Dual-Refinement: Joint Label and Feature Refinement for Unsupervised Domain Adaptive Person Re-Identification

Anonymous Author(s)
Submission Id: 679

ABSTRACT

Unsupervised domain adaptive (UDA) person re-identification (re-ID) is a challenging task due to the missing of labels for the target domain data. To handle this problem, some recent works adopt clustering algorithms to off-line generate pseudo labels, which can then be used as the supervision signal for on-line feature learning in the target domain. However, the off-line generated labels often contain lots of noise that significantly hinders the discriminability of the on-line learned features, and thus limits the final UDA re-ID performance. To this end, we propose a novel approach, called Dual-Refinement, that jointly refines pseudo labels at the off-line clustering phase and features at the on-line training phase, to alternatively boost the label purity and feature discriminability in the target domain for more reliable re-ID. Specifically, at the off-line phase, a new hierarchical clustering scheme is proposed, which selects representative prototypes for every coarse cluster. Thus, labels can be effectively refined by using the inherent hierarchical information of person images. Besides, at the on-line phase, we propose an instant memory spread-out (IM-spread-out) regularization, that takes advantage of the proposed instant memory bank to store sample features of the entire dataset and enable spread-out feature learning over the entire training data on-the-fly. Our Dual-Refinement method reduces the influence of noisy labels and refines the learned features within the alternative training process. Experiments demonstrate that our method outperforms the state-of-the-art methods by a large margin. (*Our code is anonymously shared at https://github.com/MM20Anonymous/Dual-Refinement*.)

CCS CONCEPTS

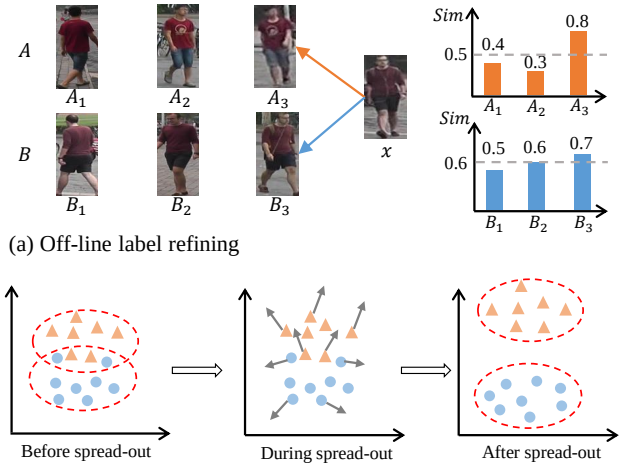
• **Computing methodologies** → **Transfer learning**; **Object identification**.

KEYWORDS

Person Re-ID, Unsupervised Domain Adaption, Pseudo Label Noise

1 INTRODUCTION

Person re-identification (re-ID) aims at identifying the same person's images as the query in a gallery database across disjoint cameras. Due to the great value in practical applications concerning public security and surveillance, person re-ID has attracted booming attention in the research community [54, 62]. Most of the existing works for person re-ID focus on fully supervised scenarios [3, 32, 41, 46], where they have obtained superior performance on some benchmarks as the model's training and testing are conducted in the same domain. However, their performances often drop dramatically when models trained on the labeled source domain are directly applied to the unlabeled target domain owing to the domain gap. To handle the domain gap issue in the cross-domain



(a) Off-line label refining

(b) On-line feature refining

Figure 1: (a) A_1, A_2, A_3 and B_1, B_2, B_3 are representative prototypes in cluster A and B respectively. Similarity (Sim) histograms denote the similarity between the sample x and each prototype. (b) Two-dimensional visualization of feature space. Points and triangles belong to two different clusters. Dashed ovals represent class decision boundaries. Arrows represent spreading out the features.

re-ID scenario, recently, plenty of unsupervised domain adaptation (UDA) approaches [10, 57, 68] have been proposed. Many UDA re-ID methods adopt clustering algorithms [10, 12, 39, 53, 57] to generate pseudo labels for the unlabeled target domain data, which can then be used as the supervision signal for model training.

Concretely, in clustering-based UDA methods [10, 12, 39, 53, 57], the model is often pre-trained with the labeled source domain data via a fully supervised manner. Then the model is fine-tuned using unlabeled target domain data in an alternative training manner including the off-line pseudo label generation phase and on-line feature learning phase. Specifically, at the off-line phase, pseudo labels are generated by performing clustering on the features of the target domain samples, which are extracted with the trained model. At the on-line phase, the model is trained under the supervision of the pseudo labels generated from the off-line phase. The off-line label generation and on-line model training are conducted alternatively and iteratively over the whole learning process. However, the quality of the off-line pseudo labels directly affects the on-line feature learning performance [12]. Meanwhile, the discriminability of the on-line learned features, in turn, affects the off-line pseudo label generation at the next epoch. Thus, we need to alleviate the influence of label noises at both off-line and on-line phases to improve UDA re-ID performance. Therefore, in this paper, we propose a novel approach called Dual-Refinement to jointly refine the off-line

pseudo label generation and also the on-line feature learning under the noisy pseudo labels, during an alternative training process.

To refine the off-line pseudo labels, we design a novel hierarchical clustering scheme at the off-line phase. Existing clustering-based UDA methods [10, 39, 57] generally perform clustering based on the local similarities among the samples [8] to assign pseudo labels, *i.e.,* the sample's nearest neighborhoods are more likely to be grouped into the same cluster. However, such sample neighborhoods tend to ignore the global and inherent characteristics of each cluster, due to the high intra-cluster variance caused by different poses or viewpoints of the person. As shown in Figure 1 (a), when only considering the local similarities, the sample x is wrongly grouped into A, because x is very similar to A_3, *i.e.,* x and A_3 share the same pose and viewpoint. As a result, the off-line pseudo label assignment based on such coarse clustering often brings about label noises. To refine the noisy pseudo labels, we propose to consider the global characteristics of every coarse cluster by selecting several representative prototypes within every coarse cluster. Specifically, we propose a hierarchical clustering method, in which we perform fine clustering after the coarse clustering. Moreover, the fine sub-cluster centers serve as the representative prototypes for every coarse cluster. Compared with local similarities, the average similarity between the sample and these representative prototypes is more powerful in capturing each person's global and inherent information. Thus it provides a more robust criterion for the off-line pseudo label assignment, which mitigates the label noise issue. As shown in Figure 1 (a), when considering the global characteristics within every cluster, the sample x should be grouped into the cluster B, because the average similarity between x and cluster B's prototypes (B_1, B_2, B_3) is 0.6, which is larger than 0.5 achieved by using A's prototypes. Thus by considering the global characteristics within each coarse cluster, we can assign more reliable pseudo labels.

At the on-line phase, to refine the feature learning and alleviate the effects of noisy pseudo labels, we propose an instant memory spread-out (IM-spread-out) regularization scheme in our Dual-Refinement method. Since the pseudo labels can contain noise, directly using such noisy pseudo labels to supervise the metric learning (classification and triplet loss) can limit the on-line feature learning performance. As shown in Figure 1 (b), the label noises originate from the off-line clustering that aims at discovering class decision boundaries [2, 21], and the noisy samples tend to be located near the decision boundary. Thus, these noisy samples confuse the on-line feature learning and limit the discriminability of the learned features. To pull noisy samples away from the decision boundaries and boost the discriminability of features, we propose an IM-spread-out regularization scheme during on-line feature learning, as shown in Figure 1 (b). The spread-out property will not break the inherent characteristics of those reliable samples, because the reliable samples are inherently compact in a cluster, which thus is still robust during our IM-spread-out regularization.

