.

The detailed results are summarized in Table II, from which we draw two observations. 1) First, comparing our proposed method ($\mathcal{L}_{\text{S2I}}\&\mathcal{L}_{\text{S2C}}$) with centroid-based learning ($\mathcal{L}_{\text{Base}}$) under K-means setting, we clearly observe that $\mathcal{L}_{\text{Base}}$ consistently surpasses our $\mathcal{L}_{\text{S2I}}\&\mathcal{L}_{\text{S2C}}$. Taking Market as an example, $\mathcal{L}_{\text{Base}}$ outperforms our $\mathcal{L}_{\text{S2I}}\&\mathcal{L}_{\text{S2C}}$ by +18.2% (mAP), +24.3% (R1), respectively. It is because that K-means assumes that the data is the spherical distribution, so that the mean feature with momentum updating as proxy is better than using a randomly picked proxy. 2) Second, comparing DBSCAN with K-means under the same method, we find that DBSCAN performs better than K-means on all results. This indicates that DBSCAN possesses a stronger clustering ability than K-means. In addition, our method consistently outperforms the centroid-based methods when using DBSCAN, clearly verifying the analyses in Section III that DBSCAN manner is more appropriate on the manifold data. For intuitive illustration, we also draw the behaviors of different clustering methods with different memory updating strategies in Figure 7. It clearly shows that DBSCAN is better



(a) Baseline+K-means (b) Baseline+DBSCAN (c) RTMem+DBSCAN (Ours)

Fig. 7. Behaviors of different clustering methods *v.s.* different memory updating strategies. The baseline method is the centroid-based method with the momentum updating strategy. Comparing with (a) and (b), DBSCAN clustering algorithm performs better than the K-means clustering method. Comparing with (b) and (c), our RTMem achieves better feature representation than the momentum updating strategy. Here, different shapes (& colors) represent different ground-truth identities.

Fig. 8. Visualization comparison with ICE [19] and our RTMem on top five retrieved images on Market1501. Red boxes indicate false results, while green boxes represent correct results.

TABLE IV

THE COMPARISONS OF MODEL PARAMETERS, COMPUTATIONAL EFFORT AND TRAINING (TESTING) TIME. FLOPS§ REPRESENTS THE CALCULATED AMOUNT DURING TRAINING PHASE, WHILE FLOPS† INDICATES COMPUTATIONAL EFFORT GENERATED BY POST-PROCESSING

Method	Training time	Testing time	Params	FLOPs§	FLOPs†
SpCL [16]	2.73h	3.18s	23.52M	4.09G	1.43M
ClusterContrast [15]	2.65h	3.18s	23.52M	4.09G	1.43M
Ours	3.47h	3.18s	23.52M	4.09G	4.27M

than K-means and our RTMem outperforms the baseline by a large margin.

4) Ablation on the Choice of the Cluster Proxy: In Table I, we have shown that the proposed RTMem outperforms the baseline of using mean features as the cluster proxy. In this section, we compare the performance of several variants of the proposed sample-to-cluster loss by replacing the cluster proxy with the hardest sample or the easiest sample in the corresponding cluster. The cluster proxy is used to update cluster-level memory bank $\mathcal{M}_C$) on Market-1501 [57], as shown in Table III. From Table III, we find that randomly picking the proxy surpasses other designed proxy choice strategies. The implementation of random samples is also easier as it does not require calculating external hardest or easiest samples. We speculate that the reason for its superior performance is two-fold. On the one hand, when the averaging operation of features within cluster is employed in an off-the-shelf manner, cluster centroids fail to depict the true semantic similarity on the feature space. Compared with it, designed proxy choice strategies only provide relatively reliable optimization direction for ReID model with the hardest and easiest samples, respectively. On the other hand, randomly sampling image features as the timely feature could cover more semantic information and effectively leverage the sample relation to enhance the representation ability of the ReID model. Thus, such a strategy brings about higher ReID accuracy than others. Furthermore, the proposed

TABLE V

COMPARISON WITH STATE-OF-THE-ART METHODS ON PERSON REID TASK, INCLUDING UNSUPERVISED DOMAIN ADAPTATION, FULLY UNSUPERVISED, AND SUPERVISED METHODS. GAP REPRESENTS EMPLOYING THE GLOBAL AVERAGE POOLING. GEM DENOTES TO EMPLOY THE GENERALIZED MEAN POOLING. AGNOSTIC MEANS THE CAMERA INFORMATION IS UNAVAILABLE, WHILE AWARE MEANS THE AVAILABLE CAMERA INFORMATION

