Intra-Inter Camera Similarity for Unsupervised Person Re-Identification

Shiyu Xuan Shiliang Zhang
Department of Computer Science, School of EECS, Peking University
Beijing 100871, China
shiyu_xuan@stu.pku.edu.cn, slzhang.jdl@pku.edu.cn

Abstract

Most of unsupervised person Re-Identification (Re-ID) works produce pseudo-labels by measuring the feature similarity without considering the distribution discrepancy among cameras, leading to degraded accuracy in label computation across cameras. This paper targets to address this challenge by studying a novel intra-inter camera similarity for pseudo-label generation. We decompose the sample similarity computation into two stage, i.e.*, the intra-camera and inter-camera computations, respectively. The intra-camera computation directly leverages the CNN features for similarity computation within each camera. Pseudo-labels generated on different cameras train the re-id model in a multi-branch network. The second stage considers the classification scores of each sample on different cameras as a new feature vector. This new feature effectively alleviates the distribution discrepancy among cameras and generates more reliable pseudo-labels. We hence train our re-id model in two stages with intra-camera and inter-camera pseudo-labels, respectively. This simple intra-inter camera similarity produces surprisingly good performance on multiple datasets,* e.g.*, achieves rank-1 accuracy of 89.5% on the Market1501 dataset, outperforming the recent unsupervised works by 9+%, and is comparable with the latest transfer learning works that leverage extra annotations.*

1. Introduction

Person Re-Identification (ReID) aims to match a given query person in an image gallery collected from non-overlapping camera networks [41, 23]. Thanks to the powerful deep Convolutional Neural Network (CNN), great progresses have been made in fully-supervised person ReID [38, 25, 19, 18, 30]. To relieve the requirement of expensive person ID annotation, increasing efforts are being made on unsupervised person ReID [36, 49, 12, 33, 32, 6], *i.e.*, training with labeled source data and unlabeled target data, or fully relying on unlabeled target data for training.

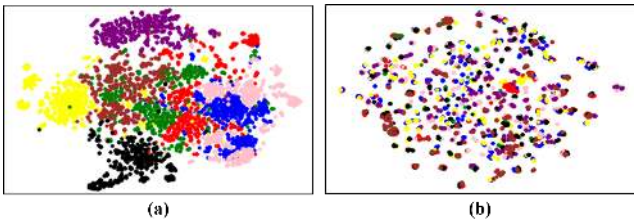

Figure 1. t-SNE visualization [20] of features from a subset of DukeMTMC-ReID. Different colors indicate samples from different cameras. Baseline features in (a) suffer from feature distribution discrepancies among cameras. Features learned by our method are visualized in (b), where features from different cameras have similar distribution.

Existing unsupervised person ReID works can be grouped into three categories: a) using domain adaptation to align distributions of features between source and target domains [32, 15, 29], b) applying Generative Adversarial Network (GAN) to perform image style transfer, meanwhile maintaining the identity annotations on source domains [49, 31, 45, 3], and c) generating pseudo-labels on target domains for training via assigning similar images with similar labels via clustering, KNN search, *etc.* [16, 5, 35, 6, 28]. The first two categories define unsupervised person ReID as a transfer learning task, which leverages the labeled data on source domains. Generating pseudo-labels makes it possible to train ReID models with fully unsupervised setting, thus shows better flexibility.

Most of pseudo-labels prediction algorithms share a similar intuition, *i.e.*, first computing sample similarities, then assigning similar samples identified by clustering or KNN with similar labels. During this procedure, the computed sample similarity largely decides the ReID accuracy. To generate high quality pseudo-labels, samples of the same identity are expected share larger similarities than with those from different identities. However, the setting of unsupervised person ReID makes it difficult to learn reliable

The code is available at https://github.com/SY-Xuan/IICS.

sample similarities, especially for samples from different cameras. For example, each identity can be recorded by multi-cameras with varied parameters and environments. Those factors may significantly change the appearance of the identity. In other words, the domain gap among cameras makes it difficult to identify samples of the same identity, as well as to optimize of intra-class feature similarity. We illustrated the feature distribution of different cameras in Fig. 1 (a).

This paper addresses the above challenge by studying a more reasonable similarity computation for pseudo-labels generation. Identifying samples of the same identify within the same camera is easier than performing the same task among different cameras. Meanwhile, domain gaps can be alleviated by learning generalizable classifiers. We hence decompose the sample similarity computation into two stages to progressively seek reliable pseudo-labels. The first stage computes sample similarity within each camera with CNN features. This "intra-camera" distance guides pseudo-label generation within each camera by clustering samples and assigning samples within the same cluster with the same label. Independent pseudo-labels in C cameras hence train the ReID model with a C-branch network, where the shared backbone is optimized by multiple tasks, and each branch is optimized by a specific classification task within the same camera. This stage simplifies pseudo-label generation, thus ensures high quality pseudo-labels and efficient backbone optimization.