To effectively capture the global distribution, we enforce the spread-out property on the whole training dataset. Specifically, we consider every sample in the target domain as an instance, and our IM-spread-out regularization scheme satisfies the positive-centered and negative-separated properties. Moreover, the sample's $k-$nearest neighborhoods can be seen as the positives, and all the remaining of the entire dataset can be seen as the negatives. However, it is hard to enforce the spread-out constraints on the whole training data in the mini-batch training manner, because the mini-batch can only capture the local data distribution. One possible solution is to use a memory bank [49, 50] to store the features of all the samples. However, existing memory bank methods can only memorize outdated features [18, 55], which may result in sub-optimal on-line feature learning. Therefore, in this paper, we propose a novel instant memory bank to memorize the on-the-fly features of all the training data. The instant memory bank memorizes the sample features upon features feed into the bank and meanwhile the instant memory bank is updated together with the network instantly at every training iteration. It effectively captures the global distribution and can be used for the IM-spread-out regularization and boost the on-line features' discriminability. This on-line feature refinement and the off-line pseudo label refinement are conducted in an alternative manner. Finally, the trained model can generalize well in the target domain.

The major contributions can be summarized as follows:

- We propose a novel approach called Dual-Refinement to jointly boost the quality of off-line pseudo labels and the discriminability of on-line features, with only a single model.
- We design a hierarchical clustering scheme to select representative prototypes for every coarse cluster, which captures the more global and inherent characteristics of each person, thus can refine the pseudo labels at the off-line phase.
- We propose an IM-spread-out regularization scheme to alleviate the effects of pseudo label noises at the on-line phase. Thus it improves the feature discriminability in the target domain. Moreover, a novel instant memory bank is proposed to store on-the-fly features and thus can enforce the spread-out property on the whole target training dataset.
- Extensive experiments have shown that our method outperforms state-of-the-art UDA approaches by a large margin.

2 RELATED WORK

Unsupervised Domain Adaptation. The existing general UDA methods fall into two main categories: closed set UDA [11, 23, 30, 31, 44] and open set UDA [34, 38]. In closed set UDA, both the target and source domain completely share the same classes. Most of the closed set UDA works [11, 30, 44] try to learn the domain invariant features to generalize the class decision boundary well on the target domain. Long *et al.* propose DAN [30] and RTN [31] to minimize Maximum Mean Discrepancy (MMD) across different domains. In open set UDA [34, 38], the target and source domain dataset only share a part of classes. Saito *et al.* [38] use adversarial training to align target samples with known source samples or recognize them as an unknown class. All the general UDA methods mentioned above assume that the source and target domain share the whole or partial classes under the image classification scenario, which are difficult to be directly applied to UDA person re-ID tasks.

Unsupervised Domain Adaptation for Person re-ID. The unsupervised domain adaptation methods for person re-ID can be mainly categorized into two aspects, one is the GAN-based [6, 48, 66], and the other is the clustering-based [10, 12, 39, 53, 57]. SPGAN [6] and PTGAN [48] use CycleGAN [70] to translate the style of the source domain to the target domain and conduct the

feature learning with the source domain labels. HHL [66] uses StarGAN [4] to learn the features with camera invariance and domain connectedness. UDAP [39] first proposes the clustering-based UDA framework for re-ID. SSG [10] and PCB-PAST [57] bring the information of both the global body and local parts into the clustering-based framework. Some clustering-based methods [12, 53, 57] are devoted to solving the pseudo label noise problem. MMT [12] proposes an on-line peer-teaching framework to refine the noisy pseudo labels, which uses the on-line reliable soft labels generated from the temporally average model of one network to supervise the training of another network. However, both MMT [12] and ACT [53] introduce other networks that will bring about extra noises, and they are not memory efficient. Besides, PCB-PAST [57] proposes the ranking-based triplet loss to alleviate the influence of label noises in on-line metric learning but it only focuses on local data distribution based on mini-batches. There are also some other works like ECN [67] and ECN+GPP [68], which use the traditional memory bank [49, 50, 52] to consider every sample as an instance to learn the invariant feature. However, these methods need additional images generated by HHL [66], and the features stored in the traditional memory bank are outdated [18, 55].

Different from all the UDA re-ID methods mentioned above, we propose a novel Dual-Refinement method that is able to jointly refine the off-line pseudo labels and the on-line features. A novel instant memory bank is also proposed to store on-the-fly features and enforce the spread-out property on the whole training data, which captures the global characteristics of the whole target domain.

Learning with Noisy Labels. Existing works on learning with noisy labels can be categorized into four main groups. The methods in the first category focus on learning a transition matrix [14, 33, 36, 51]. However, it is hard to estimate the noise label transition for UDA re-ID because the classes in the target domain are known. The second category [13, 42, 59] is to design the loss functions robust to noise labels, but they bring about extra constraints like the mean absolute loss in GCE [59]. The third category [16, 22, 25] is to utilize additional networks to refine the noisy labels. Co-teaching [16] uses two networks in a co-trained manner. These methods need other networks and complicated sample selection strategies. The last category [17, 37, 43] learns on noisy labels in a self-training manner. Han *et al.* [17] design a complicated class-prototype selection strategy to train with samples robust to noises in the ground-truth, while we utilize the inherent hierarchical structure to cluster and assign labels. Different from the aforementioned methods in the image classification scenario, we propose a hierarchical clustering scheme to capture the diversity within every coarse cluster, thus can handle the label noise issue in UDA re-ID.

Feature Embedding with Spread-out Property. Feature embedding learning with the spread-out property has improved the performance in deep local feature learning [58], unsupervised embedding learning [55], and face recognition [7, 28, 60]. Zhang *et al.* [58] propose a Global Orthogonal Regularization to fully utilize the feature space by making the negative pairs close to orthogonal. Ye *et al.* [55] use a siamese network to learn data augmentation invariant and instance spread-out features under the instance-wise supervision. These works [55, 58] guarantee the spread-out property of features within a mini-batch. Other works in face recognition [7, 28, 60] use a term regularized on the classifier weights to make class centers spread-out in the holistic feature space. However, they are under the supervision of the ground truth. Unlike the above works, our IM-spread-out regularization is used to alleviate the influence of noisy labels on on-line metric learning for UDA re-ID. Besides, we use an instant memory bank to enforce the spread-out property on the entire training data instead of the mini-batch.

3 PROPOSED METHOD

Problem Definition. In unsupervised domain adaptive person Re-ID, we are given a labeled source domain dataset and an unlabeled target domain dataset. In the source domain, the dataset $D_s = \{(x_i^s, y_i^s)|_{i=1}^{N_s}\}$ contains N_s person images and each image x_i^s corresponds to an identity label y_i^s. In the target domain, dataset $D_t = \{x_i^t|_{i=1}^{N_t}\}$ contains N_t unlabeled person images. Our goal is to use both labeled source data and unlabeled target data to learn discriminative image representations in the target domain.

3.1 Overview of Framework

As shown in Figure 2, the framework of our method contains two stages including the off-line pseudo label refinement stage and the on-line feature learning stage. The network (CNN) is initialized by pre-training on source domain data, following a similar method in [32]. CNN is a deep feature encoder $F(\cdot|\theta)$ parameterized with θ, and can encode the person image into a d-dimensional feature.

At the off-line stage, we propose a hierarchical clustering scheme for the target domain features $\{f_i^t|_{i=1}^{N_t}\}$ extracted by the network (CNN) trained in the last epoch, where $f_i^t = F(x_i^t|\theta)$. By clustering, we assign samples in the same cluster with the same pseudo label, and then each target domain sample gets two kinds of pseudo labels, including noisy label $\widetilde{y}_i^t \in \{1, 2, ..., L^t\}$ and refined label $\widehat{y}_i^t \in \{1, 2, ..., L^t\}$, where L^t is the number of the unique labels. The off-line pseudo labels are used for the on-line feature learning.

