Dual Clustering Co-teaching with Consistent Sample Mining for Unsupervised Person Re-Identification

Zeqi Chen, Zhichao Cui, Chi Zhang, Jiahuan Zhou, Yuehu Liu

***Abstract*—In unsupervised person Re-ID, peer-teaching strategy leveraging two networks to facilitate training has been proven to be an effective method to deal with the pseudo label noise. However, training two networks with a set of noisy pseudo labels reduces the complementarity of the two networks and results in label noise accumulation. To handle this issue, this paper proposes a novel Dual Clustering Co-teaching (DCCT) approach. DCCT mainly exploits the features extracted by two networks to generate two sets of pseudo labels separately by clustering with different parameters. Each network is trained with the pseudo labels generated by its peer network, which can increase the complementarity of the two networks to reduce the impact of noises. Furthermore, we propose dual clustering with dynamic parameters (DCDP) to make the network adaptive and robust to dynamically changing clustering parameters. Moreover, Consistent Sample Mining (CSM) is proposed to find the samples with unchanged pseudo labels during training for potential noisy sample removal. Extensive experiments demonstrate the effectiveness of the proposed method, which outperforms the state-of-the-art unsupervised person Re-ID methods by a considerable margin and surpasses most methods utilizing camera information.**

***Index Terms*—Unsupervised person re-identification, peer-teaching strategy, sample mining.**

I. Introduction

PERSON re-identification (Re-ID) aims to retrieve the images of the same person captured by different cameras [1]. Although supervised person Re-ID [2]–[6] has achieved excellent accuracy on publicly available datasets [7]–[9], the requirement of tremendous manual annotation limits their practicality in the real world. To tackle this issue, unsupervised person Re-ID methods [10]–[14] without any labeled data have been extensively studied.

The mainstream unsupervised methods are clustering-based [10], [11], [13]–[15], which are mainly divided into two stages: (1) generating pseudo labels by clustering; (2) training the network with pseudo labels. Although those methods achieve excellent performance, their generated pseudo labels are inevitably noisy. On the one hand, person images with different identities may have similar appearance, viewpoint, pose, and illumination. Due to subtle differences, they may be clustered into a cluster by clustering algorithms. On the other hand, the images of a person may have occlusion, different resolution, and motion blur. They may be clustered into different clusters due to their distinct differences. Training with noisy pseudo labels hinders the model's performance.

To mitigate the influence of such noisy pseudo labels, a peer-teaching strategy [16]–[20] is employed, which leverages the difference and complementarity of the two networks to filter different noises through cooperative training of the two networks. ACT [21] trains two networks in an asymmetric manner to enhance the complementarity of the two networks. One network is trained with pure samples, while the other is trained with diverse samples. To enhance the output independence of the two networks, MMT [22] utilizes the outputs of the network's temporally average model [19] as soft pseudo labels to train its peer network. However, both ACT and MMT only employ a single clustering to generate a set of noisy pseudo labels for training the two networks, resulting in the accumulation and propagation of pseudo label noise during training. Although AMMT [23] and NRMT [24] adopt different clusterings to improve the quality of pseudo labels, the two networks are trained by the same training data and pseudo labels, which reduces the differences between the two networks and is not conducive to their mutual training.

To overcome the aforementioned shortcomings, we propose a novel Dual Clustering Co-teaching (DCCT) framework to train two networks using two sets of pseudo labels obtained by different clusterings. Training the two networks separately with different data and pseudo labels can increase the differences and complementarity of the two networks, thereby reducing the effect of noises and improving the final performance. Specifically, we propose dual clustering with dynamic parameters (DCDP) to obtain different clustering parameters at each epoch. Then, the features extracted by the temporally average models ($Mean\ Nets$) of two networks are clustered to generate two sets of pseudo labels. And two memory banks are initialized according to the clustering results, as shown in Fig. 1(a). Then we adopt the pseudo labels generated by one network to train its peer network, as shown in Fig. 1(b). In addition, we propose consistent sample mining (CSM) in each mini-batch to discard potential noisy samples with incorrect

This work was supported by the National Key Research and Development Project of New Generation Artificial Intelligence of China under Grant 2018AAA0102504. (*Corresponding author: Yuehu Liu*)

Zeqi Chen, Chi Zhang and Yuehu Liu are with National Key Laboratory of Human-Machine Hybrid Augmented Intelligence, National Engineering Research Center for Visual Information and Applications, and Institute of Artificial Intelligence and Robotics, Xi'an Jiaotong University, Xi'an, China. (e-mail: chenzeqi@stu.xjtu.edu.cn; chizhang@xjtu.edu.cn; liuyh@mail.xjtu.edu.cn)

Zhichao Cui is with Chang'an University, Xi'an, China. (e-mail: cui.zhichao@chd.edu.cn)

Jiahuan Zhou is with the Wangxuan Institute of Computer Technology, Peking University, Beijing, China. (e-mail: jiahuanzhou@pku.edu.cn)

Manuscript received October 6, 2022; revised Feb 13, 2023; accepted Mar 14, 2023.

pseudo labels, which improves the network's performance.

The main contributions of this paper can be summarized as threefold:

- We design a novel peer-teaching framework called Dual Clustering Co-teaching (DCCT), which employs dual clustering with dynamic parameters (DCDP) to generate two sets of pseudo labels. On the one hand, training with different pseudo labels can enhance the differences and complementarity of the two networks and improve their final performance. On the other hand, we dynamically change the clustering parameters of the same network to make the network adaptive and robust to different clustering parameters. The proposed DCDP is so flexible to be effective on multiple clustering algorithms.
- We also propose consistent sample mining (CSM) to discard the samples whose pseudo labels are inconsistent during each training epoch. The discarded inconsistent samples are potential noisy samples that may hinder network training.
- Extensive experiments on three large-scale datasets (Market-1501 [7], MSMT17 [8], and PersonX [9]) demonstrate that our method outperforms the fully unsupervised state-of-the-art methods by a large margin, even surpasses most UDA methods and methods utilizing camera information.

II. Related Works

A. Unsupervised Person Re-ID

Unsupervised person Re-ID methods are mainly divided into unsupervised domain adaptive (UDA) methods and unsupervised learning (USL) methods.

1) UDA Person Re-ID: UDA methods generally pre-train a model using labeled data on the source domain and transfer the learned knowledge from the source domain to the unlabeled target domain. TAL-MIRN [25] leverages triple adversarial learning and multi-view imaginative reasoning to improve the generalization ability of the Re-ID model from the source domain to the target domain. HCN [26] proposes a heterogeneous convolutional network, which leverages CNN and GCN to learn the appearance and correlation information of person images. AD-cluster [27] adopts iterative density-based clustering to generate pseudo labels. It learns an image generator to augment the training samples to enforce the discrimination ability of Re-ID models. To avoid overfitting to noisy pseudo labels, AdaDC [28] adaptively and alternately utilizes different clustering methods. Although these UDA methods perform well under the cross-domain scenario, the requirement of tremendous manually annotation largely limits their usage in practice. In addition, UDA methods rely on the transferable knowledge learned from the source domain, but the discriminative information of the target domain may not be fully explored.

2) USL Person Re-ID: USL methods do not require any labeled data. In recent years, clustering-based methods [13], [14], [29] have become the mainstream of USL methods. In order to improve the discriminative ability of feature similarity and improve clustering performance, IICS [29] decomposes the sample similarity computation into two stages: intra-camera and inter-camera computation. PPLR [13] exploits the complementary relationship between global and local features to reduce pseudo label noise. In order to reduce "sub and mixed" clustering errors, ISE [14] generates support samples around cluster boundaries to associate the same identity samples. Although the clustering-based method has been proven effective and achieves state-of-the-art performance, due to the existence of some indistinguishable persons with similar appearance, the pseudo labels assigned by the clustering method will inevitably be noisy, which will seriously hinder the training of the network.

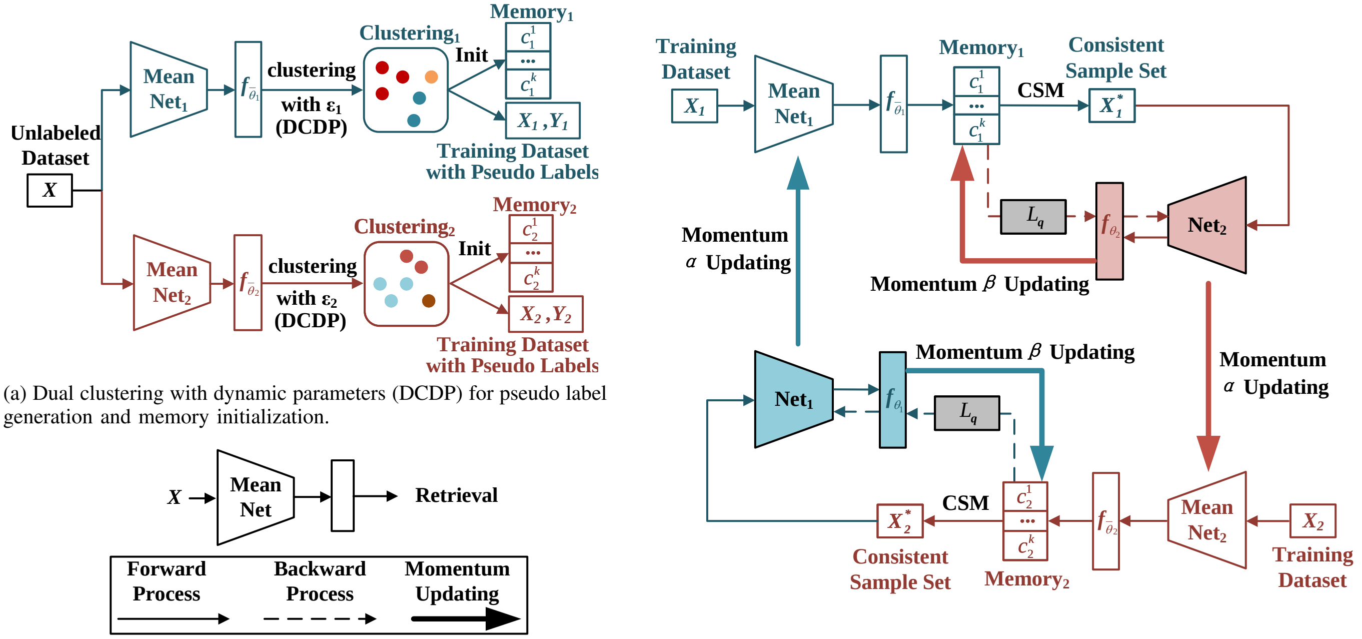
In the latest researches, some contrastive learning based methods have achieved remarkable performances. SpCL [15] stores the features of all instances in the hybrid memory bank [30], [31] and optimizes the encoder with a unified InfoNCE loss [32]. Cluster-Contrast [33] stores features and computes contrastive loss at the cluster level. CAP [34] designs both intra-camera and inter-camera contrastive learning to boost training. ICE [12] employs inter-instance pairwise similarity scores to facilitate contrastive learning. To optimize the feature distribution, CACHE [35] utilizes the instance relationship and cluster relationship to explore the hard samples for contrastive learning. CCL [36] proposes a time-based camera contrastive learning module to promote network training. However, the inevitable pseudo label noise limits the performance of these methods.