The second stage proceeds to compute sample similarities across cameras. Sample similarity computed with CNN features can be affected by domain gap, *e.g.*, large domain gap decreases the similarity among samples of the same identity as illustrated in Fig. 1 (a). As discussed in previous works [4, 26], the classification probability is more robust the domain gap than raw features. We alleviate the domain gap by enhancing the generalization ability of trained classifiers in the first stage. Specifically, we classify each sample with C classifiers, and use their classification scores as a new feature vector. To ensure the classification scores robust to the domain gap, each classifier trained on one camera should generalize well on other cameras. This is achieved with the proposed Adaptive Instance and Batch Normalization (AIBN), which enhances the generalization ability of classifier without reducing their discriminative ability. Classification scores produced by C classifiers are hence adopted to calculate the "inter-camera" similarity to seek pseudo-labels across cameras. The ReID model is finally optimized by pseudo-labels generated with both stages. Distribution of features learned by our method is illustrated in Fig. 1 (b), where the domain gaps between cameras are effectively eliminated.

We test our approach in extensive experiments on multiple ReID datasets including Market1501 [41], DukeMTMC-ReID [23] and MSMT17 [31], respectively. Experiments show that each component in our approach is valid in boosting the ReID performance. A complete approach consisting of intra-inter camera similarities exhibits the best performance. For instance, without leveraging any annotations, our approach achieves rank-1 accuracy of 89.5% on the Market1501 dataset, outperforming the recent unsupervised works by 9+%. Our method also performs better than many recent transfer learning works that leverage extra annotations. For instance, the recent MMT [7] and NRMT [40] achieves lower rank-1 accuracies of 87.7% and 87.8% respectively, even they leverage extra annotations on DukeMTMC-ReID [23] for training.

The promising performance demonstrates the validity of our method, which decomposes the similarity computation into two stages to progressively seek better pseudo-labels for training. This strategy is more reasonable than directly predicting pseudo-labels across cameras in that, it effectively alleviates the domain gap between cameras. Besides that, those two stages corresponds to different difficulty in predicting pseudo-labels, thus are complementary to each other in optimizing the ReID model. To the best of our knowledge, this is an original work studying better similarity computation strategies in unsupervised person ReID.

2. Related Work

This work is closely related to unsupervised person ReID and, domain adaptation and generalization. Recent works on those two topics will be reviewed briefly in following paragraphs.

Unsupervised person ReID has been studied with three types of methods, *i.e.*, by distribution alignment, training GANs, and generating pseudo-labels, respectively. Distribution alignment based methods follow the traditional domain adaptation methods [8, 24] to align the feature distribution of source and target domains. Wu *et al*. [32] proposed a Camera-Aware Similarity Consistency Loss to align the pairwise distribution of intra-camera matching and cross-camera matching. Lin *et al*. [15] utilized Maximum Mean Discrepancy (MMD) distance [8] to align the distribution of mid-level features from source and the target domains. Some other methods use GANs [46] to perform image-to-image style translation to transfer source images into target style. Zou *et al*. [49] disentangled id-related/unrelated features to enforce the adaptation to work on the id-related feature space. Wei *et al*. [31] proposed person transfer GAN, which can transfer person images with the style of target dataset and keep the identity label of the person.

Pseudo-labels based methods first generate pseudo-labels by formulating certain rules based on sample similarity, then train the ReID model with those pseudo-labels. The quality of computed pseudo-labels determines the perfor-

mance of these methods. Unsupervised clustering method is one of the most commonly used methods to generate pseudo-labels [12, 37, 28, 35, 17]. Fan *et al*. [5] used standard k-means clustering method to generate pseudo-labels and fine-tuned model with these labels. Lin *et al*. [16] proposed a bottom-up clustering approach to generate pseudo-labels. To avoid re-initializing the classifier at each epoch, an extra memory bank was added into the network. Wang *et al*. [28] formulated unsupervised person ReID as multi-label classification task and used memory bank to train the network. NRMT [40], MMT [7] and MEB-Net [36] used mutual-training [39] to reduce the influence of low-quality pseudo-labels.

Domain adaptation and generalization are commonly considered to improve the generalization ability of CNN models. Recently, some works have found that Batch Normalization (BN) [11] and Instance Normalization (IN) [27] could improve the network's generalization ability on multiple domains [48, 21, 1]. IBN-Net [21] integrated IN and BN to enhance the generalization capacity of CNNs to unseen domain without fine-tuning. Chang *et al*. [1] improved the performance of unsupervised domain adaptation using domain-specific BN. Zhuang *et al*. [48] designed a camera-based BN to alleviate the distribution gap between a camera pair in person ReID. Their method improved the generalization ability of the model across unseen cameras.

Most pseudo-labels based methods try to mitigate the impact of low-quality pseudo-labels or find high-quality part from generated pseudo-labels. The work most similar to us is [47] which utilizes extra ID labels within each camera as supervision and simply uses classification results to find matching candidates across cameras during inter-camera training. Different from those works, our work is motivated to seek a reliable similarity by progressively eliminating negative influences of pose variances, illumination, occlusions through intra-camera training, and domain gap through inter-camera training. This leads to the proposed AIBN and inter-camera similarities. As shown in our experiments, our method produces surprisingly good performance on multiple datasets.