At the on-line stage, we use samples with noisy labels $\widetilde{D}_t = \{(x_i^t, \widetilde{y}_i^t)|_{i=1}^{N_t}\}$ to train with the classification loss $\widetilde{L}_{cls}$ and triplet loss $\widetilde{L}_{tri}$, and use samples with refined labels $\widehat{D}_t = \{(x_i^t, \widehat{y}_i^t)|_{i=1}^{N_t}\}$ to train with the loss $\widehat{L}_{cls}$ and $\widehat{L}_{tri}$. FC is a L^t dimensional fully connected layer followed by softmax function, which is denoted as the identity classifier ϕ. To alleviate the influence of pseudo label noises in the on-line supervised metric learning, we propose a label-free IM-spread-out regularization scheme L_{spread} to train together with the classification loss and triplet loss [20]. Specifically, we propose a novel instant memory bank parameterized by $V = \{v_i|_{i=1}^{N_t}\}$ to store all the samples' features on-the-fly. Thus, the instant memory bank can enforce the spread-out property on the whole target training dataset instead of the mini-batch, which can capture the global characteristics of the target domain distribution. We first conduct the off-line stage at the beginning of every epoch and conduct the on-line stage during every epoch, both of which are conducted in an alternative and iterative manner. For simplicity, we omit the superscript t for the target domain data in the following sections.

Below, we first introduce the general clustering-based UDA procedure. We then introduce our Dual-Refinement method that can optimize both stages in the cluster-based UDA procedure, *i.e.,* jointly refine off-line pseudo labels and on-line features, and thus improve the overall UDA performance.

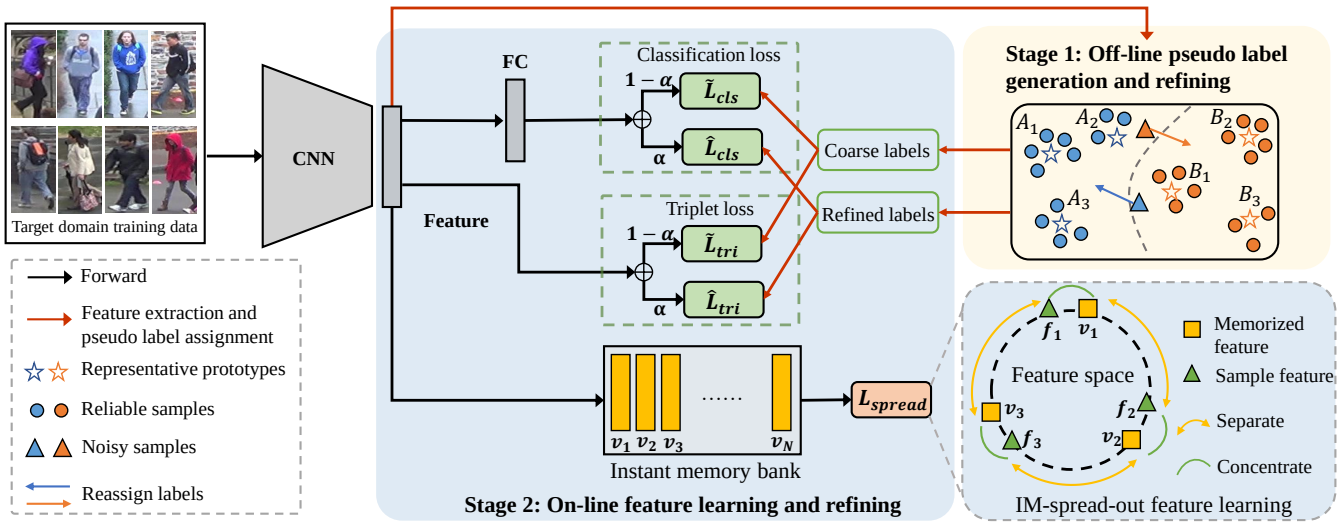


Figure 2: The framework of our method. After initializing CNN by pre-training it on the labelled source domain, we train our network in the target domain in an alternative manner including two stages. At the beginning of every training epoch, we conduct the off-line stage, where we use the trained model to extract all the sample features and then perform the hierarchical clustering on these features to assign pseudo labels. Then, we conduct the on-line stage, where we use pseudo labels generated from the off-line stage to fine-tune the model with the classification loss and triplet loss, together with a label-free IM-spread-out regularization. These two stages are performed alternatively and iteratively in the target domain. The instant memory bank is used to store on-the-fly sample features. The spread-out feature learning aims to separate different sample features (*e.g.*, f_1, f_2, f_3) and concentrate the sample feature with its corresponding memory (*e.g.*, f_1, v_1) in the feature space.

3.2 General Clustering-based UDA Procedure

Existing clustering-based UDA methods [12, 39, 57] for Re-ID usually pre-train the backbone network with classification loss and triplet loss on labeled source dataset, and then use the pre-trained model to initialize the model for training the target dataset $D = \{x_i|_{i=1}^N\}$. We follow a similar method in [32] to pre-train the backbone model with the labeled source domain, and then perform the training procedure on the target domain, which contains two stages: (1) Off-line assigning pseudo labels based on clustering at the beginning of every training epoch. (2) Utilizing target domain data with pseudo labels to train the network with metric learning loss on-line during every training epoch. These two stages are conducted alternatively and iteratively in the training process.

Off-line assigning pseudo labels. Following the existing clustering based UDA methods [10, 39, 57], we first extract features $\{f_1, f_2, ..., f_N\}$ of all N images in the target domain using the CNN trained from last epoch, and then calculate their pair-wise similarity $d_S(i, j)$ between samples i and j by:

$$d_S(i,j) = \begin{cases} e^{-\|f_i - f_j\|_2}, & if\ j \in R^*(i,k) \\ 0, & otherwise\ . \end{cases} \tag{1}$$

$R^*(i, k)$ is the k-reciprocal nearest neighbor set of sample x_i introduced in [64]. We then use the pair-wise similarity to calculate the Jaccard distance $d_J(i, j)$ by:

$$d_J(i,j) = 1 - \frac{\sum_{k=1}^N min(d_S(i,k), d_S(j,k))}{\sum_{k=1}^N max(d_S(i,k), d_S(j,k))}\ , \tag{2}$$

where N is the number of target training samples. We use such distance metric $d_J(i, j)$ to perform DBSCAN clustering [8] on target domain and obtain L clusters. We consider each cluster as a unique class and assign the same pseudo label for the samples belonging to the same cluster. Thus, the target domain images with their pseudo labels are represented as $\widetilde{D} = \{(x_i, \widetilde{y}_i)|_{i=1}^N\}$, where $\widetilde{y}_i \in \{1, 2, ..., L\}$.

On-line model training with metric learning losses. In this step, we use the target dataset $\widetilde{D} = \{(x_i, \widetilde{y}_i)|_{i=1}^N\}$ labeled by pseudo labels $\{\widetilde{y}_i\}$ to train the network with classification loss L_{cls} and triplet loss [20] L_{tri}, which are foumulated as follows:

$$\widetilde{L}_{cls} = \frac{1}{N}\sum_{i=1}^{N} L_{ce}(\phi(f_i), \widetilde{y}_i)\ , \tag{3}$$

$$\widetilde{L}_{tri} = \frac{1}{N}\sum_{i=1}^{N} max(0, m + \|f_i - f_i^+\|_2 - \|f_i - f_i^-\|_2)\ , \tag{4}$$

where ϕ denotes the classification layer FC and L_{ce} denotes the cross-entropy loss. f_i^+ and f_i^- are the features of the hardest positive and negative of the sample x_i through batch hard mining [20] under the supervision of pseudo labels $\{\widetilde{y}_i\}$, and m is the margin. We denote this general UDA re-ID method as the **baseline** in this paper and fine-tune the network with an overall loss $L_{baseline}$, that is obtained by combining the loss $\widetilde{L}_{cls}$ and $\widetilde{L}_{tri}$:

$$L_{baseline} = \widetilde{L}_{cls} + \widetilde{L}_{tri}\ . \tag{5}$$

3.3 Off-line Pseudo Label Refinement

At the off-line stage, we design a hierarchical clustering guided label refinement strategy. The refining process contains two stages of hierarchical clustering from coarse to fine. As shown in Figure 2, the coarse clusters A and B contain three sub-clusters $\{A_1, A_2, A_3\}$

and $\{B_1, B_2, B_3\}$ respectively. If only one-stage coarse clustering is employed, many noisy samples will be assigned with the false labels because the coarse clustering overlooks the high variances within a cluster, as shown in Figure 1 (a). During the second stage: fine clustering, we select sub-cluster centers as the representative prototypes in each coarse cluster and reassign the cluster labels based on the average similarity between the noisy sample and prototypes in each coarse cluster, which can capture the more global characteristics of each person, and thus can can provide more robust pseudo labels.