B. Learning with Noisy Labels

In recent years, training networks on noisy or unlabeled data has been widely studied, which can be classified into four categories: estimating the noise transition matrix [37], [38], designing the robust loss function [39], [40], correcting the noisy labels [41]–[44] and utilizing peer-teaching strategy [16]–[18].

1) Peer-teaching Strategy: Since the peer-teaching strategy does not require additional clean data and can effectively handle noisy labels on large datasets, this paper focuses on leveraging the peer-teaching strategy method to alleviate label noise. Decoupling [16] trains two networks simultaneously and updates the network only when the predictions of the two networks are different. Co-teaching [17] trains two networks, and each network selects the samples with small losses to train its peer network. Drawing inspiration from Co-teaching and Decoupling, Co-teaching+ [45] adopts small-loss samples with differences between the two networks. Inspired by Co-teaching, Co-mining [18] trains two networks for face recognition tasks, and the clean samples in each mini-batch are re-weighted. Mean teachers [19] average model weights to deal with large datasets and achieve better performance than averaging label predictions [46]. To cope with noise, DivideMix [20] trains two networks simultaneously through dataset co-divide, label co-refinement, and co-guessing.

2) Peer-teaching Strategy for Unsupervised Person Re-ID: Drawing inspiration from Co-teaching, ACT [21] trains two networks in an asymmetric way to tackle unsupervised person Re-ID. However, one of the networks is only trained with clean



(a) Dual clustering with dynamic parameters (DCDP) for pseudo label generation and memory initialization.

(c) One of the Mean Nets is employed for inference.

(b) Consistent sample mining (CSM) and co-teaching.

Fig. 1. The framework of proposed Dual Clustering Co-teaching (DCCT) approach. In order to show the co-teaching process of the two networks more clearly, Net_1 and its results are shown in blue, while Net_2 and its results are shown in red. (a) The features extracted by $Mean\ Net_1$ and $Mean\ Net_2$ are clustered with different parameters ε_1 and ε_2 at each epoch to generate two sets of pseudo labels and initialize two memory banks. ε_1 and ε_2 change dynamically during training, so we call it dual clustering with dynamic parameters (DCDP). (b) Consistent sample mining (CSM) is performed in each iteration. Specifically, $Mean\ Net_1$ and $Memory_1$ are employed to mine consistent samples X_1^* from training dataset X_1. Then X_1^* is adopted to train Net_2. (The training of Net_1 is similar.) The contrastive loss shown in Eq. (7) is used for training. (c) Since the performance of $Mean\ Net$ is better than that of Net, one of the $Mean\ Net$ with better performance is employed for inference. More details are narrated in Algorithm 1.

samples, which limits its generalization capacity. MMT [22] proposes utilizing the temporally average model [19], [31] to generate pseudo labels and soft pseudo labels to avoid training error amplification. AWB [47] proposes a new lightweight attention wave block to enhance the complementarity of the two networks and further suppress the noise in pseudo labels. However, MMT and AWB leverage noisy pseudo labels generated by a single clustering to train two networks simultaneously, which results in noise accumulation and affects the performance of the model. To improve the confidence of pseudo labels, AMMT [23] employs two different clustering methods for mutual teaching. NRMT [24] maintains two networks during training to perform collaborative clustering to obtain two sets of pseudo labels. Although AMMT and NRMT utilize different clusterings to facilitate the training of the two networks, the two networks are trained by the same training data and pseudo labels, which reduces their differences and affects their performance.

III. Dual Clustering Co-teaching (DCCT)

Inspired by previous peer-teaching strategy methods [17], [21], [22], we develop a novel Dual Clustering Co-teaching (DCCT) framework to train two networks separately using two sets of pseudo labels obtained from different clusterings. To increase the difference and independence of the two networks, we follow MMT [22] to employ the temporally average models [19], [31] for our method.

A. The Framework of DCCT

In order to better illustrate the workflow of our method, the two networks are called Net_1 and Net_2 for short, and their temporally average models are called $Mean\ Net_1$ and $Mean\ Net_2$. As shown in Fig. 1, our method mainly contains two stages: (a) pseudo label generation and memory initialization; (b) co-teaching of the two networks.

(a) In the stage of pseudo label generation and memory initialization, we adopt $Mean\ Net_1$ and $Mean\ Net_2$ to extract features from the unlabeled dataset X. Then, different clustering parameters are calculated according to the proposed dual clustering with dynamic parameters (DCDP). After that, the features extracted by $Mean\ Net_1$ and $Mean\ Net_2$ are clustered with different parameters to generate two sets of pseudo labels. And some outliers in X may be discarded according to the clustering results. Then, we get two relatively clean datasets and their pseudo labels: training dataset X_1 with pseudo labels Y_1 and training dataset X_2 with pseudo labels Y_2. At the same time, the two clustering results are exploited to initialize the memory bank [30], [31] $Memory_1$ and $Memory_2$, as shown in Fig. 1(a).

(b) In the stage of co-teaching, we perform consistent sample mining (CSM) in each iteration. Concretely, we leverage $Mean\ Net_1$ and $Memory_1$ to mine consistent sample set X_1^* from the training dataset X_1, and X_1^* is employed to train Net_2. Similarly, the consistent sample set X_2^* is employed to train Net_1, as shown in Fig. 1(b). Net_1 and Net_2 are trained by the contrastive loss shown in Eq. (7). $Memory_1$

and $Memory_2$ are updated by the momentum update strategy shown in Eq. (8), while $Mean\ Net_1$ and $Mean\ Net_2$ are updated by Eq. (9).

Compared with previous methods, we mainly made two contributions: (1) In stage (a), we proposed dual clustering with dynamic parameters (DCDP) to promote network training by generating two sets of pseudo labels (Sec. III-B). (2) In stage (b), we proposed consistent sample mining (CSM) to remove potential noise samples (Sec. III-C). More details of DCCT's procedure are narrated in Algorithm 1.

B. Pseudo Label Generation and Memory Initialization

1) Dual Clustering with Dynamic Parameters (DCDP) for Pseudo Label Generation: Training two networks with one set of pseudo labels suffers from three limitations. (1) Utilizing the same data and supervision to train two networks makes them too similar and lose their complementarity and differences. (2) Leveraging the same noisy pseudo labels to train two networks results in error accumulation and propagation. (3) Using two features to generate a set of pseudo labels may lose some information because it is not easy to find a reasonable and effective way to fuse the features extracted by the two networks.

To handle the aforementioned issues, we propose to train two networks separately with different pseudo labels generated by different clusterings, which can increase their differences and complementarity. Therefore, the samples that cannot be discriminated well by one network may be well discriminated by its peer network, so that the two networks can filter different noises and better collaborative teaching. Furthermore, to make the network adaptive and robust to different clustering parameters, the clustering parameters of the same network can also be dynamically changed to enhance the network's generalization ability. Based on the above considerations, we proposed dual clustering with dynamic parameters (DCDP) for pseudo label generation. Although the proposed DCDP can be combined with multiple clustering algorithms (such as DBSCAN [48], k-means [49], and InfoMap [50]), the DBSCAN is exploited in our framework thanks to its superior ability. The effect of DCDP on other clustering algorithms is demonstrated in Sec. IV-E.

The design details of DCDP. The maximum distance ε between two samples is the most crucial parameter in DBSCAN [48], which is adopted as the dynamic parameter in DCDP. DBSCAN with a smaller ε tends to group samples into more clusters. Conversely, DBSCAN with a larger ε tends to group samples into fewer clusters. In order to increase the complementarity of the two networks, one network always uses pseudo labels obtained by clustering with larger ε, and the other network always utilizes pseudo labels obtained by clustering with smaller ε. Based on this, we design dual clustering with different constant parameters, named **Constant Parameter**, as shown in Eq. (1).

$$\varepsilon_1^i = \varepsilon + \Delta\varepsilon, \quad \varepsilon_2^i = \varepsilon - \Delta\varepsilon, \tag{1}$$

where ε_1^i and ε_2^i denote the maximum distance parameters of $clustering_1$ and $clustering_2$ at the i-th epoch. ε and $\Delta\varepsilon$ are the initial value and the increment size of the maximum distance.

We argue that ε_1^i and ε_2^i should change at each epoch to make the networks adaptive and robust to different clustering parameters. Therefore, we design **Random Parameter**, as shown in Eq. (2).

$$\varepsilon_1^i = \varepsilon + \lambda_1^i\Delta\varepsilon, \quad \varepsilon_2^i = \varepsilon - \lambda_2^i\Delta\varepsilon, \tag{2}$$

where λ_1^i and λ_2^i are two random numbers generated at the i-th epoch. The value of λ_1^i and λ_2^i range from 0 to 1, namely [0,1].

Since the initial performance of the networks is poor, good initial values of the clustering parameters are essential. The Random Parameter often leads to a poor initial value and hinders the training of the network. Therefore, we designed two parameters to gradually distance from the initial value, named **Linear Parameter**, as shown in Eq. (3).

$$\varepsilon_1^i = \varepsilon + \frac{i\Delta\varepsilon}{E}, \quad \varepsilon_2^i = \varepsilon - \frac{i\Delta\varepsilon}{E}, \tag{3}$$

where E is the number of training epochs.