3. Methodology

3.1. Formulation

Given an unlabeled person image dataset with camera information $\mathcal{X} = \{\mathcal{X}^c\}$, where $\mathcal{X}^c$ is a collection of person images and the superscript $c = 1 : C$ denotes the index of cameras, respectively. Our goal is to train a person ReID model on $\mathcal{X}$. For any query person image q, the ReID model is expected to produce a feature vector to retrieve image I_g containing the same person from a gallery set G. The trained ReID model should guarantee q share more similar

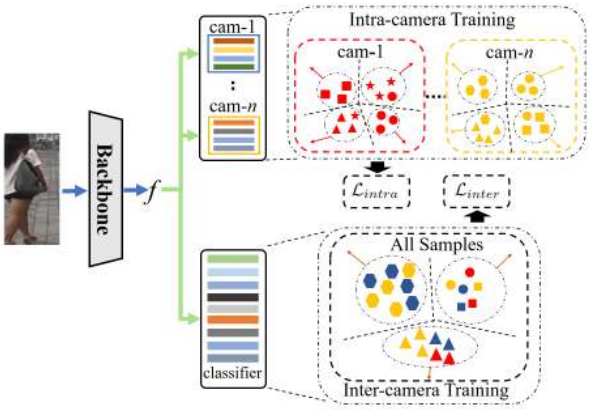

Figure 2. Illustrations of the proposed method for unsupervised person ReID. The Intra-camera training is conducted within each camera separately. It generates pseudo-labels by clustering using intra-camera similarity computed with the CNN feature $\boldsymbol{f}$. The inter-camera training generates pseudo-labels by clustering all samples using the inter-camera similarity, which is computed with classification scores. These two stages are executed alternately during the whole training process to optimize the ReID feature $\boldsymbol{f}$ with complementary intra and inter camera losses.

feature with I_g than with other images in G, *i.e.*,

$$g^* = \arg\max_{g \in G} \text{sim}(\boldsymbol{f}_g, \boldsymbol{f}_q), \tag{1}$$

where $\boldsymbol{f} \in \mathbb{R}^d$ is a d-dimensional feature vector extracted by the person ReID model. $\text{sim}(\cdot)$ computes the feature similarity.

Suppose a person p is captured by cameras in $\mathcal{X}$, the collection of images of p and $\mathcal{X}$ can be denoted as $\mathcal{X}_p$ and $\mathcal{X} = \{\mathcal{X}_p\}_{p=1:P}$, respectively, where P is the total number of persons in $\mathcal{X}$. An estimation towards $\{\mathcal{X}_p\}_{p=1:P}$ would make the optimization to Eq. (1) possible, *e.g.*, through minimizing feature distance within each $\{\mathcal{X}_p\}$, meanwhile enlarging distance between $\{\mathcal{X}_i\}$ and $\{\mathcal{X}_j\}$ with $i \neq j$. A commonly used strategy is performing clustering on $\mathcal{X}$ to generate pseudo-labels. The training objective in label prediction could be conceptually denoted as,

$$\mathcal{T}^* = \arg\min_{\mathcal{T}} \mathcal{D}(\mathcal{T}, \{\mathcal{X}_p\}_{p=1:P}), \tag{2}$$

where $\mathcal{T}$ denotes the clustering result and $\mathcal{D}(\cdot)$ computes its differences with $\{\mathcal{X}_p\}_{p=1:P}$.

The optimization towards Eq. (2) requires to identify images of the same person across cameras. This could be challenging because the appearance of an image can be affected by complicated factors. Using $I_n^c \in \mathcal{X}^c$ to denote an image of person p captured by camera c, we conceptually describe the appearance of I_n^c as,

$$I_n^c \doteq A_p + S_c + E_n, \tag{3}$$

where A_p denotes the appearance of the person p. S_c represents the setting of cameras c including its parameters, viewpoint, environment, *etc.*, that affect the appearance of its captured images. We use E_n to represent other stochastic factors affecting the appearance of I_n^c including pose, illumination, occlusions, *etc.*

According to Eq. (3), the challenge of Eq. (2) lies in learning feature $\boldsymbol{f}$ to alleviate the effects of S_c and E_n, and finding image clusters across cameras according to A_p. To conquer this challenge, we propose to perform pseudo-label prediction with two stages to progressively enhance the robustness of $\boldsymbol{f}$ to E_n and S_c, respectively.

The robustness to E_n can be enhanced by performing Eq. (2) within each camera using existing pseudo-label generation methods, then training $\boldsymbol{f}$ according to the clustering result. Suppose the clustering result for the c-th camera is $\mathcal{T}^c$, the training loss on c-th cameras can be represented as,

$$\mathcal{L}_{intra}^c = \sum_{I_n \in \mathcal{X}^c, I_n \in \mathcal{T}_m^c} \text{loss}^c(\boldsymbol{f}_n, m), \tag{4}$$

where m denotes the cluster ID, which is used as the pseudo-label of I_n for loss computation. To ensure the robustness of $\boldsymbol{f}$ towards complicated E_n under different cameras, Eq. (4) can be computed on different cameras by sharing the same $\boldsymbol{f}$. This leads to a multi-branch CNN, where each branch corresponds to a classifier, and their shared backbone learns the feature $\boldsymbol{f}$.