Method	Reference	Market1501				MSMT17			
		mAP	R1	R5	R10	mAP	R1	R5	R10
Supervised									
PCB [1]	ECCV'18	81.6	93.8	97.5	98.5	40.4	68.2	-	-
DG-Net [2]	CVPR'19	86.0	94.8	-	-	52.3	77.2	-	-
Unsupervised Domain Adaptation									
MMCL [29]	CVPR'20	60.4	84.4	92.8	95.0	16.2	43.6	54.3	58.9
JVTC [10]	ECCV'20	61.1	83.8	93.0	95.2	20.3	45.4	58.4	64.3
DG-Net++ [11]	ECCV'20	61.7	82.1	90.2	92.7	22.1	48.8	60.9	65.9
ECN+ [66]	TPAMI'20	63.8	84.1	92.8	95.4	16.0	42.5	55.9	61.5
MMT [14]	ICLR'20	71.2	87.7	94.9	96.9	23.3	50.1	63.9	69.8
DCML [67]	ECCV'20	72.6	87.9	95.0	96.7	-	-	-	-
MEB [9]	ECCV'20	76.0	89.9	96.0	97.5	-	-	-	-
SpCL [16]	NeurIPS'20	76.7	90.3	96.2	97.7	26.8	53.7	65.0	69.8
HCC-MMT [68]	TIP'21	78.9	91.2	96.7	97.9	25.2	51.8	64.7	70.5
HCD [18]	ICCV'21	80.0	91.5	-	-	29.3	56.1	-	-
CCL [33]	ICCV'21	82.2	93.6	-	-	32.7	62.7	-	-
IDM [69]	ICCV'21	82.8	93.2	97.5	98.1	33.5	61.3	73.9	78.4
Fully Unsupervised									
BUC [12]	AAAI'19	29.6	61.9	73.5	78.2	-	-	-	-
SSL [8]	CVPR'20	37.8	71.7	83.8	87.4	-	-	-	-
JVTC [10]	ECCV'20	41.8	72.9	84.2	88.7	15.1	39.0	50.9	56.8
MMCL [29]	CVPR'20	45.5	80.3	89.4	92.3	11.2	35.4	44.8	49.8
HCT [18]	CVPR'20	56.4	80.0	91.6	95.2	-	-	-	-
CycAs [70]	ECCV'20	64.8	84.8	-	-	26.7	50.1	-	-
GCL [24]	CVPR'21	66.8	87.3	93.5	95.5	21.3	45.7	58.6	64.5
SpCL [16]	NeurIPS'20	73.1	88.1	95.1	97.0	19.1	42.3	55.6	61.2
RLCC [17]	ICCV'21	77.7	90.8	96.3	97.5	27.9	56.5	68.4	73.1
HCD [38]	ICCV'21	78.1	91.1	96.4	97.7	26.9	53.7	65.3	70.2
ICE [19]	ICCV'21	79.5	92.0	97.0	98.1	29.8	59.0	71.7	77.0
ClusterContrast [15]	Arxiv'21	80.9	91.9	96.9	97.8	22.8	49.4	60.7	65.8
PPLR [71]	CVPR'22	81.5	92.8	97.1	98.1	31.4	61.1	73.4	77.8
CACL [72]	TIP'22	80.9	92.7	97.4	98.5	23.0	48.9	61.2	66.4
RTMem	This Paper	83.0	92.8	97.4	98.3	32.8	57.1	70.0	74.9
ClusterContrast [15](GEM)	Arxiv'21	82.1	92.3	96.7	97.9	27.6	56.0	66.8	71.5
RTMem(GEM)	This Paper	**86.5**	**94.3**	**97.9**	**98.5**	**38.5**	**63.3**	**75.4**	**79.6**
Camera-aware									
CAP [36]	AAAI'21	79.2	91.4	96.3	97.7	36.9	67.4	78.0	81.4
ICE [19]	ICCV'21	82.3	93.8	97.6	**98.4**	38.9	70.2	80.5	84.4
RTMem	This Paper	**83.1**	**93.9**	**97.7**	**98.4**	**40.8**	**72.0**	**81.5**	**84.6**

TABLE VI

COMPARISON WITH THE REID METHODS ON VERI DATASET

Methods	VeRi			
	mAP	R1	R5	R10
MMT [14]	35.3	74.6	82.6	87.0
SpCL [16]	38.9	80.4	86.8	89.6
ClusterContrast [15]	38.1	80.8	85.7	87.9
RTMem	**41.8**	**81.6**	**87.0**	**90.7**
ClusterContrast [15] + GEM pooling	40.2	83.8	87.7	89.9
RTMem + GEM pooling	**44.2**	**85.2**	**89.6**	**92.0**

sample-to-instance loss can boost the performance for all kinds of proxies, indicating the generalization ability of the proposed $\mathcal{L}_{\text{S2I}}$.