The increasing gap between the two parameters is not conducive to the convergence of the two networks. Therefore, we design ε_1^i and ε_2^i to be consistent again at the end of the training, named **Piecewise Parameter**, as shown in Eq. (4). We compare the impact of the four designs on network performance in Sec. IV-D4.

$$\begin{aligned} \varepsilon_1^i &= \begin{cases} \varepsilon + \dfrac{2\Delta\varepsilon}{E}i, & 0 \le i < \dfrac{E}{2} \\ \varepsilon + 2\Delta\varepsilon - \dfrac{2\Delta\varepsilon}{E}i, & \dfrac{E}{2} \le i \le E \end{cases}, \\ \varepsilon_2^i &= \begin{cases} \varepsilon - \dfrac{2\Delta\varepsilon}{E}i, & 0 \le i < \dfrac{E}{2} \\ \varepsilon - 2\Delta\varepsilon + \dfrac{2\Delta\varepsilon}{E}i, & \dfrac{E}{2} \le i \le E \end{cases}. \end{aligned} \tag{4}$$

At the start of each epoch, $Mean\ Net_1$ and $Mean\ Net_2$ are employed to extract the features of unlabeled dataset X. Then, the extracted features $\boldsymbol{f}_{\bar{\theta}_1}$ and $\boldsymbol{f}_{\bar{\theta}_2}$ are utilized for clustering with parameters ε_1 and ε_2. Since DBSCAN removes outliers, we obtain two relatively clean training datasets X_1 and X_2, and their pseudo labels Y_1 and Y_2.

2) Memory Initialization: As shown in Fig. 1(a), $clustering_1$ is employed to initialize the memory bank [30], [31] $Memory_1$. Following Cluster-Contrast [33], the mean feature vectors of each cluster are adopted to initialize the cluster representations $\{\boldsymbol{c}_1^1, ..., \boldsymbol{c}_1^K\}$, where $\boldsymbol{c}_1^k$ denotes the k-th cluster representation of $Memory_1$ and K is the cluster number. So $Memory_1$ is initialized by:

$$\boldsymbol{c}_1^k = \frac{1}{|C_1^k|} \sum_{\boldsymbol{x}_1^j \in C_1^k} \boldsymbol{f}_{\bar{\theta}_1}(\boldsymbol{x}_1^j), \tag{5}$$

where C_1^k denotes the k-th cluster of $clustering_1$ and $|\cdot|$ indicates the number of instances per cluster. $\boldsymbol{x}_1^j$ denotes the j-th samples in X_1, and $\boldsymbol{f}_{\bar{\theta}_1}(\cdot)$ denotes the features extracted by $Mean\ Net_1$. The initialization of the memory bank $Memory_2$ is similar to that of $Memory_1$.

C. Consistent Sample Mining and Co-teaching

1) Consistent Sample Mining (CSM): Directly using noisy pseudo labels to train the network will reduce its final performance. Therefore, we propose consistent sample mining (CSM) to extract consistent samples and remove potential noise samples. Since the trainings of Net_1 and Net_2 are similar, we only introduce the training details of Net_2 and CSM details of $Mean\ Net_1$ and $Memory_1$.

Inconsistency of pseudo labels. For any sample $\boldsymbol{x}_1^j$ in training datasets X_1, its feature extracted by $Mean\ Net_1$ is denoted as $\boldsymbol{f}_{\bar{\theta}_1}(\boldsymbol{x}_1^j)$. And y_1^j represents the pseudo label of $\boldsymbol{x}_1^j$. At the beginning of each epoch, the features $\boldsymbol{f}_{\bar{\theta}_1}$ extracted by $Mean\ Net_1$ are employed for clustering, and the clustering results are utilized to initialize $Memory_1$ and generate pseudo labels (see Fig. 1(a)). At this time, calculating the similarity between $\boldsymbol{f}_{\bar{\theta}_1}(\boldsymbol{x}_1^j)$ and each cluster representation $\boldsymbol{c}_1^k$ stored in $Memory_1$, $\boldsymbol{f}_{\bar{\theta}_1}(\boldsymbol{x}_1^j)$ will be most similar to the cluster indicated by the pseudo label y_1^j.

In each iteration, the parameters of $Mean\ Net_1$ are updated by parameters of Net_1 with momentum α (see Fig. 1(b) and Eq. (9)), and each clustering representation $\boldsymbol{c}_1^k$ stored in $Memory_1$ is updated by the features extracted by Net_2 with momentum β (see Fig. 1(b) and Eq. (8)), while the pseudo labels are not updated synchronously. At this time, calculating the similarity between each $\boldsymbol{f}_{\bar{\theta}_1}(\boldsymbol{x}_1^j)$ and each $\boldsymbol{c}_1^k$, some samples may be most similar to the clustering representations that are inconsistent with their pseudo labels.

Definition of consistent samples. In each iteration, we calculate the cosine similarity between each sample's feature $\boldsymbol{f}_{\bar{\theta}_1}(\boldsymbol{x}_1^j)$ and each clustering representation $\boldsymbol{c}_1^k$. Then we obtain the most similar cluster k^* of each sample $\boldsymbol{x}_1^j$ by:

$$k^* = \underset{k\in\{1,2,\dots,K\}}{\arg\max}\ sim(\boldsymbol{f}_{\bar{\theta}_1}(\boldsymbol{x}_1^j), \boldsymbol{c}_1^k), \tag{6}$$

where $sim()$ denotes the cosine similarity between two vectors. When k^* is consistent with the pseudo label of $\boldsymbol{x}_1^j$, the sample $\boldsymbol{x}_1^j$ is considered consistent. Otherwise, the sample is considered inconsistent, and its pseudo label is changed to -1.

We argue that inconsistent samples hamper network training, while only consistent samples should be employed for training. The number of consistent samples increases with the gradual convergence of the network (see Fig. 4(a)). Eventually, it tends to exploit all samples for training, which is in accordance with the concept of self-paced learning [51].

Clustering quality evaluation for CSM. However, in the early stage of training, the clustering quality may be poor. Using CSM at this time, the number of consistent samples selected in each iteration may be too small, which impairs network training. Therefore, we propose to employ the Davies-Bouldin index (DBI) [52] to measure the clustering quality. DBI is an internal clustering evaluation scheme without the demand for ground truth. The lower bound of the DBI is 0, and a lower DBI value means a better clustering quality. Therefore, we set a **threshold** γ to judge whether the clustering quality is good. When DBI is less than γ, the clustering quality is considered good enough to mine consistent samples. We analyze how the parameter γ affects the network performance in Sec. IV-H.

Algorithm 1: Procedure of the DCCT.

```
Input: unlabeled dataset $X$; ImageNet pre-trained
       ResNet-50 $\boldsymbol{\theta}$; maximum distance $\varepsilon$ and its
       increment size $\Delta\varepsilon$ for Eq. (4); threshold $\gamma$ for
       clustering quality; temperature $\tau$ for Eq. (7);
       momentum $\beta$ for Eq. (8); momentum $\alpha$ for Eq.
       (9); maximal epoch $E$; maximal iteration $I$.
Output: Best $Mean\ Net$ $\bar{\boldsymbol{\theta}}^*$ after training.
Initialize: $Net_1$ $\boldsymbol{\theta}_1 \leftarrow \boldsymbol{\theta}$, $Net_2$ $\boldsymbol{\theta}_2 \leftarrow \boldsymbol{\theta}$, $Mean\ Net_1$
  $\bar{\boldsymbol{\theta}}_1 \leftarrow \boldsymbol{\theta}_1$, $Mean\ Net_2$ $\bar{\boldsymbol{\theta}}_2 \leftarrow \boldsymbol{\theta}_2$;
for $epoch = 1$ to $E$ do
    Extract feature $\boldsymbol{f}_{\bar{\theta}_1}$ and $\boldsymbol{f}_{\bar{\theta}_2}$ from $X$ by $\bar{\boldsymbol{\theta}}_1$ and $\bar{\boldsymbol{\theta}}_2$;
    Calculate parameters $\varepsilon_1$ and $\varepsilon_2$ for DCDP by Eq.
      (4);
    Perform clustering on $\boldsymbol{f}_{\bar{\theta}_1}$ and $\boldsymbol{f}_{\bar{\theta}_2}$;
    Generate training dataset $X_1$, $X_2$ and their pseudo
      labels $Y_1$, $Y_2$ by two clustering results;
    Initialize $Memory_1$ and $Memory_2$ by Eq. (5);
    Calculate $DBI_1$ and $DBI_2$ of two clusterings;
    for $iter = 1$ to $I$ do
        if $min(DBI_1, DBI_2) < \gamma$ then
            Perform CSM by Eq. (6) to obtain $X_1^*$ and
              $X_2^*$;
        else
            $X_1^* = X_1$, $X_2^* = X_2$;
        end
        Train $\boldsymbol{\theta}_1$ ($\boldsymbol{\theta}_2$) using $X_2^*$ ($X_1^*$) and loss function
          in Eq. (7);
        Update $Memory_1$ and $Memory_2$ by Eq. (8);
        Update $\bar{\boldsymbol{\theta}}_1$ and $\bar{\boldsymbol{\theta}}_2$ by Eq. (9);
    end
end
```

2) Loss Function: For any query instances, its features extracted by Net_2 are recorded as $\boldsymbol{q}$, which is compared to all the cluster representations $\{\boldsymbol{c}_1^1, ..., \boldsymbol{c}_1^K\}$ stored in $Memory_1$ using the following InfoNCE loss [32]:

$$L_{\boldsymbol{q}} = -\log \frac{\exp(\boldsymbol{q}\cdot\boldsymbol{c}_1^+)/\tau}{\sum_{k=1}^{K}\exp(\boldsymbol{q}\cdot\boldsymbol{c}_1^k)/\tau}, \tag{7}$$

where $\boldsymbol{c}_1^+$ is the clustering representation indicated by the pseudo label of the query instance, τ is a temperature hyper-parameter [53].

3) Memory Updating: Following Cluster-Contrast [33], we adopt the momentum update strategy [15], [31] to update $Memory_1$, which is formulated as follows:

$$\boldsymbol{c}_1^k \leftarrow \beta\boldsymbol{c}_1^k + (1-\beta)\boldsymbol{q}^k, \tag{8}$$

where $\boldsymbol{q}^k$ is the sample feature extracted by Net_2, which has the same identity as the cluster representations $\boldsymbol{c}_1^k$, and β is the ensembling momentum to be within the range of $[0, 1)$.