The robustness to S_c is enhanced in the second stage by clustering images of the same person across cameras. Directly using the learned $\boldsymbol{f}$ to measure similarity for clustering suffers from S_c. We propose to compute a more robust inter-camera similarity. The intuition is to train classifiers with domain adaption strategies to gain enhanced generalization ability, *e.g.*, the classifier on camera c is expected to be discriminative on other cameras. We thus could identify images of the same person from different cameras based on their classification scores, and enlarge their similarity with the inter-camera similarity, *i.e.*,

$$\text{SIM}_{inter}(I_m, I_n) = \text{sim}(\boldsymbol{f}_m, \boldsymbol{f}_n) + \mu\Delta(\mathbf{s}_m, \mathbf{s}_n), \tag{5}$$

where $\mathbf{s}_n$ denotes the classification score of image I_n. $\Delta(\mathbf{s}_m, \mathbf{s}_n)$ is the probability that I_m and I_n are from the same identity. Eq. (5) enlarges the similarity of two images from different cameras, if they are identified as the same person. It effectively alleviates S_c during similarity computation and image clustering. We hence further optimize $\boldsymbol{f}$ with the inter-camera loss based on the clustering result $\mathcal{T}$, *i.e.*,

$$\mathcal{L}_{inter} = \sum_{I_n \in \mathcal{T}_m} \text{loss}(\boldsymbol{f}_n, m). \tag{6}$$

Our method is progressively optimized by Eq. (4) and Eq. (6), respectively to gain $\boldsymbol{f}$ with the robustness to S_c, E_n. Their detailed computations, as well the implementation of $\Delta(\cdot)$ and generalization ability enhancement will be presented in the following parts.

3.2. The Intra-camera Training

Fig. 2 illustrates our framework, where the person ReID feature $\boldsymbol{f}$ is optimized by two stages. The intra-camera training stage divides the training set $\mathcal{X}$ into subsets $\{\mathcal{X}^c\}$ according to the camera index of each image. Then, it performs clustering on each subset according to the similarity computed with feature $\boldsymbol{f}$. Assigning images within each cluster with identical label turns each $\mathcal{X}^c$ into a labeled dataset, allowing the function $\text{loss}^c(\cdot)$ in $\mathcal{L}_{intra}^c$ can be computed as

$$\text{loss}^c(\boldsymbol{f}_n, m) = \ell\left(\mathcal{F}\left(\boldsymbol{w}^{\boldsymbol{c}}, \boldsymbol{f}_n\right), m\right), \tag{7}$$

where $\mathcal{F}(\boldsymbol{w}^{\boldsymbol{c}}, \cdot)$ denotes a classier with learnable parameters $\boldsymbol{w}^{\boldsymbol{c}}$. $\ell(\cdot)$ computes the softmax cross entropy loss on classifier outputs and the groundtruth label m.

As illustrated in Fig. 2, the intra-camera training treats each cameras as a training task and trains the $\boldsymbol{f}$ with multiple tasks. The overall training loss can be denoted as

$$\mathcal{L}_{intra} = \sum_{c=1}^{C} \mathcal{L}_{intra}^c, \tag{8}$$

where C is the total number of cameras. As discussed in Sec. 3.1, Eq. (8) effectively boosts the discriminative power of $\boldsymbol{f}$ within each camera. Besides that, optimizing $\boldsymbol{f}$ on multi-tasks boosts its discriminative power on different domains, which in-turn enhances the generalization ability of learned classifiers.

3.3. The Inter-camera Training

To estimate the probability that two samples from different cameras belong to the same identity, a domain-independent feature is needed. As discussed in related works [4, 26], samples belonging to the same identity should have similar distribution of classification probability produced by each classifier. We use the jaccard similarity of classification probability to compute the $\Delta(\mathbf{s}_m, \mathbf{s}_n)$, which reflects the probability that I_m and I_n are from the same identity

$$\Delta(\mathbf{s}_m, \mathbf{s}_n) = \frac{\mathbf{s}_m \cap \mathbf{s}_n}{\mathbf{s}_m \cup \mathbf{s}_n}, \tag{9}$$

where $\cap$ is the element-wise min of two vectors and $\cup$ is the element-wise max of two vectors. The classification score $\boldsymbol{s}_m$ is acquired by concatenating the classification scores from C classifiers,

$$\begin{aligned} \boldsymbol{s}_m &= [\boldsymbol{s}_m^1, \cdots, \boldsymbol{s}_m^c], \\ \mathbf{s}_m^c &= [p(1|\boldsymbol{f}_m, \boldsymbol{w}_{\boldsymbol{c}}), \cdots, p(k|\boldsymbol{f}_m, \boldsymbol{w}_{\boldsymbol{c}})], \end{aligned} \tag{10}$$

where $p(k|\boldsymbol{f}_m, \boldsymbol{w_c})$ is the classification probability of at class k computed by the classifier $\mathcal{F}(\boldsymbol{w_c}; \cdot)$ and $\mathbf{s}_m^c$ is the classification score of image I_m on camera c.