Following the pseudo label assignment in Section 3.2, we can obtain L coarse clusters and the target sample set of the class l is denoted as $D_l = \{(x_i, \widetilde{y}_i)|\forall \widetilde{y}_i = l\}$. We extract features of each class to constitute the feature set $F_l = \{(f_i, \widetilde{y}_i)|\forall \widetilde{y}_i = l\}$ and then perform k-means clustering [29] on every coarse class feature set F_l, *i.e,* the coarse cluster l will be splited into R sub-clusters. Afters such fine clustering, we pick the center of every sub-cluster as a prototype. Thus, we can obtain R representative prototypes $\{c_{l,1}, c_{l,2}, ..., c_{l,R}\}$ for every coarse cluster l. By considering all the prototypes, we can get more global and inherent characteristics of every coarse cluster.

As shown in Figure 1 (a), the global characteristics within each coarse cluster can help assign more reliable pseudo labels. Thus, we define the refined similarity score of sample x_i belonging to class l as $s_{i,l}$, which is calculated by

$$s_{i,l} = \frac{1}{R}\sum_{r=1}^{R} f_i^T c_{l,r} \ . \tag{6}$$

The feature f_i and the prototype are all L2-normalized. Similarity score $s_{i,l}$ can be viewed as the average similarity between the sample x_i and the representative prototypes of class l. Instead of assigning the coarse label for all samples within a cluster, our refined similarity score $s_{i,l}$ can provide a more reliable similarity between every sample and coarse clusters. With this similarity, we can reassign more reliable labels for target data by

$$\widehat{y}_i = \underset{l}{arg\ max}\ s_{i,l} \ \ , l \in \{1, 2, ..., L\} \ , \tag{7}$$

where $\widehat{y}_i$ is the refined pseudo label of sample x_i. Now every target sample x_i has two kinds of pseudo labels including coarse noisy label $\widetilde{y}_i$ and refined label $\widehat{y}_i$.

3.4 On-line Feature Refinement

Metric learning with pseudo labels. We can use the refined labels to optimize the network with metric learning losses including $\widehat{L}_{cls}$ and $\widehat{L}_{tri}$, where $\widehat{L}_{cls}$ and $\widehat{L}_{tri}$ are obtained by replacing the noisy pseudo labels $\{\widetilde{y}_i\}$ with our refined labels $\{\widehat{y}_i\}$ in Eq. (3) (4). We combine metric losses under the supervision of both noisy pseudo labels and refined pseudo labels by

$$\begin{cases} L_{cls} = (1-\alpha)\widetilde{L}_{cls} + \alpha\widehat{L}_{cls} \\ L_{tri} = (1-\alpha)\widetilde{L}_{tri} + \alpha\widehat{L}_{tri} \end{cases} . \tag{8}$$

We use $\alpha \in [0, 1]$ to control the weights of reliable and noisy pseudo labels on both classification loss and triplet loss. Moreover, when $\alpha > 0$, the parameter α can prevent mistaking hard samples as noisy samples, *i.e.,* refining pseudo labels excessively, because the clustering quality is low at the early training stage.

IM-pread-out regularization with instant memory bank. Although we have designed the off-line pseudo label refining strategy, it is not possible to eliminate all the label noises. To alleviate the effects of noisy labels, on feature learning, we propose label-free regularization, which aims to spread out the features in the whole feature space and pull the samples assigned with false labels out of the same class. Although the spread-out property has shown its effectiveness in recent works [7, 55], to capture the whole characteristics of target domain, in our task, we enforce the spread-out property on the entire target training dataset instead of the mini-batch. Specifically, we propose an instant memory bank, that can store the on-the-fly features and can be updated together with the network instantly at every iteration.

As shown in Figure 2, we propose the instant memory bank V where each entry in $\{v_i|_{i=1}^N\}$ is a d-dimensional vector, *i.e.,* $v_i \in \mathbb{R}^d$. We use v_i to approximate the feature $f_i \in \mathbb{R}^d$ and thus the memory bank V can memorize the approximated features of the entire dataset. For simplicity, we use the L_2 normalized feature via $f_i \leftarrow f_i/\|f_i\|_2$ and $v_i \leftarrow v_i/\|v_i\|_2$. To make the memory entries approximate the sample features more accurately, the similarity between entry v_i and feature f_i should be as large as possible, *i.e.,* $f_i^T v_i$ is close to 1.

The spread-out property means the feature f_i of every sample x_i in the entire training dataset should be dissimilar with each other, *i.e.,* $f_i^T f_j$ should be close to -1 when $j \neq i$. To further improve the discrimination of the feature learning, the feature should satisfy not only the spread-out property but also the positive-centered property. We assume that the k-nearest neighborhoods of the sample x_i in memory belong to the same class, where we denote the index set of k-nearest neighborhoods as $\mathcal{K}_i$. We can consider all the samples in $\mathcal{K}_i$ along with x_i itself as the positives of x_i (*i.e.,* $\mathcal{K}_i \leftarrow \mathcal{K}_i \cup \{i\}$) and the samples not in $\mathcal{K}_i$ as the negatives of x_i. To concentrate the positives and separate the negatives far away, our IM-spread-out regularization can be formulated as:

$$L_{spread} = \frac{1}{N}\sum_{i=1}^{N} log[1 + \sum_{k\in\mathcal{K}_i}\sum_{\substack{n=1 \\ n\notin\mathcal{K}_i}}^{N} exp(f_i^T v_n - f_i^T v_k + m)] \ , \tag{9}$$

where m is the margin to spread the negatives. The calculation of Eq. (9) with the instant memory bank can be easily implemented by normal matrix operations, which is training efficient. The gradients of L_{spread} with respect to memory entry v_i is derived as follows:

$$\frac{\partial L_{spread}}{\partial v_j} = \frac{1}{N}\sum_{i=1}^{N} T_j/[1 + \sum_{k\in\mathcal{K}_i}\sum_{\substack{n=1 \\ n\notin\mathcal{K}_i}}^{N} exp(f_i^T v_n - f_i^T v_k + m)] \ , \tag{10}$$

$$T_j = \begin{cases} \sum_{\substack{n=1 \\ n\notin\mathcal{K}_i}}^{N} exp(f_i^T v_n - f_i^T v_j + m)\cdot(-f_i), & if \ \ j \in \mathcal{K}_i \\ \sum_{k\in\mathcal{K}_i} exp(f_i^T v_j - f_i^T v_k + m)\cdot f_i, & if \ \ j \notin \mathcal{K}_i \end{cases} . \tag{11}$$

The traditional memory bank methods [49, 52] can only memorize outdated features [18, 55], because a sample's representation in the traditional memory bank is only updated once during every epoch but the network is updated at every iteration [55]. The outdated features memorized in the traditional memory bank may result in sub-optimal on-line feature learning, which is not suitable for handling our on-line feature refinement problem. However, all

the entries in our instant memory bank are updated instantly by Eq. (10) (11) together with the network at every training iteration, *i.e.,* it stores the on-the-fly features of all the training samples, and is able to capture the characteristics of the whole target domain distribution in real time. Our spread loss can also be seen as the variant of the circle loss [40], yet it is significantly different from the circle loss, as the circle loss needs to be trained with mini-batch in a fully supervised way.

Overall loss. By combining Eq. (8) under the supervision of pseudo labels and the label-free regularization Eq. (9) together, the overall objective loss is formulated as:

$$\begin{aligned} L_{joint} &= L_{cls} + L_{tri} + \mu L_{spread} \\ &= (1-\alpha)(\widetilde{L}_{cls} + \widetilde{L}_{tri}) + \alpha(\widehat{L}_{cls} + \widehat{L}_{tri}) + \mu L_{spread}\,, \end{aligned} \tag{12}$$

where α and μ are the parameters to balance the losses. Our proposed off-line pseudo label refinement and on-line feature refinement are conducted alternatively and iteratively over the whole learning process. The details about the overall training procedure can be seen in the supplementary material.