4) Temporally Average Models Updating: Let $\boldsymbol{\theta}^i$ and $\bar{\boldsymbol{\theta}}^i$ denote the parameters of a network and the network's temporally average model at iteration i, then $\bar{\boldsymbol{\theta}}^{i+1}$ can be updated as

$$\bar{\boldsymbol{\theta}}^{i+1} = \alpha\bar{\boldsymbol{\theta}}^i + (1-\alpha)\boldsymbol{\theta}^i, \tag{9}$$

where α is the ensembling momentum to be within the range of $[0, 1)$. The initial parameters of the temporally average model are $\bar{\boldsymbol{\theta}}^0 = \boldsymbol{\theta}^0$.

IV. Experiments

A. Datasets and Evaluation Protocols

1) Datasets.: We evaluate the proposed method on three large-scale datasets - Market-1501 [7], MSMT17 [8], and PersonX [9].

Market-1501 dataset contains 32,668 annotated images of 1,501 identities captured by 6 cameras on a university campus. In Market-1501, 12,936 images of 751 identities are used as the training set, and 19,732 images of 750 identities are utilized as the test set.

MSMT17 dataset contains 126,441 annotated images of 4,101 identities captured by 15 cameras. In MSMT17, 32,621 images of 1,041 identities are used as the training set, and 93,820 images of the remaining 3,060 identities are utilized as the test set.

PersonX is a synthetic dataset with manually designed difficulties such as different viewpoints, illumination, occlusions, and backgrounds. It contains 45,792 annotated images of 1,266 identities captured by 6 cameras. In PersonX, 9,840 images of 410 identities are used as the training set, and 35,952 images of the remaining 856 identities are utilized as the test set.

2) Evaluating Setting: The mean average precision (mAP) [54] and cumulative matching characteristic (CMC) [55] curve are employed to evaluate the performance of each method. And the top-1, top-5, and top-10 accuracies are reported to represent the CMC curve.

B. Implementation Details

Our method is implemented based on PyTorch on Linux. We employ four NVIDIA RTX 2080Ti GPUs for training and only one GPU for testing. We adopt a pre-trained ResNet-50 [57] on ImageNet [58] as the backbone networks to conduct all the experiments. The model is modified following Cluster-Contrast [33]. The Net_1 and Net_2 are initialized by the ImageNet pre-trained model, and they are updated by the loss function shown in Eq. (7) with the temperature hyper-parameter $\tau = 0.05$. The temporal momentum α in Eq. (9) is 0.99. The momentum β for memory updating in Eq. (8) is 0.1. We set the number of training epochs to be 50, and the number of training iterations is 300. During the training, all images are resized to 320×128, and random cropping, flipping as well as random erasing are adopted for data augmentation [59]. The batch size is set to 128, which contains 32 identities, and each identity has 4 images. We employ Adam optimizer to train the model with weight decay 5×10^{-4}. The initial learning rate is 3.5×10^{-4}, which is reduced to 1/10 of its previous value every 20 epochs. Following SpCL [15], we utilize Jaccard distance based on k-reciprocal encoding [60] for clustering, where k_1 is set to 30 and k_2 is set to 6. In DBSCAN, the minimum number of samples in the neighborhood of the core point is set to 4, and the maximum distance ε is set to 0.5, 0.7, and 0.7 for Market-1501, MSMT17, and PersonX. The $\Delta\varepsilon$ in Eq. (4) is set to 0.35, 0.15 and 0.15 for Market-1501, MSMT17 and PersonX. The DBI threshold γ in Sec. III-C is 1.3.

C. Comparison with SOTA methods

In Table I and Table II, we compare our method with several state-of-the-art unsupervised methods on three widely-used person Re-ID datasets (Market-1501, MSMT17, and PersonX). We obtain the best performance among all the compared methods with top-1 =94.4% and mAP =86.3% on Market-1501. On PersonX, we also achieve the best performance with top-1 =95.0%, and mAP =87.6%. Note that on the above two datasets, our method not only surpasses other fully unsupervised methods by considerable margins, but also outperforms other UDA methods that require a large number of manually labeled data on the source domain. On MSMT17, we outperform the state-of-the-art fully unsupervised method ISE [14] by considerable margins of 4.8% mAP. We are also better than all UDA methods. In addition, we also refer to PPLR [13] to add the inter-camera contrastive loss to employ camera information. The results are shown in the middle block in Table I. On Market-1501, although camera information has brought limited improvement, our DCCT still outperforms other well-known unsupervised methods utilizing camera information. On MSMT17, camera information improves performance by 1.9% mAP and 2.3% top-1. Although our method is lower than PPLR on top-1, it has the best mAP. More effective ways to leverage camera information deserve further research.

D. Ablation Study

In this part, we verify the effectiveness of our proposed dual clustering with dynamic parameters (DCDP) and consistent sample mining (CSM). We define the baseline as DCCT without DCDP and CSM. Not adopting DCDP means that $\Delta\varepsilon$ in Eq. (4) is 0, so the two clustering parameters are equal and unchanged during training. Not utilizing CSM means directly using all samples and their pseudo labels for training.

1) Effectiveness of Dual Clustering with Dynamic Parameters: The performances of using DCDP (denoted as "Baseline + DCDP" and "DCCT") or not are shown in Table III. It can be observed that using the proposed DCDP can always obtain better performance on all three datasets. The reason is that the two networks are trained with pseudo labels obtained by different clusterings, which increases the differences and complementarity of the two networks. Therefore, the two networks can cope with different types of noises and improve the final performance. Meanwhile, the dynamically changing clustering parameters also enhance the network's generalization ability.

In Fig. 3(a), we compared the cluster number of the two clusterings with and without DCDP at different epochs on Market-1501. It can be observed that the difference in the number of clusters increases significantly when using DCDP,

TABLE I
COMPARISON OF THE PROPOSED DCCT AND STATE-OF-THE-ART METHODS ON MARKET-1501 AND MSMT17. THE "LABELS" COLUMN LISTS THE TYPE OF LABELS USED BY THE METHOD. "TRANSFER" DENOTES THAT THE MANUALLY ANNOTATED LABELS FROM ANOTHER RE-ID DATASET ARE UTILIZED FOR TRAINING. "CAMERA" MEANS THE CAMERA INFORMATION IS EMPLOYED BY THE METHOD. "NONE" MEANS THAT IT IS A FULLY UNSUPERVISED METHOD. † INDICATES THAT THE RESULTS ARE REPRODUCED BY THE AUTHOR OF MMT [22] IN THE PAPER SPCL [15]. ‡ INDICATES THAT THE RESULTS ARE REPRODUCED BY US. PERFORMANCES SURPASSING ALL COMPETING METHODS ARE **BOLD**, AND THE SECOND-BEST PERFORMANCES ARE HIGHLIGHTED USING UNDERLINE.

Method	Reference	Labels	Market-1501				MSMT17			
			mAP	top-1	top-5	top-10	mAP	top-1	top-5	top-10
TAL-MIRN [25]	TCSVT'22	Transfer	42.9	74.6	87.6	-	14.2	39.0	51.5	-
ACT [21]	AAAI'2020	Transfer	60.6	80.5	-	-	-	-	-	-
AD-Cluster [27]	CVPR'2020	Transfer	68.3	86.7	94.4	96.5	-	-	-	-
HCN [26]	TCSVT'22	Transfer	70.5	90.7	-	-	29.9	58.7	-	-
NRMT [24]	ECCV'2020	Transfer	71.7	87.8	94.6	96.5	20.6	45.2	57.8	63.3
MMT-kmeans [22]	ICLR'2020	Transfer	71.2	87.7	94.9	96.9	23.3	50.1	63.9	69.8
MMT-DBSCAN† [22]	ICLR'2020	Transfer	75.6	89.3	95.8	97.5	24.0	50.1	63.5	69.3
SpCL [15]	NeurIPS'20	Transfer	77.5	89.7	96.1	97.6	26.8	53.7	65.0	69.8
AWB [47]	TIP'22	Transfer	81.0	93.5	97.4	98.3	29.5	61.0	73.5	77.9
AMMT [23]	Access'21	Transfer	83.3	93.2	97.7	98.6	28.4	49.4	62.1	67.4
CACHE [35]	TCSVT'22	Transfer	83.1	93.4	97.5	98.2	31.3	58.0	69.8	74.5
AdaDC [28]	TCSVT'22	Transfer	83.2	92.9	97.5	98.5	32.7	60.7	73.6	78.7
IICS [29]	CVPR'2021	Camera	72.9	89.5	95.2	97.0	26.9	56.4	68.8	73.4
CAP [34]	AAAI'2021	Camera	79.2	91.4	96.3	97.7	36.9	67.4	78.0	81.4
ICE [12]	ICCV'2021	Camera	82.3	93.8	97.6	98.4	38.9	70.2	80.5	84.4
PPLR [13]	CVPR'2022	Camera	84.4	94.3	97.8	98.6	42.2	73.3	83.5	86.5
CCL [36]	TCSVT'23	Camera	85.3	94.1	-	-	41.8	71.4	-	-
DCCT (Ours)	This paper	Camera	86.7	94.4	97.8	98.7	43.7	71.0	81.5	84.6
BUC [10]	AAAI'2019	None	38.3	66.2	79.6	84.5	-	-	-	-
HCT [11]	CVPR'2020	None	56.4	80.0	91.6	95.2	-	-	-	-
MMT-DBSCAN† [22]	ICLR'2020	None	70.8	86.4	95.1	97.2	26.5	53.8	66.4	71.6
SpCL [15]	NeurIPS'20	None	73.1	88.1	95.1	97.0	19.1	42.3	55.6	61.2
ICE [12]	ICCV'2021	None	79.5	92.0	97.0	98.1	29.8	59.0	71.7	77.0
Co-teaching‡ [17]	NIPS'2018	None	82.1	92.2	96.6	97.7	31.5	60.2	72.6	77.4
Cluster-Contrast [33]	ACCV'2022	None	82.1	92.3	96.7	97.9	27.6	56.0	66.8	71.5
PPLR [13]	CVPR'2022	None	81.5	92.8	97.1	98.1	31.4	61.1	73.4	77.8
ISE [14]	CVPR'2022	None	85.3	94.3	**98.0**	**98.8**	37.0	67.6	77.5	81.0
DCCT (Ours)	This paper	None	**86.3**	**94.4**	97.7	98.5	**41.8**	**68.7**	**79.0**	**82.6**

TABLE II
COMPARISON OF THE PROPOSED DCCT AND STATE-OF-THE-ART METHODS ON PERSONX DATASET. † INDICATES THAT THE RESULTS ARE REPRODUCED BY CLUSTER-CONTRAST [33].