To make the $\Delta(\mathbf{s}_m, \mathbf{s}_n)$ work as expected, classifier trained on each camera needs to generalize well on other cameras. The $\boldsymbol{f}$ trained by multi-task learning in the intra-camera stage provides basic guarantee for generalization ability of the network. In order to further improve generalization of different classifiers, we propose AIBN which will be described in detail at Sec. 3.4.

With $\Delta(\boldsymbol{s}_m, \boldsymbol{s}_n)$, clustering can be performed based on inter-camera similarity to generate pseudo-labels on $\mathcal{X}$. Then Eq. (6) can be computed as:

$$\mathcal{L}_{inter} = \frac{1}{|\mathcal{B}|} \sum_{I_n \in \mathcal{B}} \ell\left(\mathcal{F}\left(\boldsymbol{w}, \boldsymbol{f}_n\right), m\right) + \lambda L_{triplet}, \quad (11)$$

where $\mathcal{B}$ is a training mini-batch, ℓ is the softmax cross entropy loss, m is its pseudo-label assigned by clustering result, λ is loss weight and $L_{triplet}$ is the hard-batch triplet loss [10]. We randomly select P clusters and K samples from each cluster to construct the training mini-batch $\mathcal{B}$.

3.4. Adaptive Instance and Batch Normalization

As discussed above, we propose AIBN to boost the generalization ability of learned classifiers. Instance Normalization (IN) [27] can make the network invariant to appearance changes. However, IN reduces the inter-class variance, making the network less discriminative. Different from IN, Batch Normalization (BN) [11] retains variations across different classes and reduces the internal covariate shift during network training. In other words, IN and BN are complementary to each other.

In order to gain the advantages of both IN and BN, we propose the AIBN. It is computed by linearly fusing the statistics (mean and var) obtained by IN and BN, *i.e.*,

$$\hat{\mathbf{x}}[i,j,n] = \gamma \frac{\mathbf{x}[i,j,n] - (\alpha\mu_{bn} + (1-\alpha)\mu_{in})}{\sqrt{\alpha\sigma_{bn}^2 + (1-\alpha)\sigma_{in}^2 + \epsilon}} + \beta, \quad (12)$$

where $\mathbf{x}[i,j,n] \in \mathbb{R}^{H\times W\times N}$ is the feature map of each channel, μ_{bn} and σ_{bn} are the mean and variance calculated by BN, μ_{in} and σin are the mean and variance calculated by IN, γ and β are affine parameters and α is a learnable weighting parameter. The optimization of α can be conducted with back-propagation during CNN training. We add no constraints to α during training back-propagation. During network forward inference using Eq. (12), we clamp α into $[0, 1]$ to avoid negative values.

Discussion: To show the effects of two training stages, we visualize the distribution of similarities between samples in Fig. 3. We use Red color to indicate the distribution of similarity between samples from different cameras. It is clear in Fig. 3 (a) that, the Red color is mixed with the Yellow and Green color, which indicate the distribution of similarity between samples of different identities. Therefore, clustering using similarity in Fig. 3 (a) would lead to poor performance. It also can be observed that, the intra-camera training and inter-camera training progressively improves the discriminative power of feature similarity. The inter-camera training produces the most reliable similarity. More evaluations will be presented in the following section.

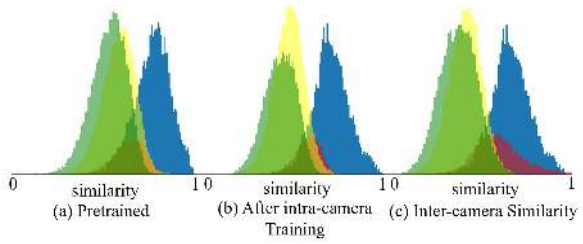

Figure 3. The distribution of similarity on DukeMTMC-ReID. Blue and Red color indicates the distribution of similarity between samples of the same identity from the same camera and different cameras, respectively. Yellow and Green color indicates the distribution of similarity between samples of different identities from the same camera and different cameras, respectively. To show an intuitive visualization of real data, similarities are normalized into [0, 1].

4. Experiments

4.1. Dataset and Evaluation Metrics

We evaluate our methods on three commonly used person ReID datasets, *e.g.*, DukeMTMC-ReID [23], Market1501 [41], and MSMT17 [31], respectively.

DukeMTMC-ReID is collected from 8 non-overlapping camera views, containing 16,522 images of 702 identities for training, 2,228 images of the other 702 identities for query, and 17,661 gallery images.