5) Discussion on Momentum-Based Memory Updating: As shown in Figure 5, the performance of the baseline is sensitive to the momentum coefficient (*i.e.*, controlling the feature updating rate). The result is consistent with [15], and [16]. As a result, it usually needs to carefully tune the hyper-parameter to get the best results. However, the inconsistency between the clustering and the memory updating still exists. In contrast, our RTMem directly replaces the memory proxies with the up-to-date features to ensure the original data distribution, resulting in better performance. In addition, the random sampling scheme in our RTMem is more robust when using momentum updating since we maintain the original data rather than the fake proxy of the mean feature.

Figure 6 shows the influences of different values of λ in our method. We can see that as λ increases, the model performance gradually becomes better. The best result is achieved with the parameter $\lambda = 1.2.$, but model performance begins to degrade again after $\lambda = 1.2$. Overall, our approach is not very sensitive to parameter and consistently achieves comparable performance in the range.

Table IV presents the comparisons of model parameters, computational effort and training (testing) time. From the results of Table IV, our method increases some additional computational effort and consequently training time compared to other methods (e.g, SpCL [16] and ClusterContrast [15]),

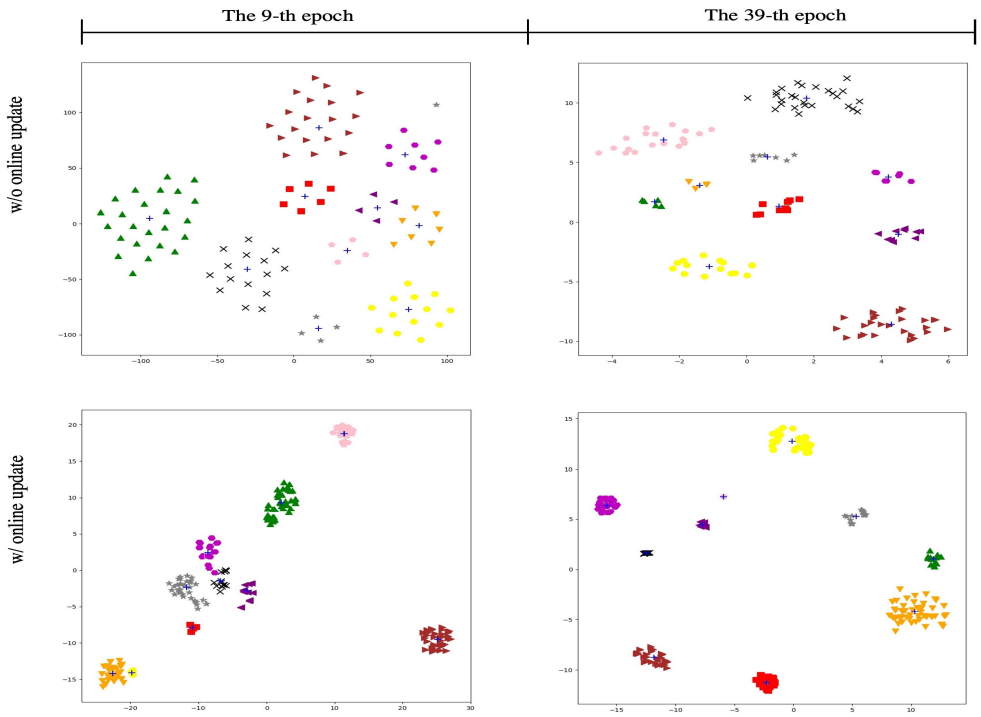


Fig. 9. The change process of cluster centroids (i.e. feature means). The top row shows the results of cluster centroids updated with momentum mechanism, while the bottom row presents the results of clusters proxy with online update. For the above results, we calculate mean feature from the same pseudo cluster as cluster centroids and highlight them with blue "+". Different colors represents the different clusters. Zoom in for a better view of the results.

but achieves the significant improvement on identification accuracy. Additionally, our method maintains the same inference efficiency as other methods in the inference phase.

B. Comparison With State-of-the-Art Methods

In this section, we compare the proposed RTMem with other state-of-the-art methods on Market-1501, DukeMTMC-reID and MSMT17. The quantity results are in Table V and the quality results are in Figure 8.