Method	Labels	PersonX			
		mAP	top-1	top-5	top-10
MMT-DBSCAN† [22]	Transfer	78.9	90.6	96.8	98.2
SPCL† [15]	Transfer	78.5	91.1	97.8	99.0
SPCL† [15]	None	72.3	88.1	96.6	98.3
Cluster-Contrast [33]	None	84.7	94.4	98.3	99.3
DCCT (Ours)	None	**87.6**	**95.0**	**98.7**	**99.4**

indicating that the two clusterings are more different. In Fig. 3(b), we compare the average cosine similarity between the features extracted by the two $Mean\ Nets$ with and without DCDP on Market-1501. Throughout the training process, the average cosine similarity with DCDP is always smaller than that without DCDP, indicating that DCDP increases the differences between the two networks. Since the learning rate is reduced to 1/10 of its previous value every 20 epochs, the cosine similarity has a noticeable drop every 20 epochs.

2) Effectiveness of Consistent Sample Mining: The performances of using CSM (denoted as "Baseline + CSM" and "DCCT") or not are shown in Table III. It can be observed that using the proposed CSM can always obtain better performance on all three datasets.

Fig. 2 shows the CSM results of the last iteration in the last epoch on the Market-1501. It can be seen that some of the inconsistent samples mined by CSM have wrong pseudo labels (green boxes). Although some samples with correct pseudo labels may be considered inconsistent by CSM (red boxes),

TABLE III
ABLATION STUDY OF OUR PROPOSED DUAL CLUSTERING WITH DYNAMIC PARAMETERS (DCDP) AND CONSISTENT SAMPLE MINING (CSM).

Method	Market-1501				MSMT17				PersonX			
	mAP	top-1	top-5	top-10	mAP	top-1	top-5	top-10	mAP	top-1	top-5	top-10
Baseline	82.9	92.9	97.1	98.1	38.5	65.9	76.8	81.0	86.3	94.4	98.4	99.2
Baseline + DCDP	85.1	93.7	97.5	98.4	40.8	68.0	78.4	82.4	86.4	94.0	98.5	**99.4**
Baseline + CSM	85.1	93.6	97.3	98.0	40.3	67.6	78.7	82.4	86.7	94.4	98.4	**99.4**
DCCT (Baseline + DCDP + CSM)	**86.3**	**94.4**	**97.7**	**98.5**	**41.8**	**68.7**	**79.0**	**82.6**	**87.6**	**95.0**	**98.7**	**99.4**

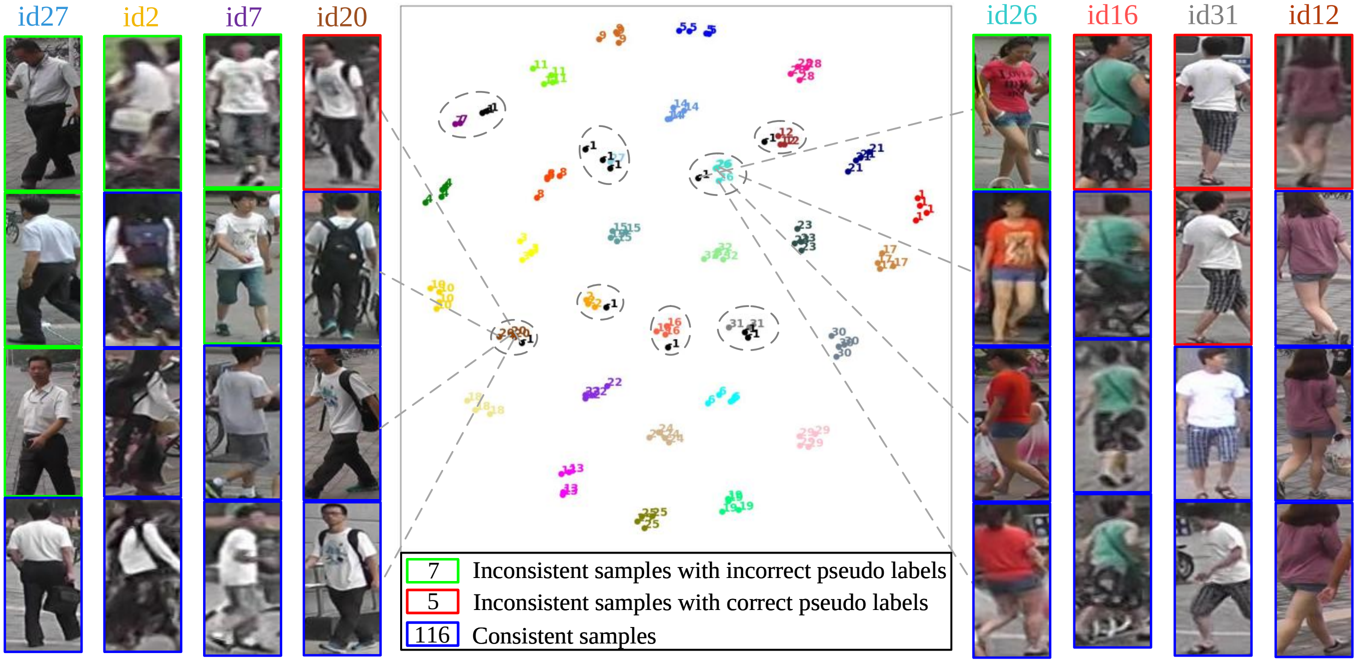
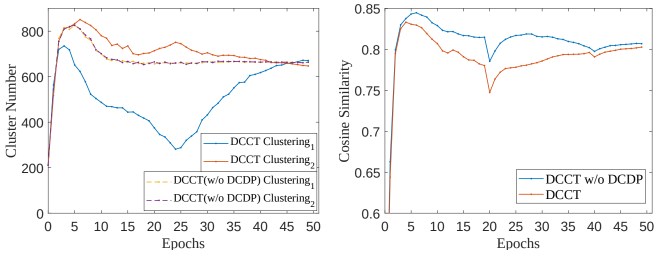
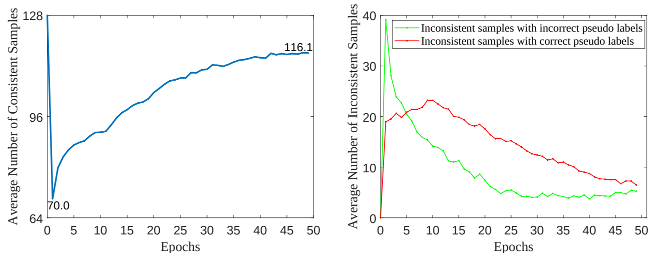


Fig. 2. T-SNE [56] visualization of the learned feature embeddings and their pseudo labels in the last iteration of the last epoch on the Market-1501 (32 identities, each identity with 4 images, total 128 images). The inconsistent samples (red and green boxes) mined by CSM are given a new pseudo label -1 and are not adopted for network training. Only consistent samples (blue boxes) are used for network training.



(a) (b)

Fig. 3. (a) The cluster number of the two clusterings over different epochs with and without DCDP on Market-1501. (b) The average cosine similarity between the features of the two networks at different epochs with and without DCDP on Market-1501.



(a) (b)

Fig. 4. (a) The average number of consistent samples selected in each mini-batch over different epochs on Market-1501. (b) The average number of inconsistent samples with correct and incorrect pseudo labels in each mini-batch over different epochs on Market-1501.

the experiment results in Table III show that training with wrong samples has a greater adverse effect than discarding a portion of the correct samples.

Fig. 4(a) shows the average number of consistent samples selected in each mini-batch at different epochs on Market-1501. It can be observed that the number of selected consistent samples gradually increases as the training progresses. Furthermore, the average numbers of inconsistent samples with correct and incorrect pseudo labels are also illustrated in Fig. 4(b). We can see that in the early stage of training, the vast majority of inconsistent samples have incorrect pseudo labels. Dropping these inconsistent samples can effectively reduce the impact of noises on training. Although about half of the discarded samples have correct pseudo labels, the experimental

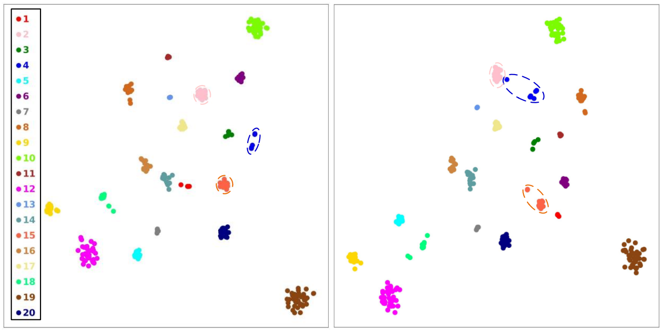
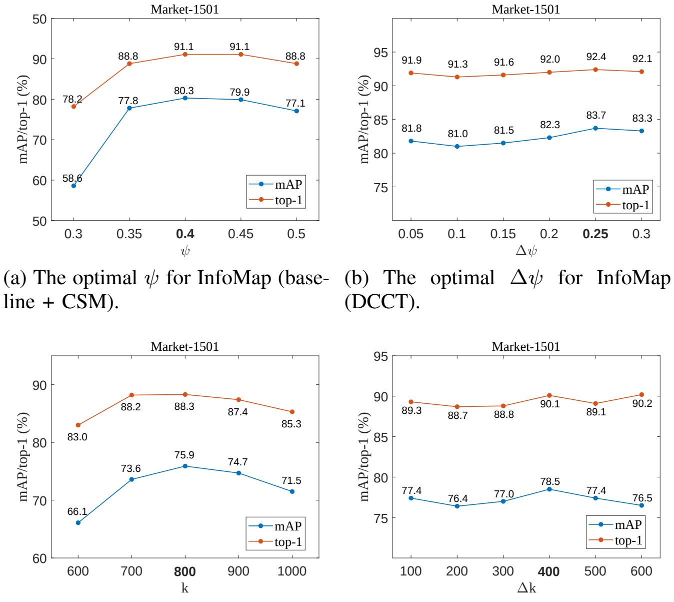


Fig. 5. T-SNE [56] visualization of 20 random identities on Market-1501 between DCCT (Left) and the baseline (Right). Different numbers and colors represent different identities.