Market1501 is a large-scale dataset captured from 6 cameras, containing 32,668 images with 1,501 identities. It is divided into 12,936 images of 751 identities for training and 19,732 images of 750 identities for testing.

MSMT17 is a newly published person ReID dataset. It contains 126,441 images of 4,101 identities captured from 15 cameras. It is divided into 32,621 images of 1,041 identities for training and 93,820 images of 3,060 identities for testing.

During training, we only use images and camera labels from the training set of each dataset and do not utilize any other annotation information. Performance is evaluated by the Cumulative Matching Characteristic (CMC) and mean Average Precision (mAP).

4.2. Implementation Details

We use ResNet-50 [9] pre-trained on ImageNet [2] as backbone to extract the feature. The layers after pooling-

5 layer are removed and a BN-Neck [19] is added behind it. During testing and clustering, we extract the pooling-5 feature to calculate the similarity. All models are trained with PyTorch.

During training, the input image is resized to 256×128. Image augmentation strategies such as random flipping and random erasing are performed. At each round we perform intra-camera stage and inter-camera stage in order. The number of training round is set as 40.

At intra-camera training stage, the batch size is 8 for each camera. The SGD is used to optimize the model. The learning rate for ResNet-50 base layers is 0.0005, and the one for other layers is 0.005.

At inter-camera training stage, a mini-batch of 64 is sampled with $P = 16$ randomly selected clusters and $K = 4$ randomly sampled images per cluster. The SGD is also used to optimize the model. The learning rate for ResNet-50 base layers is 0.001, and the one for other layers is 0.01. The loss weight λ in Eq. (11) is fixed to 1. Margin in triplet loss is fixed to 0.3. The training progressively uniforms the distribution of features from different cameras. Therefore, the initial μ in Eq. (5) is set as 0.02, and follows the poly policy for decay.

For Market1501 and DukeMTMC-ReID, we train the model for 2 epochs at both stages. For MSMT17 we train the model for 12 epochs at intra-camera stage and 2 epochs at inter-camera stage. We use the standard Agglomerative Hierarchical method [22] for clustering. For Market1501 and DukeMTMC-ReID, the number of clusters is 600 for each camera at intra-camera stage and 800 at inter-camera stage. For MSMT17, the number of clusters is 600 for each camera at intra-camera stage and 1200 at inter-camera stage.

Although additional clustering within each camera is performed, this is more efficient than clustering on the entire set. Therefore, the computational complexity of our method is acceptable. It takes about 4-5 hours to finish the training with a GPU on Market1501.

For the AIBN, the mixture weight α is initialized to 0.5. We replace all BNs in layer3 and layer4 of ResNet-50 with AIBN. Mixture weights are shared at each BottleNeck module. The detailed analysis of this component is performed in Section. 4.3.

4.3. Ablation Study

The impact of individual components. In this section we evaluate the effectiveness of each component in our method, the experimental results of each setting are summarized in Table 1. As shown in the table, when only inter-camera stage is used for training, performance is not satisfactory. This shows that, the similarity between samples from different cameras is unreliable. Clustering directly using this similarity can lead to a poor performance. The rank-1 accuracy on Market1501 and DukeMTMC-ReID can achieve 71.6% and 62.9%, respectively, with only intra-camera training stage. This indicates that the similarity between samples from the same camera is reliable. Without considering the distribution gap between cameras, the addition of the inter-camera training stage leads to a decrease in performance on DukeMTMC-ReID. It is clear that although the feature produced by the model has been improved after the intra-camera stage, the similarity between samples from different cameras is still unreliable.

Dataset	Market		Duke	
Settings	mAP	Rank-1	mAP	Rank-1
Stage 1	45.0	71.6	41.4	62.9
Stage 2*	26.6	48.8	7.2	16.7
Stage1 + Stage 2*	55.1	78.6	35.8	54.2
Stage1 + Stage 2 + Eq. (9)	69.1	88.2	57.0	75.7
Stage1 + Stage 2 + Eq. (5)	72.1	88.8	59.1	76.9

Table 1. Ablation study on individual components of IICS. Stage 1 denotes intra-camera training stage. Stage 2 denotes inter-camera training stage. * denotes the CNN features similarity is used in stage 2.

Our method achieves the best performance when the inter camera similarity in Eq. (5) is used in inter-camera training stage. It demonstrates that the inter-camera similarity is more effective than CNN features similarity and is crucial for the our performance enhancement. Since the jaccard similarity can be used to calculate the probability that samples belong to the same identity, Eq. (9) can also be used as similarity for inter-camera clustering. This setting can also achieve good performance, which means the jaccard similarity is also more robust to domain gap between cameras. Experimental results show that each component in our method is important for performance boost, and their combination achieves the best performance.