1) Comparison With UDA Methods: The existing UDA re-ID methods usually adopt labeled source data to reduce label noise. Benefitting from the prior knowledge of external labeled data, the results of UDA methods are commonly superior to fully unsupervised methods. For instance, SpCL [16] obtains 76.7% mAP when using an external source domain when tested on Market-1501, surpassing fully unsupervised SpCL [16] by 3.6% mAP. Despite the success on UDA, our RTMem can still outperform the current SOTA UDA methods when using the same backbone and pooling operation (*i.e.*, ResNet-50 and GAP). Interestingly, our RTMem is even superior than IDM [69], especially on DukeMTMC-reID, with only unlabeld data. The main reason is that the great progress of UDA methods heavily rely on an assumption that the distribution gap between source and target domain is not significant. Indeed, having labeled source data for unsupervised ReID might be a sub-optimal solution. On the contrary, our method focuses more on unlabeled dataset and fully exploits identity information from unsupervised clustering, therefore owning unlimited scalability.

2) Comparison With Fully Unsupervised Methods: Under the fully unsupervised setting, our RTMem also consistently outperforms the existing methods on all three datasets. Specifically, as a milestone, SpCL [16] treats each cluster and outlier as a single class while performing contrastive learning based on a memory containing unlabeled image features. Along this direction, a series of works [15], [17], [38] have achieved better performance by refining pseudo labels or designing hard instance contrast. Relying on a simple real-time memory updating strategy, our RTMem surpasses the prior approaches. The improvement is further enlarged with GEM pooling. Compared with camera-aware methods [19], [36], our RTMem (GAP-agnostic) is camera-agnostic and still outperforms them on Market1501 and DukeMTMC-reID. When camera information is available, our RTMem (GAP-aware) further enlarges the performance lead and outperforms ICE [19]. Figure 8 shows the visualization on the top five retrieval results of ICE [19] and our RTMem . The results show that our method can focus on more detailed information since the real-time memory updating strategy can well maintain the original data distribution.

3) Comparison With Supervised Methods: Finally, we present two supervised methods for reference, including PCB [1] and DG-Net [2]. We also report the performance of our backbone network trained with different GAP and GEM, which indicates the compatibility of our methods with different pooling operations. It can be observed that our unsupervised model (RTMem) consistently surpasses PCB [1] and reduces the gap with DG-Net [2] on two standard benchmarks (i.e. Market1501 and DukeMTMC-reID). Further, with the GEM pooling, our method surpasses DG-Net [2] by