TABLE IV
THE COMPARISON OF DIFFERENT DCDP DESIGNS ON MARKET-1501.

Method	mAP	top-1	top-5	top-10
Baseline	82.9	92.9	97.1	98.1
Baseline + Constant Parameter	83.7	93.2	97.3	98.2
Baseline + Random Parameter	83.2	93.1	97.2	98.1
Baseline + Linear Parameter	83.9	93.3	97.3	98.3
Baseline + Piecewise Parameter (DCDP)	85.1	93.7	97.5	98.4



(a) The optimal ψ for InfoMap (baseline + CSM).

(b) The optimal $\Delta\psi$ for InfoMap (DCCT).

(c) The optimal cluster number k for k-means (baseline + CSM).

(d) The optimal Δk for k-means (DCCT).

Fig. 6. The effectiveness of DCDP on other clustering algorithms.

results in Table III show that the negative impact of discarding some correct samples is smaller than the positive impact of discarding those incorrect samples.

3) Qualitative Analysis of T-SNE Visualization: To further illustrate that the proposed DCDP and CSM can improve the model's discriminative ability, we employ T-SNE to visualize the feature embeddings of person images with 20 random identities on Market-1501. As shown in Fig. 5, after employing DCDP and CSM, the feature distribution of persons with the same identity is more compact. Furthermore, there is less mixing and overlap among person features with different identities.

4) The Impact of Different DCDP Designs: The results of four DCDP designs are shown in Table IV. It can be observed that all designs can improve the network's performance because they can increase the differences and complementarity of the two networks. When adopting the Piecewise Parameter, we get the best result with the mAP = 85.1% and top-1 = 93.7%. A slightly lower performance can be observed when utilizing the Linear Parameter because the large parameter difference hinders the convergence of the two networks. The performance with Constant Parameter is slightly lower than that with Linear Parameter. The reason is that constant clustering parameters reduce the network's generalization ability. The Random Parameter has the lowest performance because it is difficult to randomize to a good initial value.

E. The Effectiveness of DCDP on Other Clustering Algorithms

This experiment aims to demonstrate that the proposed DCDP is applicable for not only DBSCAN but also InfoMap [50] and k-means [49].

1) InfoMap: To use InfoMap for clustering, we need to convert all samples into a directed graph, where nodes are samples. Let $D(i, j)$ represents the distance between any two samples, we link the two nodes when $D(i, j)$ is less than maximum distance ψ. And the weight of the link is represented as $1 - D(i, j)$. We adopt ψ as the dynamic parameter in DCDP. Given the initial value ψ and the increment size $\Delta\psi$ of the maximum distance, the maximum distance ψ_1 and ψ_2 of $clustering_1$ and $clustering_2$ will vary in the range of $[\psi - \Delta\psi,\ \psi + \Delta\psi]$. Then the ψ_1^i and ψ_2^i at the i-th epoch can be obtained from Eq. (4). (Replace ε with ψ in Eq. (4).) As shown in Fig. 6(a), the best ψ on Market-1501 is 0.4. After tuning $\Delta\psi$ in Fig. 6(b), it can be observed that better performance can be obtained with DCDP. When $\Delta\psi$ is set to the optimal value of 0.25, the mAP and top-1 are improved by 3.4% and 1.3%, respectively.

2) k-means: The cluster number k is the most crucial parameter in k-means, which is adopted as the dynamic parameter in DCDP. Given the initial value k and the increment size Δk of the cluster number, the cluster number k_1 and k_2 of $clustering_1$ and $clustering_2$ will vary in the range of $[k - \Delta k,\ k + \Delta k]$. Then the k_1^i and k_2^i at the i-th epoch can be obtained from Eq. (4). (Replace ε with k in Eq. (4), and round up the result to the nearest integer.) As shown in Fig. 6(c), the best cluster number k on Market-1501 is 800. After tuning Δk in Fig. 6(d), it can be observed that better performance can be obtained with DCDP, which is consistent with the conclusion on DBSCAN and InfoMap. When Δk is set to the optimal value of 400, DCDP brings $+2.6\%$/$+1.8\%$ mAP/top-1 improvements.

F. Generalization Analysis

We verify the generalization of the proposed DCDP and CSM strategies on other methods that utilize peer-teaching strategy: MMT [22] and Co-teaching [17]. DCDP can be directly employed for MMT and Co-teaching, while CSM

TABLE V
THE GENERALIZATION ANALYSIS OF THE PROPOSED DCDP AND CSM STRATEGIES ON OTHER RE-ID METHODS.

Method	**Market-1501**			
	mAP	top-1	top-5	top-10
MMT [22]	70.8	86.4	95.1	97.2
MMT + DCDP	72.7	87.9	95.2	97.3
MMT + CSM	72.3	87.5	95.4	97.1
MMT + DCDP + CSM	73.5	88.2	95.4	97.4
Co-teaching [17]	82.1	92.2	96.6	97.7
Co-teaching + DCDP	84.1	93.3	97.1	97.9
Co-teaching + CSM	83.3	92.8	96.9	97.7
Co-teaching + DCDP + CSM	85.4	93.8	97.5	98.3

requires additional settings. In order to utilize CSM on MMT, we create two additional memory banks according to Eq. (5). We adopt the same loss function and memory banks as this paper to implement Co-teaching so that CSM can be directly employed. Table V shows the results on Market-1501. It can be observed that using the proposed DCDP or CSM on both MMT and Co-teaching can obtain better performance. In addition, the combination of DCDP and CSM can further improve performance. The above experimental results show that DCDP and CSM have good generalization.

G. Computational Complexity Analysis

We compare the training computation costs of DCCT with other methods that utilize peer-teaching strategy: MMT [22] and Co-teaching [17]. We refer to PPLR [13] to analyze the number of training parameters, the pseudo label generation stage time, and the training stage time for each method, and the results are in Table VI. MMT appends a linear layer at the end of ResNet-50 as the classifier, thus having more parameters. Co-teaching is reproduced by us, using the same backbone and loss function as DCCT, and the number of parameters is the same. Since DCCT and Co-teaching perform clustering twice in the pseudo label generation stage, they have more time consumption. In the training stage, MMT needs to calculate multiple loss functions, so it takes the longest time. Compared with Co-teaching, although the $Mean\ Nets$ used by DCCT increase the calculation of feature extraction, DCCT performs consistent sample mining (CSM) to reduce the number and calculation of training samples. Overall, DCCT takes less time than Co-teaching in the training stage.

Compared with the baseline (defined in Sec. IV-D), DCCT employs DCDP and CSM. DCDP only additionally utilizes Eq. (4) to calculate the clustering parameters without increasing the computational complexity in the pseudo label generation stage. CSM reduces the number of training samples, thereby reducing the computation time in the training stage.

Since these methods utilize only one network in the inference stage without additional computational or memory costs, their inference speeds are the same. It takes about 25.8189 seconds to evaluate on Market-1501. Therefore, compared with the previous peer-teaching strategy methods, DCCT slightly reduces the computation while improving the performance.

TABLE VI
TRAINING COST COMPARISON ON MARKET-1501. THE PSEUDO LABEL GENERATION STAGE TIME INCLUDES THE TIME FOR FEATURE EXTRACTION, CLUSTERING, AND MEMORY INITIALIZATION. SINCE THE NUMBER OF ITERATIONS PER EPOCH IS DIFFERENT FOR EACH METHOD, WE MEASURE "SEC/ITER" FOR A FAIR EVALUATION OF TRAINING STAGE TIME. IN ADDITION, WE REFER TO SEC. III-C TO SET THE SAME IMAGE SIZE AND BATCH SIZE FOR ALL EXPERIMENTS.

Method	Parameters (M)	Pseudo Label Generation Stage Time (sec / epoch)	Training Stage Time (sec / iter)
MMT-DBSCAN [22]	49.8914	43.0273	1.2860
Co-teaching [17]	47.0242	67.3528	0.8732
Baseline	47.0242	68.3845	0.9616
DCCT (Ours)	47.0242	68.4997	0.8471