4 EXPERIMENTS

4.1 Datasets and Evaluation Protocol

We conduct experiments on three large-scale person re-ID datasets namely Market1501 [61], DukeMTMC-ReID [63] and MSMT17 [48] The mean average precision (mAP) and Cumulative Matching Characteristic (CMC) curve [15] are used as the evaluation metrics. Specially, we use the rank-1 accuracy (R1), rank-5 accuracy (R5) and rank-10 accuracy (R10) in CMC. There is no post-processing like re-ranking [64] applied at the testing stage.

4.2 Implementation Details

We utilized ResNet50 [19] pre-trained on ImageNet [5] as the backbone network. We add a batch normalization (BN) layer followed by ReLU after the global average pooling (GAP) layer. The stride size of the last residual layer is set as 1. The identity classifier layer is a fully connected layer (FC) followed by softmax function. We resize the image size to 256×128. For data augmentation, we perform random cropping, random flipping, and random erasing [65]. The margin of the triplet loss is set as 0.5, and the margin of our IM-spread-out regularization is set as 0.35. If not specified, we set the parameter $\alpha = 0.5$ and $\mu = 0.1$ to balance the joint loss in Eq (12). We use the Adam [24] optimizer with weight decay 5×10^{-4} and momentum 0.9 to train the network. The learning rate in the pre-training stage follows the warmup learning strategy where the learning rate linearly increases from 3.5×10^{-5} to 3.5×10^{-4} during the first 10 epochs. The learning rate is divided by 10 at the 40th epoch and 70th epoch, respectively, in a total of 80 epochs. We set the batch size as 64 in all our experiments. When training on target data, the learning rate is initialized as 3.5×10^{-4} and divided by 10 at the 20th epoch in a total of 40 epochs. During testing, we extract the $L2-$normalized feature after the BN layer and use Euclidean distance to measure the similarity between the query and gallery images in the testing set. Our model is implemented on PyTorch [35] platform and trained with 4 NVIDIA TITAN XP GPUs.

4.3 Comparisons with State-of-the-Arts

Results on Market1501 and DukeMTMC-ReID. In Table 1, we compare our method with state-of-the-art methods. GAN-based methods include PTGAN [48], SPGAN [6], ATNet [27], CamStyle [69], HHL [66] and PDA-Net [26]; UDAP [39], PCB-PAST [57], SSG [10], ACT [53] and MMT [12] are based on clustering; AD-Cluster [56] combines GAN and clustering; ECN [67], ECN-GPP [68] and MMCL [45] use memory bank. Our method achieves the performance of 78.0% on mAP and 90.9% on rank-1 accuracy when DukeMTMC-ReID→Market1501. Our method outperforms state-of-the-art GAN-based method AD-Cluster [56] by 9.7% on mAP and 4.2% on rank-1 accuracy. Compared with the best clustering-based method MMT [12], our method is more than 6.8% on mAP and 3.2% on rank-1 accuracy when DukeMTMC-ReID→Market1501. Besides, our method outperforms the best memory-bank-based method ECN-GPP [68] by 14.2% on mAP and 6.8% on rank-1 accuracy when DukeMTMC-ReID→Market1501. When using Market1501 as the source dataset and DukeMTMC-ReID as the target dataset, we get the performance of 67.7% on mAP and 82.1% on rank-1 accuracy, which is 13.6% and 9.5% higher than AD-Cluster [56]. Compared with state-of-the-art UDA methods, our method improves the performance by a large margin. It should be noted that our method uses a single model in training and does not use any other images generated by GAN. However, MMT [12] uses dual networks for target domain training, where the amount of parameters is more than twice that of our method. For ECN [67] and ECN-GPP [68], their performances heavily depend on the quality of extra augmented images produced by GAN.

Results on MSMT17. Our method still outperforms state-of-the-art methods on this challenging dataset by a large margin. When considering DukeMTM-ReID as the source dataset, our method achieves the performance of 26.9% on mAP and 55.0% on rank-1 accuracy, which is 3.6% and 4.9% higher than state-of-the-art MMT [12]. When using Market1501 as the source dataset, we get 25.1% performance on mAP and 53.3% on rank-1 accuracy, which surpasses MMT [12] by 2.2% and 4.1%. The improvement of the performance in such a challenging dataset has strongly demonstrated the effectiveness of our method.

4.4 Ablation Study

Comparisons between supervised learning, direct transfer, and baseline. In Table 2, the fully supervised method can be seen as the upper bound for UDA of re-ID, and they use the ground-truth of the target domain to train with classification loss and triplet loss suggested in [32]. When the model is trained on DukeMTMC-ReID and directly tested on Market1501, mAP drops from 70.3% to 28.6%. Furthermore, rank-1 accuracy drops from 84.2% to 58.0%. The baseline method uses the coarse clustering to assign the noisy pseudo labels and trains the model with classification loss and triplet loss, which is mentioned in Section 3.2. The baseline method improves the performance by a larger margin compared with the direct transfer method. When transferring from DukeMTMC-ReID to Market1501, mAP and rank-1 accuracy of the baseline are 39.3% and 27.7% higher than the direct transfer method. Due to the huge domain gap, there is still a large margin in performance when comparing the baseline and the fully supervised method.

Table 1: Comparisons between the proposed method and state-of-the-art unsupervised domain adaptation methods for person Re-ID. The best results are highlighted with bold and the second best results are highlighted with underline.

Methods	Reference	DukeMTMC-ReID → Market1501				Market1501 → DukeMTMC-ReID			
		mAP	R1	R5	R10	mAP	R1	R5	R10
PTGAN [48]	CVPR 2018	-	38.6	-	66.1	-	27.4	-	50.7
PUL [9]	TOMM 2018	20.5	45.5	60.7	66.7	16.4	30.0	43.4	48.5
SPGAN [6]	CVPR 2018	22.8	51.5	70.1	76.8	22.3	41.1	56.6	63.0
ATNet [27]	CVPR 2019	25.6	55.7	73.2	79.4	24.9	45.1	59.5	64.2
TJ-AIDL [47]	CVPR 2018	26.5	58.2	74.8	81.1	23.0	44.3	59.6	65.0
SPGAN+LMP [6]	CVPR 2018	26.7	57.7	75.8	82.4	26.2	46.4	62.3	68.0
CamStyle [69]	TIP 2019	27.4	58.8	78.2	84.3	25.1	48.4	62.5	68.9
HHL [66]	ECCV 2018	31.4	62.2	78.8	84.0	27.2	46.9	61.0	66.7
ECN [67]	CVPR 2019	43.0	75.1	87.6	91.6	40.4	63.3	75.8	80.4
PDA-Net [26]	ICCV 2019	47.6	75.2	86.3	90.2	45.1	63.2	77.0	82.5
UDAP [39]	PR 2020	53.7	75.8	89.5	93.2	49.0	68.4	80.1	83.5
PCB-PAST [57]	ICCV 2019	54.6	78.4	-	-	54.3	72.4	-	-
SSG [10]	ICCV 2019	58.3	80.0	90.0	92.4	53.4	73.0	80.6	83.2
MMCL [45]	CVPR 2020	60.4	84.4	92.8	95.0	51.4	72.4	82.9	85.0
ACT [53]	AAAI 2020	60.6	80.5	-	-	54.5	72.4	-	-
ECN-GPP [68]	TPAMI 2020	63.8	84.1	92.8	95.4	54.4	74.0	83.7	87.4
AD-Cluster [56]	CVPR 2020	68.3	86.7	94.4	96.5	54.1	72.6	82.5	85.5
MMT [12]	ICLR 2020	71.2	87.7	94.9	96.9	65.1	78.0	88.8	**92.5**
Dual-Refinement	This paper	**78.0**	**90.9**	**96.4**	**97.7**	**67.7**	**82.1**	**90.1**	**92.5**
Methods	Reference	DukeMTMC-ReID → MSMT17				Market1501 → MSMT17			
		mAP	R1	R5	R10	mAP	R1	R5	R10
ECN [67]	CVPR 2019	10.2	30.2	41.5	46.8	8.5	25.3	36.3	42.1
SSG [10]	ICCV 2019	13.3	32.2	-	51.2	13.2	31.6	-	49.6
ECN-GPP [68]	TPAMI 2020	16.0	42.5	55.9	61.5	15.2	40.4	53.1	58.7
MMCL [45]	CVPR 2020	16.2	43.6	54.3	58.9	15.1	40.8	51.8	56.7
MMT [12]	ICLR 2020	23.3	50.1	63.9	69.8	22.9	49.2	63.1	68.8
Dual-Refinement	This paper	**26.9**	**55.0**	**68.4**	**73.2**	**25.1**	**53.3**	**66.1**	**71.5**

Table 2: Ablation studies on supervised, direct transfer and variants combined with baseline. LR means off-line pseudo label refinement with hierarchical clustering. IM-SP means on-line feature refinement with the IM-spread-out regularization in Eq. (9). Our method (Baseline with both LR and IM-SP) is comparable to the fully supervised methods.