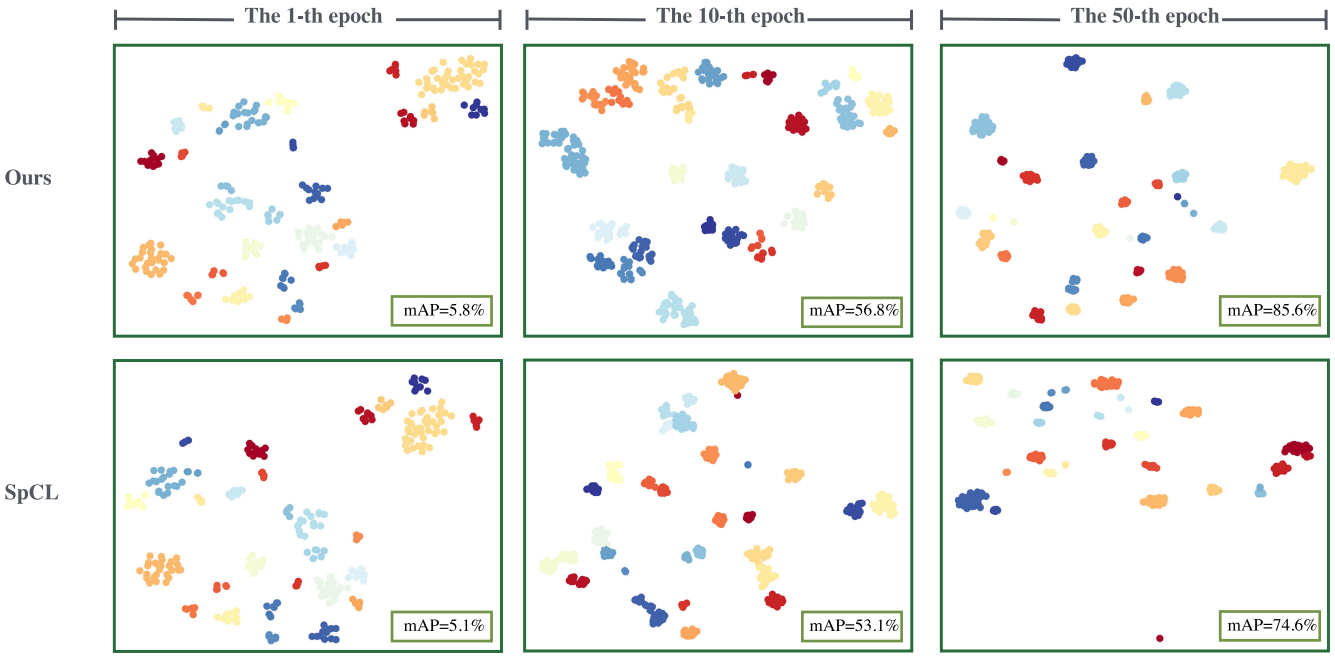


Fig. 10. The visualization of t-SNE on 30 randomly sampled clusters from the Market-1501 training set. Different colors represent different clusters. The top and the bottom figures represent the change of the sampled clusters along with the training process in our RTMem and the baseline model, respectively. Figure shows that the variation of the feature distribution in our RTMem is more stable than that of the baseline. Our method finally achieves more compact and correct clusters compared to the baseline scheme.

0.5% and 1.2% mAP on Market1501 and DukeMTMC-reID, respectively. On a larger and more challenging dataset, i.e., MSMT17, we observe that our approach still mitigates the gap with PCB [1].

To show the robustness of our approach, we conduct experiments with the real-world vehicle datasets (i.e., VeRi-776). As shown in Table VI, our proposed method outperforms prior state-of-the-art methods on VeRi dataset with +3.7% and +4.0% of mAP than ClusterContrast on GAP and GEM poolings, respectively.

4) More Visualization: To better show the effectiveness of our RTMem along with the training process, we visualize the feature distribution of 30 randomly sampled clusters using t-SNE [73]. Different color points represent clusters with different pseudo labels generated by DBSCAN [34]. Note that t-SNE can not show the true feature distribution since the data points distribute in a manifold. But t-SNE can still reflect the transformation of the data during the training process. From Figure 10, it can be seen that the distribution of same cluster is highly variable and unbalanced at the early phase of training. Our RTMem, using the real-time memory updating strategy, does not group the features of the same pseudo-label into one cluster as quickly as the baseline does, which uses the clustering center with momentum as a proxy. At the 10-th epoch, the feature distribution of the same pseudo-label still preserves certain shape in our method, while the baseline model achieves more compact clusters. However, our method outperforms the baseline model by 3.7% mAP for identification accuracy. As time goes on, our RTMem gradually compacts the samples within the same pseudo clusters. At the end of the training, our method achieves not only the best performance, but also the optimal intra-cluster compactness and inter-cluster separation.

We have presented the cluster centroids at different epochs during training, highlighting the changes in their positions with online update. The top and bottom rows of Figure 9 illustrates the changes in cluster centroids obtained without and with online update. The results show that more compact feature distributions are achieved when online update is added. Without online update, the cluster centroids almost appear in the center of each cluster and may group person images with different identities into the same pseudo clusters, which contain multiple person identities. Our method aligns the current image feature into its randomly sampled cluster feature with online update, rather than feature center points, This only provides a relatively reliable optimization direction for the model and enhances the representation ability of model progressively.

Figure 9 shows that how the cluster centroids change at different epochs during training when the online update is added. Top and bottom rows show the change process of clusters obtained without and with online update, respectively. From the results of Figure 9, we can observe that the more compact feature distributions are achieved when online update is added. With cluster centroid as cluster proxy, features within the same pseudo labels are grouped into their clusters. However, these pseudo labels are not correct, and the same pseudo labels may contain different person identities. On the contrary, our method can alleviate the problem with randomly sampled features as cluster proxy. This equally treats features within the same pseudo clusters as cluster centroids and provides a relatively reliable proxy for model optimization, rather than converging into unreliable cluster centroids.

V. Conclusion

Our work directly addresses deficiencies of prior work that uses the inaccurate cluster proxies and the momentum updating strategy to unsupervised representation learning. First, a real-time memory updating strategy (RTMem) is proposed for unsupervised person ReID task. With the RTMem, we can directly replace the feature stored in the memory bank with a random feature sampled from the current mini-batch without momentum updating. And thus we could effectively enforce two kinds of contrastive learning for unlabeled images into the network training process, including the sample-to-instance loss and the sample-to-cluster loss. Experimental results demonstrate that representing each sample and cluster with RTMem produces better identification accuracy than the previously-dominant approaches of the centroid-based method with the momentum updating strategy.