The impact of AIBN. To test the validity of AIBN, we test it with different training settings. The results are summarized in Table 2. Replacing BNs in backbone with IN can improve the performance on DukeMTMC-ReID but decrease performance on Market1501, which shows that only applying IN can not bring stable performance enhancement. AIBN can improve the performance on both dataset even though the mixture weight α of AIBN is fixed at 0.5 during training, which indicates that the combination of IN and BN brings more stable performance gains. Optimizing mixture weight α can further improve the performance on Market1501 and DukeMTMC-ReID. It is clear that AIBN can improve the generalization ability of trained network on different domains and cameras. IBN [21] is another method of combining BN and IN to improve network generalization ability. The result shows that our AIBN substantially outperforms IBN.

Dataset	Market		Duke	
Settings	mAP	Rank-1	mAP	Rank-1
Backbone	67.1	85.5	51.4	71.3
+ IBN [21]	59.8	81.1	35.4	56.3
+ IN	59.6	83.2	53.0	72.7
+ AIBN (fixed)	70.7	88.0	56.9	75.2
+ AIBN	72.1	88.8	59.1	76.9

Table 2. Ablation study on AIBN. The ResNet-50 is used as backbone.

Training Set	Dataset	Market		Duke	
	Settings	mAP	Rank-1	mAP	Rank-1
Market	w/o AIBN	79.9	92.0	22.4	38.2
	w/ AIBN	80.0	92.0	29.5	49.5
Duke	w/o AIBN	21.8	48.9	68.7	83.9
	w/ AIBN	27.2	54.5	69.2	84.8

Table 3. Evaluation on the generalization ability of backbone with/without AIBN.

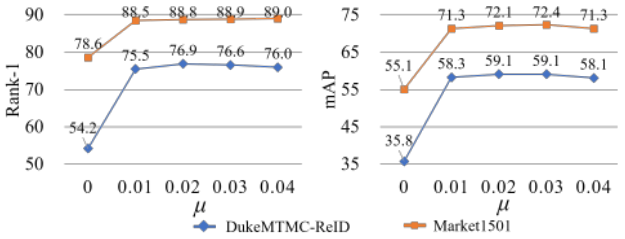

Figure 4. Evaluation of parameter μ in Eq. (5).

To further test the generalization of the network with AIBN, we train the network with labels on each dataset and test it directly on another dataset without fine-tuning. The results are shown in Table 3. On Market1501 and DukeMTMC-ReID, the AIBN improves the rank-1 accuracy by 5.6% and 11.3% for the direct transfer task, respectively. It is clear that AIBN can improve the generalization of the network.

Hyper-parameter Analysis. We investigate some important hyper-parameters in this section. Fig. 4 shows effects of parameter μ in Eq. (5). We can see that, as μ increases from 0 to 0.02, rank-1 accuracy on Market1501 and DukeMTMC-ReID increases from 78.6% and 55.1% to 88.8% and 72.1%, respectively. This shows that larger u brings considerable performance gains. It is also clear that μ is easy to tune, i.e., $\mu > 0.01$ leads to similar performance on different datasets.

We also conduct several experiments to verify the impacts of replacing BN with AIBN in different layers of the network and different weight sharing methods of α in Eq. (12). The results are shown in Table 4.

Replacing BNs in the deep layer of the network brings more substantial performance gains than replacing BNs in the shallow layer of the network. It is clear that, replacing

Dataset		Market		Duke	
		mAP	Rank-1	mAP	Rank-1
Position	Not Replace	67.1	85.5	51.4	71.3
	All	72.1	88.7	58.7	77.1
	Layer 1+2	64.1	86.2	53.3	73.2
	Layer 4	72.1	88.5	57.5	76.1
	Layer 2+3+4	71.3	88.7	**59.8**	**77.4**
	Layer 3+4	**72.1**	**88.8**	59.1	76.9
Sharing	Not sharing	**72.4**	**88.9**	58.4	76.2
	BottleNeck	72.1	88.8	**59.1**	**76.9**
	Layer	71.0	88.3	58.3	76.4

Table 4. Ablation study on inserted layer of AIBN and weight sharing methods of α in Eq. (12).

BNs with AIBN in Layer4 brings more significant performance gains than the replacements in Layer1 and Layer2. Since replacing BNs of Layer3 and Layer4 gives slightly better results, this setting is used in our experiments. We evaluate three settings of weight sharing methods of α: (a) Each AIBN has its own α; (b) AIBNs in the same BottleNeck module share the same α; (c) AIBNs in the same Layer of ResNet-50 share the same α. The results show that different weight sharing methods of α has limited impacts on the model performance.

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

We compare our method with recent unsupervised and transfer learning methods on Market1501 [41], DukeMTMC-ReID [23] and MSMT17 [31]. Table 5 and Table 6 summarize the comparison.

We first compare our method with methods trained with only unlabeled data. Compared methods include hand-craft features based methods, and deep learning based methods. It can be seen from Table 5 that compared with other deep learning based methods, our method surpasses these methods by a large margin. This significant improvement is mainly thanks to the more reliable similarity between samples used in clustering.