Methods	DukeMTMC-ReID→Market1501				Market1501→DukeMTMC-ReID			
	mAP	R1	R5	R10	mAP	R1	R5	R10
Fully Supervised (upper bound)	81.2	93.1	97.7	98.6	70.3	84.2	91.7	93.9
Direct Transfer (lower bound)	28.6	58.0	73.7	79.8	27.6	44.5	60.6	66.1
Baseline	67.9	85.7	94.3	96.3	56.4	72.5	84.5	88.2
Baseline with only LR	74.4	88.7	95.1	97.1	65.5	80.0	89.8	92.7
Baseline with only IM-SP	75.5	89.0	95.8	97.5	66.3	80.5	89.6	92.4
Baseline with both LR and IM-SP	**78.0**	**90.9**	**96.4**	**97.7**	**67.7**	**82.1**	**90.1**	**92.5**

Effectiveness of the off-line pseudo label refinement. In Table 2, we evaluate the effectiveness of the off-line pseudo label refinement, denoted as LR in Table 2. Baseline with only LR means that we only conduct the off-line refinement to generate pseudo labels to supervised the on-line metric learning. When testing on DukeMTMC-ReID, Baseline with only LR outperforms the baseline method by 9.1% on mAP and 7.5% on rank-1 accuracy. It has shown that the hierarchical clustering guided pseudo label refinement plays an important role in off-line pseudo label refinement, which will promote the on-line discriminative feature learning.

Effectiveness of the on-line feature refinement. In Table 2 we denote the IM-spread-out regularization as IM-SP. When only considering the on-line feature refinement, *i.e.,* Baseline with only IM-SP, it is 7.6% higher on mAP and 3.3% higher on rank-1 accuracy than Baseline, which is tested on DukeMTMC-ReID→Market1501. It shows that the on-line feature refinement can also improve the performance based without the off-line refinement. From the performance of our method, *i.e.,* Baseline with both LR and IM-SP, we can conclude that both the off-line label and on-line feature refinement are indispensable in our Dual-Refinement method.

Comparisons between the IM-spread-out regularization and its variants. In Table 3, we compare different implementations and variants of the IM-spread-out regularization. The method SP+TM represents substituting the traditional memory bank [50] for our instant memory bank, whose performance is 2.9% lower than our method (SP+IM) on mAP when tested on Market1501.

Table 3: Analysis on our IM-spread-out regularization and its variants. SP: Spread-out regularization. IN: Invariance loss. IM: Instant memory bank. MB: Mini-batch. TM: Traditional memory bank.

Method	Duke→Market1501		Market1501→Duke	
	mAP	R1	mAP	R1
Ours+SP+IM	**78.0**	**90.9**	**67.7**	**82.1**
Ours+SP+TM	75.1	88.7	66.3	80.9
Ours+SP+MB	73.2	88.4	62.6	77.2
Ours+IN+TM	74.9	89.2	66.2	79.7

It shows that the out-dated features stored in memory bank can degenerate the performance. We also compare with the spread-out regularization training with mini-batch (SP+MB) and the invariance loss [67, 68] equipped with traditional memory bank (IN+TM) in Table 3. However, the performances of IN+TM evaluated on DukeMTMC-ReID and Market1501 are all worse than ours. From the above analysis, we can see that the invariance loss, training with mini-batch or the traditional memory bank can not learn the features as discriminative as our method.

Analysis on the quality of pseudo labels. In Figure 3, we evaluate the effects of the pseudo labels' quality. We use F-score [1] to evaluate the quality of clustering, and higher F-score towards 1.0 implies better clustering quality and less noise in pseudo labels. As shown in Figure 3 (a), the quality of the off-line hierarchical clustering in our method is better than the baseline with only coarse clustering. During training, the pseudo labels with descending noise can improve the performance for UDA re-ID as shown in Figure 3 (b), which is evaluated on DukeMTMC-ReID → Market1501.

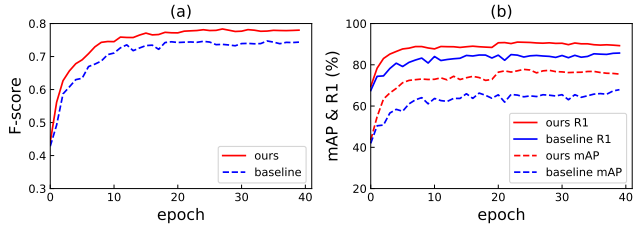


Figure 3: Evaluation on the effects of the quality of pseudo labels. (a) F-score evaluated on the clustering quality of baseline and our method. (b) The performance comparison between baseline and our method during training (DukeMTMC-ReID→Market1501).

4.5 Parameter Analysis.

In this section, we evaluate the influences of four hyper-parameters including the weight α, the weight μ in Eq. (12), the size of $k-$nearest neighborhoods in Eq. (9) and the fine cluster number R in Eq. (6). When evaluating one of the four parameters, we fix the others. We set the parameters $\alpha = 0.5$, $\mu = 0.1$, $k = 6$, $R = 5$ on Market1501 and $R = 2$ on DukeMTMC-ReID based on the following analysis.

Loss weight α. As shown in Figure 4 (a), when the parameter α increases, the performance increases consistently and when $\alpha = 0.5$ it achieves the peak. We can see that, after α achieves 0.5, the performance gets a slight slip, which means that if the pseudo label is refined excessively, extra noise may be induced.

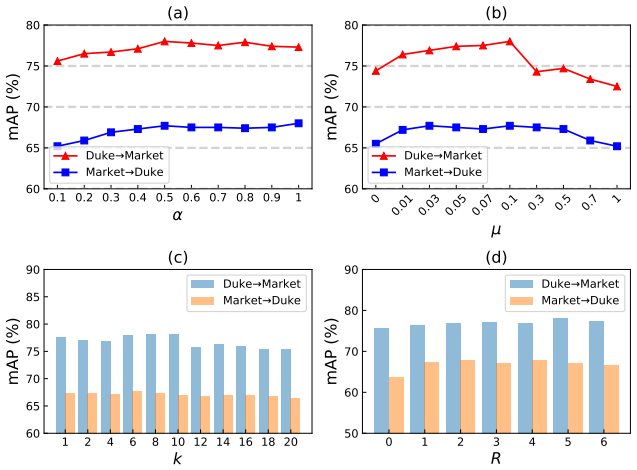


Figure 4: (a) Evaluation on different values of the parameter α in Eq. (12). (b) Evaluation on different values of the parameter μ in Eq. (12). (c) Evaluation on different value of $k-$nearest neighborhoods. (d) Evaluation on different fine clustering number R.

Loss weight μ. As shown in Figure 4 (b), when $\mu = 0$ it means learning without the IM-spread-out regularization, and our method gets low performance. As μ increases, the performance is enhanced greatly and it achieves the peak when $\mu = 0.1$. When μ is larger than 0.1, the performance will drop consistently, it can be explained that the features are spreading-out too much, which will break the inherent similarities between samples.