References

[1] Y. Sun, L. Zheng, Y. Yang, Q. Tian, and S. Wang, "Beyond part models: Person retrieval with refined part pooling (and a strong convolutional baseline)," in *Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 480–496.

[2] Z. Zheng, X. Yang, Z. Yu, L. Zheng, Y. Yang, and J. Kautz, "Joint discriminative and generative learning for person re-identification," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 2138–2147.

[3] L. Wei et al., "SIF: Self-inspirited feature learning for person re-identification," *IEEE Trans. Image Process.*, vol. 29, pp. 4942–4951, 2020.

[4] J. Yin, J. Xie, Z. Ma, and J. Guo, "MPCCL: Multiview predictive coding with contrastive learning for person re-identification," *Pattern Recognit.*, vol. 129, Sep. 2022, Art. no. 108710.

[5] R. Quan, X. Dong, Y. Wu, L. Zhu, and Y. Yang, "Auto-ReID: Searching for a part-aware ConvNet for person re-identification," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 3750–3759.

[6] M. Li, X. Zhu, and S. Gong, "Unsupervised person re-identification by deep learning tracklet association," in *Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 737–753.

[7] F. Yang et al., "Asymmetric co-teaching for unsupervised cross-domain person re-identification," in *Proc. AAAI Conf. Artif. Intell.*, 2020, pp. 12597–12604.

[8] Y. Lin, L. Xie, Y. Wu, C. Yan, and Q. Tian, "Unsupervised person re-identification via softened similarity learning," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 3390–3399.

[9] Y. Zhai, Q. Ye, S. Lu, M. Jia, R. Ji, and Y. Tian, "Multiple expert brainstorming for domain adaptive person re-identification," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2020, pp. 594–611.

[10] J. Li and S. Zhang, "Joint visual and temporal consistency for unsupervised domain adaptive person re-identification," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2020, pp. 483–499.

[11] Y. Zou, X. Yang, Z. Yu, B. V. Kumar, and J. Kautz, "Joint disentangling and adaptation for cross-domain person re-identification," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2020, pp. 87–104.

[12] Y. Lin, X. Dong, L. Zheng, Y. Yan, and Y. Yang, "A bottom-up clustering approach to unsupervised person re-identification," in *Proc. AAAI*, vol. 33, Aug. 2019, pp. 8738–8745.

[13] X. Zhang, J. Cao, C. Shen, and M. You, "Self-training with progressive augmentation for unsupervised cross-domain person re-identification," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 8222–8231.

[14] Y. Ge, D. Chen, and H. Li, "Mutual mean-teaching: Pseudo label refinery for unsupervised domain adaptation on person re-identification," in *Proc. Int. Conf. Learn. Represent.*, 2020, pp. 1–15.

[15] Z. Dai, G. Wang, W. Yuan, X. Liu, S. Zhu, and P. Tan, "Cluster contrast for unsupervised person re-identification," 2021, *arXiv:2103.11568*.

[16] Y. Ge, F. Zhu, D. Chen, R. Zhao, and H. Li, "Self-paced contrastive learning with hybrid memory for domain adaptive object re-ID," in *Proc. Adv. Neural Inf. Process. Syst.*, 2020, pp. 11309–11321.

[17] X. Zhang, Y. Ge, Y. Qiao, and H. Li, "Refining pseudo labels with clustering consensus over generations for unsupervised object re-identification," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 3436–3445.

[18] K. Zeng, M. Ning, Y. Wang, and Y. Guo, "Hierarchical clustering with hard-batch triplet loss for person re-identification," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 13657–13665.

[19] H. Chen, B. Lagadec, and F. Bremond, "ICE: Inter-instance contrastive encoding for unsupervised person re-identification," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 14960–14969.

[20] L. Wei, S. Zhang, W. Gao, and Q. Tian, "Person transfer GAN to bridge domain gap for person re-identification," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 79–88.

[21] W. Deng, L. Zheng, Q. Ye, G. Kang, Y. Yang, and J. Jiao, "Image-image domain adaptation with preserved self-similarity and domain-dissimilarity for person re-identification," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 994–1003.

[22] Y. Huang, Q. Wu, J. Xu, and Y. Zhong, "SBSGAN: Suppression of inter-domain background shift for person re-identification," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 9527–9536.