H. Parameter Analysis

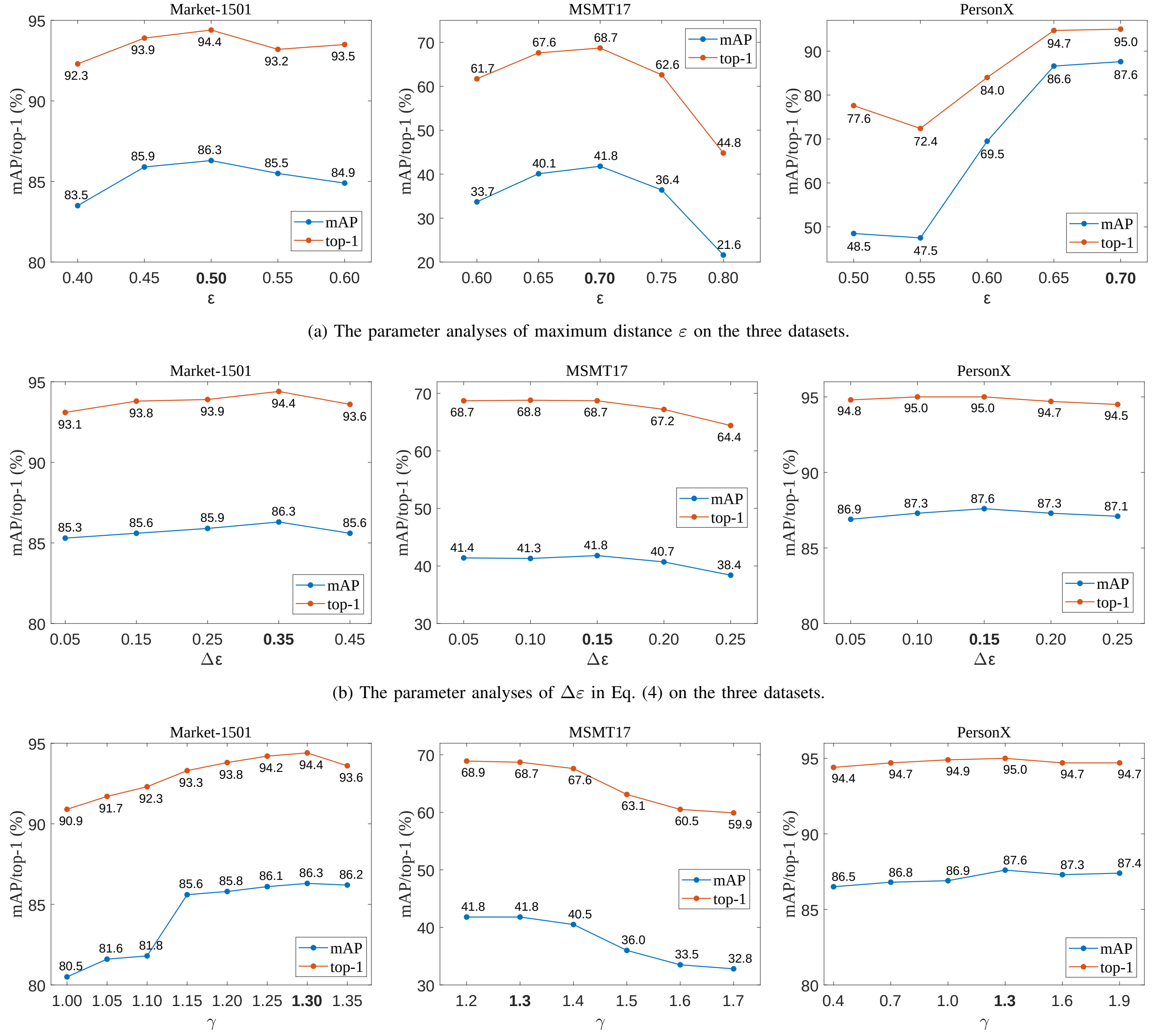
1) Maximum Distance for DBSCAN: Due to distribution differences on various datasets, many state-of-the-art methods use inconsistent clustering hyper-parameters on different datasets [12]–[14], [29]. In DBSCAN [48], hyper-parameter ε represents the maximum distance between two samples. DBSCAN with a smaller ε tends to group persons with the same identity into different clusters. Conversely, DBSCAN with a larger ε tends to group persons with different identities into the same cluster. Both too large and too small ε degrade the clustering quality and hinder the network training.

Fig. 7(a) shows the sensitivity of the performance of DCCT to ε on three datasets. The optimal ε on each dataset is **bold**. It can be observed that the best value of ε on the PersonX dataset is 0.7. When ε is increased to 0.75 on PersonX, the network cannot converge due to too few clusters. The optimal ε on Market-1501 and MSMT17 is 0.5 and 0.7, respectively. However, different methods may have different optimal ε on the same dataset. State-of-the-art unsupervised person Re-ID method ISE [14] sets ε to 0.4 on Market-1501 and 0.7 on MSMT17. And SOTA method PPLR [13] sets ε to 0.6 on Market-1501 and 0.7 on MSMT17. Both the above methods set ε on Market-1501 smaller than that on MSMT17, which is consistent with our experimental results.

2) Hyper-parameters Introduced in Our Method : Our method introduces hyper-parameters $\Delta\varepsilon$ (see Eq. (4)) and γ (see Sec. III-C) related to clustering. As mentioned above, the best clustering hyper-parameters are usually inconsistent on different datasets due to distribution differences. Therefore, we tune the hyper-parameters on the three datasets.

The optimal hyper-parameters on each dataset are **bold** in Fig. 7(b). It can be observed that both too large and too small $\Delta\varepsilon$ lead to performance degradation. Too small $\Delta\varepsilon$ provides little difference between the two clusterings, leading to a finite improvement in the differences and complementarity of the network. While too large $\Delta\varepsilon$ may result in a poor clustering parameter ε, which reduces the clustering quality and hinders the network's training.

As shown in Fig. 7(c), the best value for γ on all three datasets is 1.3. Experiments on Market-1501 show that network performance is degraded due to too small γ. The reason is that CSM starts too late, and many noise samples are used for network training. The results on MSMT17 show that too large γ also hinders network training. The reason is that prematurely mining consistent samples on low-quality



(a) The parameter analyses of maximum distance ε on the three datasets.

(b) The parameter analyses of $\Delta\varepsilon$ in Eq. (4) on the three datasets.

(c) The parameter analyses of γ in Sec. III-C on the three datasets.

Fig. 7. The parameter analyses on the three datasets. The optimal parameters are **bold**.

clustering results in too few samples available for network training. The network performance on PersonX is not sensitive to γ.

V. Conclusion and Limitations

Conclusion. This paper proposes a novel Dual Clustering Co-teaching (DCCT) framework to deal with noisy pseudo labels in unsupervised person Re-ID tasks. Unlike the previous peer-teaching methods utilizing a set of noisy pseudo labels to train the two networks, we propose dual clustering with dynamic parameters (DCDP) to generate two sets of pseudo labels for network training, which can increase the two networks' differences and complementarity, so that our method is more robust to the noisy pseudo labels. Meanwhile, the clustering parameters change dynamically in each epoch, making the network adaptive and robust to different clustering parameters, thus improving the generalization ability of the network. Furthermore, we also propose consistent sample mining (CSM) to find the samples with unchanged pseudo labels during training and remove potential noisy samples. Extensive experimental results on various person Re-ID datasets demonstrate that our method outperforms the prior state-of-the-art unsupervised methods.

Limitations. We propose DCDP and CSM to reduce the impact of pseudo label noise. In recent studies, many methods reduce pseudo label noise by leveraging additional camera labels [12], [13], [29], [34]. However, the performance improvement obtained by directly utilizing camera labels in DCCT is limited. In future work, we will explore more effective ways to leverage camera labels to facilitate peer-teaching methods.

In addition, the DCDP and CSM are designed to promote peer-teaching methods and are only fully verified on peer-teaching methods. The idea of dynamic clustering parameters and consistent sample mining may also be effective for the training of a single network. In future work, we will further generalize the above ideas to the training of a single network.

References

[1] L. Zheng, Y. Yang, and A. G. Hauptmann, "Person re-identification: Past, present and future," *arXiv preprint arXiv:1610.02984*, 2016.

[2] H. Tian, X. Zhang, L. Lan, and Z. Luo, "Person re-identification via adaptive verification loss," *Neurocomputing*, vol. 359, pp. 93–101, 2019.

[3] X. Ning, K. Gong, W. Li, L. Zhang, X. Bai, and S. Tian, "Feature refinement and filter network for person re-identification," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 31, no. 9, pp. 3391–3402, 2020.

[4] H. Park and B. Ham, "Relation network for person re-identification," in *AAAI*, vol. 34, no. 07, 2020, pp. 11 839–11 847.

[5] J. Zhou, B. Su, and Y. Wu, "Online joint multi-metric adaptation from frequent sharing-subset mining for person re-identification," in *CVPR*, 2020, pp. 2909–2918.

[6] X. Shu, X. Wang, X. Zang, S. Zhang, Y. Chen, G. Li, and Q. Tian, "Large-scale spatio-temporal person re-identification: Algorithms and benchmark," *IEEE Transactions on Circuits and Systems for Video Technology*, 2021.

[7] L. Zheng, L. Shen, L. Tian, S. Wang, J. Wang, and Q. Tian, "Scalable person re-identification: A benchmark," in *ICCV*, 2015, pp. 1116–1124.

[8] L. Wei, S. Zhang, W. Gao, and Q. Tian, "Person transfer gan to bridge domain gap for person re-identification," in *CVPR*, 2018, pp. 79–88.

[9] X. Sun and L. Zheng, "Dissecting person re-identification from the viewpoint of viewpoint," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 608–617.

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

[11] K. Zeng, M. Ning, Y. Wang, and Y. Guo, "Hierarchical clustering with hard-batch triplet loss for person re-identification," in *CVPR*, 2020.

[12] H. Chen, B. Lagadec, and F. Bremond, "Ice: Inter-instance contrastive encoding for unsupervised person re-identification," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 14 960–14 969.

[13] Y. Cho, W. J. Kim, S. Hong, and S.-E. Yoon, "Part-based pseudo label refinement for unsupervised person re-identification," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 7308–7318.

[14] X. Zhang, D. Li, Z. Wang, J. Wang, E. Ding, J. Q. Shi, Z. Zhang, and J. Wang, "Implicit sample extension for unsupervised person re-identification," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 7369–7378.

[15] Y. Ge, F. Zhu, D. Chen, R. Zhao *et al.*, "Self-paced contrastive learning with hybrid memory for domain adaptive object re-id," *Advances in Neural Information Processing Systems*, vol. 33, pp. 11 309–11 321, 2020.

[16] E. Malach and S. Shalev-Shwartz, "Decoupling" when to update" from" how to update"," *Advances in neural information processing systems*, vol. 30, 2017.

[17] B. Han, Q. Yao, X. Yu, G. Niu, M. Xu, W. Hu, I. Tsang, and M. Sugiyama, "Co-teaching: Robust training of deep neural networks with extremely noisy labels," in *NIPS*, 2018, pp. 8527–8537.

[18] X. Wang, S. Wang, J. Wang, H. Shi, and T. Mei, "Co-mining: Deep face recognition with noisy labels," in *ICCV*, 2019, pp. 9358–9367.

[19] A. Tarvainen and H. Valpola, "Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results," in *NIPS*, 2017, pp. 1195–1204.

[20] J. Li, R. Socher, and S. C. Hoi, "Dividemix: Learning with noisy labels as semi-supervised learning," in *ICLR*, 2020. [Online]. Available: https://openreview.net/forum?id=HJgExaVtwr

[21] F. Yang, K. Li, Z. Zhong, Z. Luo, X. Sun, H. Cheng, X. Guo, F. Huang, R. Ji, and S. Li, "Asymmetric co-teaching for unsupervised cross-domain person re-identification," *AAAI*, vol. 34, pp. 12 597–12 604, 04 2020.