The number k of $k-$nearest neighborhoods. As shown in Figure 4 (c), when k varies from 1 to 10, its performance stays relatively high, however when k goes larger than 10, its performance drops. It shows that the IM-spread-out regularization is robust to the small k values but when k gets larger, it may degenerate the performance because too many neighborhoods means too many noisy positives.

The fine cluster number R. Figure 4 (d) shows that our method achieves the best performance when $R = 5$ for Duke→Marke1501 and $R = 2$ for Market1501→Duke. These results can reveal that through hierarchical clustering, we can utilize the hierarchical information from the target data itself to further assign more reliable pseudo labels for samples. $R = 0$ means only using the coarse clustering and $R = 1$ means that we calculate the average feature within a coarse cluster and represent it as the fine cluster centroid.

5 CONCLUSION

In this work, we propose a novel approach to jointly refining the off-line pseudo labels and the on-line features for unsupervised domain adaptation in person re-ID. Specially, we design an off-line pseudo label refining strategy by utilizing the hierarchical information in target domain data. We also propose an on-line IM-spread-out regularization to alleviate the effects of the noisy samples. The IM-spread-out regularization is equipped with an instant memory bank that can consider the entire target data during training. Compared with state-of-the-art multi-model methods on UDA re-ID, Dual-Refinement is trained with only a single model and has shown significant improvement of performances.

REFERENCES

[1] Enrique Amigó, Julio Gonzalo, Javier Artiles, and Felisa Verdejo. 2009. A comparison of extrinsic clustering evaluation metrics based on formal constraints. *Information retrieval* 12, 4 (2009), 461–486.

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

[3] Tianlong Chen, Shaojin Ding, Jingyi Xie, Ye Yuan, Wuyang Chen, Yang Yang, Zhou Ren, and Zhangyang Wang. 2019. Abd-net: Attentive but diverse person re-identification. In *Proceedings of the IEEE International Conference on Computer Vision*. 8351–8361.

[4] Yunjey Choi, Minje Choi, Munyoung Kim, Jung-Woo Ha, Sunghun Kim, and Jaegul Choo. 2018. StarGAN: Unified Generative Adversarial Networks for Multi-Domain Image-to-Image Translation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*.

[5] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. 2009. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*. Ieee, 248–255.

[6] Weijian Deng, Liang Zheng, Qixiang Ye, Guoliang Kang, Yi Yang, and Jianbin Jiao. 2018. Image-image domain adaptation with preserved self-similarity and domain-dissimilarity for person re-identification. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 994–1003.

[7] Yueqi Duan, Jiwen Lu, and Jie Zhou. 2019. Uniformface: Learning deep equidistributed representation for face recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 3415–3424.

[8] Martin Ester, Hans-Peter Kriegel, Jörg Sander, and Xiaowei Xu. 1996. A density-based algorithm for discovering clusters in large spatial databases with noise.. In *Kdd*, Vol. 96. 226–231.

[9] Hehe Fan, Liang Zheng, Chenggang Yan, and Yi Yang. 2018. Unsupervised person re-identification: Clustering and fine-tuning. *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)* 14, 4 (2018), 1–18.

[10] Yang Fu, Yunchao Wei, Guanshuo Wang, Yuqian Zhou, Honghui Shi, and Thomas S Huang. 2019. Self-similarity grouping: A simple unsupervised cross domain adaptation approach for person re-identification. In *Proceedings of the IEEE International Conference on Computer Vision*. 6112–6121.

[11] Yaroslav Ganin and Victor Lempitsky. 2015. Unsupervised Domain Adaptation by Backpropagation. In *International Conference on Machine Learning*. 1180–1189.

[12] Yixiao Ge, Dapeng Chen, and Hongsheng Li. 2020. Mutual Mean-Teaching: Pseudo Label Refinery for Unsupervised Domain Adaptation on Person Re-identification. *arXiv preprint arXiv:2001.01526* (2020).

[13] Aritra Ghosh, Naresh Manwani, and PS Sastry. 2015. Making risk minimization tolerant to label noise. *Neurocomputing* 160 (2015), 93–107.

[14] Jacob Goldberger and Ehud Ben-Reuven. 2016. Training deep neural-networks using a noise adaptation layer. (2016).

[15] Douglas Gray, Shane Brennan, and Hai Tao. 2007. Evaluating appearance models for recognition, reacquisition, and tracking. In *Proc. IEEE international workshop on performance evaluation for tracking and surveillance (PETS)*, Vol. 3. Citeseer, 1–7.

[16] Bo Han, Quanming Yao, Xingrui Yu, Gang Niu, Miao Xu, Weihua Hu, Ivor Tsang, and Masashi Sugiyama. 2018. Co-teaching: Robust training of deep neural networks with extremely noisy labels. In *Advances in neural information processing systems*. 8527–8537.

[17] Jiangfan Han, Ping Luo, and Xiaogang Wang. 2019. Deep self-learning from noisy labels. In *Proceedings of the IEEE International Conference on Computer Vision*. 5138–5147.

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

[19] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 770–778.

[20] Alexander Hermans, Lucas Beyer, and Bastian Leibe. 2017. In defense of the triplet loss for person re-identification. *arXiv preprint arXiv:1703.07737* (2017).

[21] Jiabo Huang, Qi Dong, Shaogang Gong, and Xiatian Zhu. 2019. Unsupervised Deep Learning by Neighbourhood Discovery. In *International Conference on Machine Learning*. 2849–2858.

[22] Lu Jiang, Zhengyuan Zhou, Thomas Leung, Li-Jia Li, and Li Fei-Fei. 2018. MentorNet: Learning Data-Driven Curriculum for Very Deep Neural Networks on Corrupted Labels. In *International Conference on Machine Learning*. 2304–2313.

[23] Guoliang Kang, Lu Jiang, Yi Yang, and Alexander G Hauptmann. 2019. Contrastive adaptation network for unsupervised domain adaptation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 4893–4902.

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

[25] Kuang-Huei Lee, Xiaodong He, Lei Zhang, and Linjun Yang. 2018. Cleannet: Transfer learning for scalable image classifier training with label noise. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 5447–5456.

[26] Yu-Jhe Li, Ci-Siang Lin, Yan-Bo Lin, and Yu-Chiang Frank Wang. 2019. Cross-dataset person re-identification via unsupervised pose disentanglement and adaptation. In *Proceedings of the IEEE International Conference on Computer Vision*. 7919–7929.

[27] Jiawei Liu, Zheng-Jun Zha, Di Chen, Richang Hong, and Meng Wang. 2019. Adaptive transfer network for cross-domain person re-identification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 7202–7211.

[28] Weiyang Liu, Rongmei Lin, Zhen Liu, Lixin Liu, Zhiding Yu, Bo Dai, and Le Song. 2018. Learning towards minimum hyperspherical energy. In *Advances in neural information processing systems*. 6222–6233.

[29] Stuart Lloyd. 1982. Least squares quantization in PCM. *IEEE transactions on information theory* 28, 2 (1982), 129–137.

[30] Mingsheng Long, Yue Cao, Jianmin Wang, and Michael Jordan. 2015. Learning Transferable Features with Deep Adaptation Networks. In *International Conference on Machine Learning*. 97–105.

[31] Mingsheng Long, Han Zhu, Jianmin Wang, and Michael I Jordan. 2016. Unsupervised domain adaptation with residual transfer networks. In *Advances in neural information processing systems*. 136–144.

[32] Hao Luo, Wei Jiang, Youzhi Gu, Fuxu Liu, Xingyu Liao, Shenqi Lai, and Jianyang Gu. 2019. A Strong Baseline and Batch Normalization Neck for Deep Person Re-identification. *IEEE Transactions on Multimedia* (2019).

[33] Aditya Menon, Brendan Van Rooyen, Cheng Soon Ong, and Bob Williamson. 2015. Learning from corrupted binary labels via class-probability estimation. In *International Conference on Machine Learning*. 125–134.