[23] Q. Yang, H.-X. Yu, A. Wu, and W.-S. Zheng, "Patch-based discriminative feature learning for unsupervised person re-identification," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 3633–3642.

[24] H. Chen, Y. Wang, B. Lagadec, A. Dantcheva, and F. Bremond, "Joint generative and contrastive learning for unsupervised person re-identification," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 2004–2013.

[25] L. Song et al., "Unsupervised domain adaptive re-identification: Theory and practice," *Pattern Recognit.*, vol. 102, Jun. 2020, Art. no. 107173.

[26] Y. Fu et al., "Self-similarity grouping: A simple unsupervised cross domain adaptation approach for person re-identification," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 6112–6121.

[27] Z. Zhong, L. Zheng, Z. Luo, S. Li, and Y. Yang, "Invariance matters: Exemplar memory for domain adaptive person re-identification," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 598–607.

[28] H.-X. Yu, W.-S. Zheng, A. Wu, X. Guo, S. Gong, and J.-H. Lai, "Unsupervised person re-identification by soft multilabel learning," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 2148–2157.

[29] D. Wang and S. Zhang, "Unsupervised person re-identification via multi-label classification," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 10981–10990.

[30] H. Ji, L. Wang, S. Zhou, W. Tang, N. Zheng, and G. Hua, "Meta pairwise relationship distillation for unsupervised person re-identification," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 3661–3670.

[31] J. Yin, S. Zhang, J. Xie, Z. Ma, and J. Guo, "Unsupervised person re-identification via simultaneous clustering and mask prediction," *Pattern Recognit.*, vol. 126, Jun. 2022, Art. no. 108568.

[32] Y. Zhai et al., "AD-Cluster: Augmented discriminative clustering for domain adaptive person re-identification," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 9021–9030.

[33] T. Isobe, D. Li, L. Tian, W. Chen, Y. Shan, and S. Wang, "Towards discriminative representation learning for unsupervised person re-identification," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 8526–8536.

[34] M. Ester et al., "A density-based algorithm for discovering clusters in large spatial databases with noise," in *Proc. ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 1996, pp. 226–231.

[35] R. Zhou, X. Chang, L. Shi, Y.-D. Shen, Y. Yang, and F. Nie, "Person reidentification via multi-feature fusion with adaptive graph learning," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 31, no. 5, pp. 1592–1601, May 2020.

[36] M. Wang, B. Lai, J. Huang, X. Gong, and X.-S. Hua, "Camera-aware proxies for unsupervised person re-identification," in *Proc. AAAI Conf. Artif. Intell.*, 2021, pp. 2764–2772.

[37] Z. Hu, Y. Sun, Y. Yang, and J. Zhou, "Divide-and-regroup clustering for domain adaptive person re-identification," in *Proc. AAAI Conf. Artif. Intell.*, 2022, vol. 36, no. 1, pp. 980–988.

[38] Y. Zheng et al., "Online pseudo label generation by hierarchical cluster dynamics for adaptive person re-identification," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 8371–8381.

[39] S. Lloyd, "Least squares quantization in PCM," *IEEE Trans. Inf. Theory*, vol. IT-28, no. 2, pp. 129–137, Mar. 1982.

[40] J. Yang, D. Parikh, and D. Batra, "Joint unsupervised learning of deep representations and image clusters," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 5147–5156.

[41] J. Chang, L. Wang, G. Meng, S. Xiang, and C. Pan, "Deep adaptive image clustering," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 5879–5887.

[42] M. Caron, P. Bojanowski, A. Joulin, and M. Douze, "Deep clustering for unsupervised learning of visual features," in *Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 132–149.

[43] J. Li, P. Zhou, C. Xiong, and S. C. Hoi, "Prototypical contrastive learning of unsupervised representations," in *Proc. Int. Conf. Learn. Represent.*, 2021, pp. 1–16.

[44] X. Zhang et al., "Implicit sample extension for unsupervised person re-identification," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 7369–7378.

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

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

[47] X. Wang, H. Zhang, W. Huang, and M. R. Scott, "Cross-batch memory for embedding learning," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 6388–6397.

[48] O. Vinyals, C. Blundell, T. Lillicrap, K. Kavukcuoglu, and D. Wierstra, "Matching networks for one shot learning," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 29, 2016, pp. 3630–3638.

[49] Z. Wu, A. A. Efros, and S. X. Yu, "Improving generalization via scalable neighborhood component analysis," in *Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 685–701.