[22] Y. Ge, D. Chen, and H. Li, "Mutual mean-teaching: Pseudo label refinery for unsupervised domain adaptation on person re-identification," in *ICLR*, 2020. [Online]. Available: https://openreview.net/forum?id=rJlnOhVYPS

[23] Y. Dong, H. Liu, and C. Xu, "Asymmetric mutual mean-teaching for unsupervised domain adaptive person re-identification," *IEEE Access*, vol. 9, pp. 69 971–69 984, 2021.

[24] F. Zhao, S. Liao, G.-S. Xie, J. Zhao, K. Zhang, and L. Shao, "Unsupervised domain adaptation with noise resistible mutual-training for person re-identification," in *ECCV*. Springer, 2020, pp. 526–544.

[25] H. Li, N. Dong, Z. Yu, D. Tao, and G. Qi, "Triple adversarial learning and multi-view imaginative reasoning for unsupervised domain adaptation person re-identification," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 32, no. 5, pp. 2814–2830, 2022.

[26] Z. Zhang, Y. Wang, S. Liu, B. Xiao, and T. S. Durrani, "Cross-domain person re-identification using heterogeneous convolutional network," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 32, no. 3, pp. 1160–1171, 2022.

[27] Y. Zhai, S. Lu, Q. Ye, X. Shan, J. Chen, R. Ji, and Y. Tian, "Adcluster: Augmented discriminative clustering for domain adaptive person re-identification," in *CVPR*, 2020, pp. 9021–9030.

[28] S. Li, M. Yuan, J. Chen, and Z. Hu, "Adadc: Adaptive deep clustering for unsupervised domain adaptation in person re-identification," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 32, no. 6, pp. 3825–3838, 2022.

[29] S. Xuan and S. Zhang, "Intra-inter camera similarity for unsupervised person re-identification," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2021, pp. 11 926–11 935.

[30] Z. Wu, Y. Xiong, S. X. Yu, and D. Lin, "Unsupervised feature learning via non-parametric instance discrimination," in *CVPR*, June 2018.

[31] K. He, H. Fan, Y. Wu, S. Xie, and R. Girshick, "Momentum contrast for unsupervised visual representation learning," in *CVPR*, 2020, pp. 9729–9738.

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

[33] Z. Dai, G. Wang, W. Yuan, S. Zhu, and P. Tan, "Cluster contrast for unsupervised person re-identification," in *ACCV*, December 2022, pp. 1142–1160.

[34] M. Wang, B. Lai, J. Huang, X. Gong, and X.-S. Hua, "Camera-aware proxies for unsupervised person re-identification," in *AAAI*, vol. 35, no. 4, 2021, pp. 2764–2772.

[35] Y. Liu, H. Ge, L. Sun, and Y. Hou, "Complementary attention-driven contrastive learning with hard-sample exploring for unsupervised domain adaptive person re-id," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 33, no. 1, pp. 326–341, 2022.

[36] G. Zhang, H. Zhang, W. Lin, A. K. Chandran, and X. Jing, "Camera contrast learning for unsupervised person re-identification," *IEEE Transactions on Circuits and Systems for Video Technology*, 2023.

[37] A. Vahdat, "Toward robustness against label noise in training deep discriminative neural networks," in *NIPS*, 2017, pp. 5596–5605.

[38] G. Patrini, A. Rozza, A. Krishna Menon, R. Nock, and L. Qu, "Making deep neural networks robust to label noise: A loss correction approach," in *CVPR*, 2017, pp. 1944–1952.

[39] A. Ghosh, H. Kumar, and P. Sastry, "Robust loss functions under label noise for deep neural networks," in *AAAI*, 2017, pp. 1919–1925.

[40] Z. Zhang and M. Sabuncu, "Generalized cross entropy loss for training deep neural networks with noisy labels," in *NIPS*, 2018, pp. 8778–8788.

[41] K.-H. Lee, X. He, L. Zhang, and L. Yang, "Cleannet: Transfer learning for scalable image classifier training with label noise," in *CVPR*, 2018, pp. 5447–5456.

[42] J. Han, P. Luo, and X. Wang, "Deep self-learning from noisy labels," in *ICCV*, 2019, pp. 5138–5147.

[43] B. Zhang, Y. Wang, W. Hou, H. Wu, J. Wang, M. Okumura, and T. Shinozaki, "Flexmatch: Boosting semi-supervised learning with curriculum pseudo labeling," *Advances in Neural Information Processing Systems*, vol. 34, pp. 18 408–18 419, 2021.

[44] Y. Xu, J. Ding, L. Zhang, and S. Zhou, "Dp-ssl: Towards robust semi-supervised learning with a few labeled samples," *Advances in Neural Information Processing Systems*, vol. 34, pp. 15 895–15 907, 2021.

[45] X. Yu, B. Han, J. Yao, G. Niu, I. Tsang, and M. Sugiyama, "How does disagreement help generalization against label corruption?" in *ICML*. PMLR, 2019, pp. 7164–7173.

[46] S. Laine and T. Aila, "Temporal ensembling for semi-supervised learning," in *ICLR*, 2017. [Online]. Available: https://openreview.net/forum?id=BJ6oOfqge

[47] W. Wang, F. Zhao, S. Liao, and L. Shao, "Attentive waveblock: Complementarity-enhanced mutual networks for unsupervised domain adaptation in person re-identification and beyond," *IEEE Transactions on Image Processing*, vol. 31, pp. 1532–1544, 2022.

[48] M. Ester, H.-P. Kriegel, J. Sander, X. Xu *et al.*, "A density-based algorithm for discovering clusters in large spatial databases with noise." in *kdd*, vol. 96, no. 34, 1996, pp. 226–231.

[49] S. Lloyd, "Least squares quantization in pcm," *IEEE transactions on information theory*, vol. 28, no. 2, pp. 129–137, 1982.

[50] M. Rosvall and C. T. Bergstrom, "Maps of random walks on complex networks reveal community structure," *Proceedings of the national academy of sciences*, vol. 105, no. 4, pp. 1118–1123, 2008.

[51] M. Kumar, B. Packer, and D. Koller, "Self-paced learning for latent variable models," *Advances in neural information processing systems*, vol. 23, 2010.

[52] D. L. Davies and D. W. Bouldin, "A cluster separation measure," *IEEE transactions on pattern analysis and machine intelligence*, no. 2, pp. 224–227, 1979.

[53] Z. Wu, Y. Xiong, S. X. Yu, and D. Lin, "Unsupervised feature learning via non-parametric instance discrimination," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 3733–3742.

[54] L. Zheng, L. Shen, L. Tian, S. Wang, J. Wang, and Q. Tian, "Scalable person re-identification: A benchmark," in *ICCV*, 2015, pp. 1116–1124.

[55] D. Gray, S. Brennan, and H. Tao, "Evaluating appearance models for recognition, reacquisition, and tracking," in *Proc. IEEE international workshop on performance evaluation for tracking and surveillance (PETS)*, vol. 3, no. 5. Citeseer, 2007, pp. 1–7.

[56] L. Van Der Maaten, "Accelerating t-sne using tree-based algorithms," *The Journal of Machine Learning Research*, vol. 15, no. 1, pp. 3221–3245, 2014.

[57] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *CVPR*, 2016, pp. 770–778.

[58] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *NIPS*, 2012.

[59] Z. Zhong, L. Zheng, G. Kang, S. Li, and Y. Yang, "Random erasing data augmentation," *arXiv preprint arXiv:1708.04896*, 2017.

[60] Z. Zhong, L. Zheng, D. Cao, and S. Li, "Re-ranking person re-identification with k-reciprocal encoding," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 1318–1327.

Zeqi Chen received his B.E. degree in automation at Northwestern Polytechnical University, China, in 2016. He is currently a Ph.D. candidate in the Institute of Artificial Intelligence and Robotics, Xi'an Jiaotong University, China. His research interests include image processing, computer vision, pattern recognition, and machine learning.

Zhichao Cui received his B.E. degree in information engineering and Ph.D. degree in control science and engineering at the Xi'an Jiaotong University, Xi'an, China in 2013 and 2021. He is currently a lecturer in school of electronics and control engineering, at the Chang'an University, Xi'an, China. His research interests include machine learning, computer vision and computer graphics.

Chi Zhang received his B.S. and Ph.D. degrees in Control Science and Engineering from Xi'an Jiaotong University, Xi'an, China, in 2011 and 2021, respectively. During 2016 to 2017, he visited the Department of Computer Science at Northwestern University, USA, under the supervision of Prof. Ying Wu. He is currently an Assistant Professor with the Institute of Artificial Intelligence and Robotics, Xi'an Jiaotong University, Xi'an, China. His research interests include intelligence testing for autonomous driving systems, computer vision and machine learning.

Jiahuan Zhou received his B.E. (2013) from Tsinghua University, the Ph.D. degree (2018) in the Department of Electrical Engineering & Computer Science, Northwestern University. During summer 2018, he was a research intern with Microsoft Research, Redmond, Washington. From 2019 to 2022, he was a Postdoctoral Fellow and Research Assistant Professor at Northwestern University. Currently, he is a Tenure-Track Assistant Professor with the Wangxuan Institute of Computer Technology, Peking University. His current research interests include computer vision, deep learning, and machine learning. He has authored about 20 papers in international journals and conferences including IEEE T-PAMI, IEEE TIP, CVPR, ICCV, ECCV, and so on. He serves as an area chair for CVPR'2023, ICME'2020-2021, ICPR'2022, and a regular reviewer member for a number of journals and conferences, e.g., T-PAMI, IJCV, TIP, TCSVT, CVPR, ICCV, ECCV, NeurIPS, AAAI, and so on.

Yuehu Liu received the B.E. and M.E. degrees from Xi'an Jiaotong University, Xi'an, China, in 1984 and 1989, respectively, and the Ph.D. degree in electrical engineering from Keio University, Tokyo, Japan, in 2000.

He is currently a Professor with Xi'an Jiaotong University. His current research interests include computer vision, computer graphics, and simulation testing for autonomous vehicle.

Dr. Liu is a member of the IEICE.