[34] Pau Panareda Busto and Juergen Gall. 2017. Open set domain adaptation. In *Proceedings of the IEEE International Conference on Computer Vision*. 754–763.

[35] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. 2019. PyTorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems*. 8024–8035.

[36] Giorgio Patrini, Alessandro Rozza, Aditya Krishna Menon, Richard Nock, and Lizhen Qu. 2017. Making deep neural networks robust to label noise: A loss correction approach. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 1944–1952.

[37] Scott Reed, Honglak Lee, Dragomir Anguelov, Christian Szegedy, Dumitru Erhan, and Andrew Rabinovich. 2014. Training deep neural networks on noisy labels with bootstrapping. *arXiv preprint arXiv:1412.6596* (2014).

[38] Kuniaki Saito, Shohei Yamamoto, Yoshitaka Ushiku, and Tatsuya Harada. 2018. Open set domain adaptation by backpropagation. In *Proceedings of the European Conference on Computer Vision (ECCV)*. 153–168.

[39] Liangchen Song, Cheng Wang, Lefei Zhang, Bo Du, Qian Zhang, Chang Huang, and Xinggang Wang. 2020. Unsupervised domain adaptive re-identification: Theory and practice. *Pattern Recognition* (2020), 107173.

[40] Yifan Sun, Changmao Cheng, Yuhan Zhang, Chi Zhang, Liang Zheng, Zhongdao Wang, and Yichen Wei. 2020. Circle loss: A unified perspective of pair similarity optimization. *arXiv preprint arXiv:2002.10857* (2020).

[41] Yifan Sun, Liang Zheng, Yali Li, Yi Yang, Qi Tian, and Shengjin Wang. 2019. Learning Part-based Convolutional Features for Person Re-identification. *IEEE transactions on pattern analysis and machine intelligence* (2019).

[42] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. 2016. Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2818–2826.

[43] Daiki Tanaka, Daiki Ikami, Toshihiko Yamasaki, and Kiyoharu Aizawa. 2018. Joint optimization framework for learning with noisy labels. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 5552–5560.

[44] Eric Tzeng, Judy Hoffman, Ning Zhang, Kate Saenko, and Trevor Darrell. 2014. Deep domain confusion: Maximizing for domain invariance. *arXiv preprint arXiv:1412.3474* (2014).

[45] Dongkai Wang and Shiliang Zhang. 2020. Unsupervised Person Re-identification via Multi-label Classification. *arXiv preprint arXiv:2004.09228* (2020).

[46] Guanshuo Wang, Yufeng Yuan, Xiong Chen, Jiwei Li, and Xi Zhou. 2018. Learning discriminative features with multiple granularities for person re-identification. In *Proceedings of the 26th ACM international conference on Multimedia*. 274–282.

[47] Jingya Wang, Xiatian Zhu, Shaogang Gong, and Wei Li. 2018. Transferable joint attribute-identity deep learning for unsupervised person re-identification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2275–2284.

[48] Longhui Wei, Shiliang Zhang, Wen Gao, and Qi Tian. 2018. Person transfer gan to bridge domain gap for person re-identification. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 79–88.

[49] Zhirong Wu, Alexei A Efros, and Stella X Yu. 2018. Improving generalization via scalable neighborhood component analysis. In *Proceedings of the European Conference on Computer Vision (ECCV)*. 685–701.

[50] Zhirong Wu, Yuanjun Xiong, Stella X Yu, and Dahua Lin. 2018. Unsupervised feature learning via non-parametric instance discrimination. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 3733–3742.

[51] Xiaobo Xia, Tongliang Liu, Nannan Wang, Bo Han, Chen Gong, Gang Niu, and Masashi Sugiyama. 2019. Are Anchor Points Really Indispensable in Label-Noise Learning?. In *Advances in Neural Information Processing Systems*. 6835–6846.

[52] Tong Xiao, Shuang Li, Bochao Wang, Liang Lin, and Xiaogang Wang. 2017. Joint detection and identification feature learning for person search. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 3415–3424.

[53] Fengxiang Yang, Ke Li, Zhun Zhong, Zhiming Luo, Xing Sun, Hao Cheng, Xiaowei Guo, Feiyue Huang, Rongrong Ji, and Shaozi Li. 2019. Asymmetric Co-Teaching for Unsupervised Cross Domain Person Re-Identification. *arXiv preprint arXiv:1912.01349* (2019).

[54] Mang Ye, Jianbing Shen, Gaojie Lin, Tao Xiang, Ling Shao, and Steven CH Hoi. 2020. Deep Learning for Person Re-identification: A Survey and Outlook. *arXiv preprint arXiv:2001.04193* (2020).

[55] Mang Ye, Xu Zhang, Pong C Yuen, and Shih-Fu Chang. 2019. Unsupervised embedding learning via invariant and spreading instance feature. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 6210–6219.

[56] Yunpeng Zhai, Shijian Lu, Qixiang Ye, Xuebo Shan, Jie Chen, Rongrong Ji, and Yonghong Tian. 2020. AD-Cluster: Augmented Discriminative Clustering for Domain Adaptive Person Re-identification. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

[57] Xinyu Zhang, Jiewei Cao, Chunhua Shen, and Mingyu You. 2019. Self-training with progressive augmentation for unsupervised cross-domain person re-identification. In *Proceedings of the IEEE International Conference on Computer Vision*. 8222–8231.

[58] Xu Zhang, Felix X Yu, Sanjiv Kumar, and Shih-Fu Chang. 2017. Learning spread-out local feature descriptors. In *Proceedings of the IEEE International Conference on Computer Vision*. 4595–4603.

[59] Zhilu Zhang and Mert Sabuncu. 2018. Generalized cross entropy loss for training deep neural networks with noisy labels. In *Advances in neural information processing systems*. 8778–8788.

[60] Kai Zhao, Jingyi Xu, and Ming-Ming Cheng. 2019. Regularface: Deep face recognition via exclusive regularization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 1136–1144.

[61] Liang Zheng, Liyue Shen, Lu Tian, Shengjin Wang, Jingdong Wang, and Qi Tian. 2015. Scalable person re-identification: A benchmark. In *Proceedings of the IEEE international conference on computer vision*. 1116–1124.

[62] Liang Zheng, Yi Yang, and Alexander G Hauptmann. 2016. Person re-identification: Past, present and future. *arXiv preprint arXiv:1610.02984* (2016).

[63] Zhedong Zheng, Liang Zheng, and Yi Yang. 2017. Unlabeled samples generated by gan improve the person re-identification baseline in vitro. In *Proceedings of the IEEE International Conference on Computer Vision*. 3754–3762.

[64] Zhun Zhong, Liang Zheng, Donglin Cao, and Shaozi Li. 2017. Re-ranking person re-identification with k-reciprocal encoding. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 1318–1327.

[65] Zhun Zhong, Liang Zheng, Guoliang Kang, Shaozi Li, and Yi Yang. 2017. Random erasing data augmentation. *arXiv preprint arXiv:1708.04896* (2017).

[66] Zhun Zhong, Liang Zheng, Shaozi Li, and Yi Yang. 2018. Generalizing a person retrieval model hetero-and homogeneously. In *Proceedings of the European Conference on Computer Vision (ECCV)*. 172–188.

[67] Zhun Zhong, Liang Zheng, Zhiming Luo, Shaozi Li, and Yi Yang. 2019. Invariance matters: Exemplar memory for domain adaptive person re-identification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 598–607.

[68] Zhun Zhong, Liang Zheng, Zhiming Luo, Shaozi Li, and Yi Yang. 2020. Learning to adapt invariance in memory for person re-identification. *IEEE Transactions on Pattern Analysis and Machine Intelligence* (2020).

[69] Z Zhong, L Zheng, Z Zheng, S Li, and Y Yang. 2019. CamStyle: A Novel Data Augmentation Method for Person Re-Identification. *IEEE Transactions on Image Processing* (2019).

[70] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. 2017. Unpaired Image-to-Image Translation using Cycle-Consistent Adversarial Networks. In *Computer Vision (ICCV), 2017 IEEE International Conference on*.