[50] Q. Meng, S. Zhao, Z. Huang, and F. Zhou, "MagFace: A universal representation for face recognition and quality assessment," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 14225–14234.

[51] J. Deng, J. Guo, J. Yang, A. Lattas, and S. Zafeiriou, "Variational prototype learning for deep face recognition," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 11906–11915.

[52] M. U. Gutmann and A. Hyvarinen, "Noise-contrastive estimation of unnormalized statistical models, with applications to natural image statistics," *J. Mach. Learn. Res.*, vol. 13, no. 2, pp. 307–361, 2012.

[53] G. Hinton, O. Vinyals, and J. Dean, "Distilling the knowledge in a neural network," 2015, *arXiv:1503.02531*.

[54] X. Wang, L. Zhu, Z. Zheng, M. Xu, and Y. Yang, "Align and Tell: Boosting text-video retrieval with local alignment and fine-grained supervision," *IEEE Trans. Multimedia*, early access, Sep. 5, 2022, doi: 10.1109/TMM.2022.3204444.

[55] X. Wang, L. Zhu, H. Wang, and Y. Yang, "Interactive prototype learning for egocentric action recognition," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 8168–8177.

[56] Z. Hu, C. Zhu, and G. He, "Hard-sample guided hybrid contrast learning for unsupervised person re-identification," in *Proc. 7th IEEE Int. Conf. Netw. Intell. Digit. Content (IC-NIDC)*, Nov. 2021, pp. 91–95.

[57] L. Zheng, L. Shen, L. Tian, S. Wang, J. Wang, and Q. Tian, "Scalable person re-identification: A benchmark," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 1116–1124.

[58] E. Ristani, F. Solera, R. Zou, R. Cucchiara, and C. Tomasi, "Performance measures and a data set for multi-target, multi-camera tracking," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2016, pp. 17–35.

[59] Z. Zhong, L. Zheng, D. Cao, and S. Li, "Re-ranking person re-identification with *k*-reciprocal encoding," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 1318–1327.

[60] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: A large-scale hierarchical image database," in *Proc. IEEE Conf. Comput. Vis. Patt. Recognit.*, Jun. 2009, pp. 248–255.

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

[62] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *Proc. Int. Conf. Mach. Learn.*, 2015, pp. 448–456.

[63] M. Saran, F. Nar, and A. N. Saran, "Perlin random erasing for data augmentation," in *Proc. 29th Signal Process. Commun. Appl. Conf. (SIU)*, Jun. 2021, pp. 13001–13008.

[64] A. Hermans, L. Beyer, and B. Leibe, "In defense of the triplet loss for person re-identification," 2017, *arXiv:1703.07737*.

[65] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *Proc. Int. Conf. Learn. Represent.*, 2014, pp. 1–15.

[66] Z. Zhong, L. Zheng, Z. Luo, S. Li, and Y. Yang, "Learning to adapt invariance in memory for person re-identification," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 43, no. 8, pp. 2723–2738, Aug. 2021.

[67] G. Chen, Y. Lu, J. Lu, and J. Zhou, "Deep credible metric learning for unsupervised domain adaptation person re-identification," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2020, pp. 643–659.

[68] Y. Bai, C. Wang, Y. Lou, J. Liu, and L. Y. Duan, "Hierarchical connectivity-centered clustering for unsupervised domain adaptation on person re-identification," *IEEE Trans. Image Process.*, vol. 30, pp. 6715–6729, 2021.

[69] Y. Dai, J. Liu, Y. Sun, Z. Tong, C. Zhang, and L.-Y. Duan, "IDM: An intermediate domain module for domain adaptive person re-ID," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 11864–11874.

[70] Z. Wang et al., "CycAs: Self-supervised cycle association for learning re-identifiable descriptions," in *Proc. Eur. Conf. Comput. Vis.*, 2020, pp. 72–88.

[71] Y. Cho, W. J. Kim, S. Hong, and S.-E. Yoon, "Part-based pseudo label refinement for unsupervised person re-identification," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 7308–7318.

[72] M. Li, C.-G. Li, and J. Guo, "Cluster-guided asymmetric contrastive learning for unsupervised person re-identification," *IEEE Trans. Image Process.*, vol. 31, pp. 3606–3617, 2022.

[73] L. Van der Maaten and G. Hinton, "Visualizing data using t-SNE," *J. Mach. Learn. Res.*, vol. 9, no. 11, pp. 2579–2605, 2008.