We also compare with the unsupervised domain adaptation methods, including GAN based methods (PTGAN [31], *etc.*), Distribution alignment based methods (TJ-AIDL [29], *etc.*), and Pseudo-labels based methods (MAR [33], *etc.*). Pseudo-labels based methods perform better than other types of methods in most cases. Many transfer learning methods use extra labeled source domain data for training. Our method still outperforms them using only unlabeled data for training. The performance of our method can be further improved by using the re-ranking similarity [42] instead of the cosine similarity. Note that, re-ranking similarity is a commonly used similarity in unsupervised ReID [6, 13] and is only used during training. Therefore, it only increases the training time and has no effect on the network inference time and online ReID time.

Methods	Reference	Market1501					DukeMTMC-ReID				
		Source	mAP	Rank-1	Rank-5	Rank-10	Source	mAP	Rank-1	Rank-5	Rank-10
PTGAN [31]	CVPR18	Duke	-	38.6	-	66.1	Market	-	27.4	-	50.7
HHL [43]	ECCV18	Duke	31.4	62.2	78.8	84.0	Market	27.2	46.9	61.0	66.7
DG-Net++ [49]	ECCV20	Duke	61.7	82.1	90.2	92.7	Market	63.8	78.9	87.8	90.4
TJ-AIDL [29]	CVPR18	Duke	26.5	58.2	74.8	81.8	Market	23.0	44.3	59.6	65.0
MMFA [15]	BMVC18	Duke	27.4	56.7	75.0	81.8	Market	24.7	45.3	59.8	66.3
CSCL [32]	ICCV19	Duke	35.6	64.7	80.2	85.6	Market	30.5	51.5	66.7	71.7
MAR [33]	CVPR19	MSMT17	40.0	67.7	81.9	-	MSMT17	48.0	67.1	79.8	-
AD-Cluster [35]	CVPR20	Duke	68.3	86.7	94.4	96.5	Market	54.1	72.6	82.5	85.5
NRMT [40]	ECCV20	Duke	71.7	87.8	94.6	96.5	Market	62.2	77.8	86.9	89.5
MMT-500 [7]	ICLR20	Duke	71.2	87.7	94.9	96.9	Market	63.1	76.8	88.0	**92.2**
MEB-Net* [36]	ECCV20	Duke	71.9	87.5	95.2	96.8	Market	63.5	77.2	87.9	91.3
LOMO [14]	CVPR15	None	8.0	27.2	41.6	49.1	None	4.8	12.3	21.3	26.6
BOW [41]	ICCV15	None	14.8	35.8	52.4	60.3	None	8.3	17.1	28.8	34.9
BUC [16]	AAAI19	None	29.6	61.9	73.5	78.2	None	22.1	40.4	52.5	58.2
HCT [34]	CVPR20	None	56.4	80.0	91.6	95.2	None	50.7	69.6	83.4	87.4
MMCL [28]	CVPR20	None	45.5	80.3	89.4	92.3	None	40.2	65.2	75.9	80.0
JVTC+ [13]	ECCV20	None	47.5	79.5	89.2	91.9	None	50.7	74.6	82.9	85.3
IICS†	This paper	None	72.1	88.8	95.3	96.9	None	59.1	76.9	86.1	89.8
IICS‡	This paper	None	**72.9**	**89.5**	**95.2**	**97.0**	None	**64.4**	**80.0**	**89.0**	91.6

Table 5. Performance comparison with recent methods on Market1501 and DukeMTMC-ReID. IICS denotes our method. †denotes using the cosine similarity to compute the CNN features similarity. ‡denotes using the re-ranking similarity [42] to replace the cosine similarly. * denotes the same backbone ResNet-50 is used in MEB-Net.

Methods	Source	MSMT17			
		mAP	Rank-1	Rank-5	Rank-10
PTGAN [31]	Market	2.9	10.2	-	24.4
ECN [44]	Market	8.5	25.3	36.3	42.1
SSG [6]	Market	13.2	31.6	-	49.6
NRMT [40]	Market	19.8	43.7	56.5	62.2
DG-Net++ [49]	Market	22.1	48.4	60.9	66.1
MMT-1500 [7]	Market	22.9	49.2	63.1	68.8
PTGAN [31]	Duke	3.3	11.8	-	27.4
ECN [44]	Duke	10.2	30.2	41.5	46.8
SSG [6]	Duke	13.3	32.2	-	51.2
NRMT [40]	Duke	20.6	45.2	57.8	63.3
DG-Net++ [49]	Duke	22.1	48.8	60.9	65.9
MMT-1500 [7]	Duke	23.3	50.1	63.9	69.8
MMCL [28]	None	11.2	35.4	44.8	49.8
JVTC+ [13]	None	17.3	43.1	53.8	59.4
IICS†	None	18.6	45.7	57.7	62.8
IICS‡	None	**26.9**	**56.4**	**68.8**	**73.4**

Table 6. Performance comparison with recent methods on MSMT17 [31]. IICS denotes our method. †denotes using the cosine similarity to compute the CNN feature similarity. ‡denotes using the re-ranking similarity [42] to replace the cosine similarity.

To further verify the effectiveness of our algorithm, we conduct experiments on a larger and more challenging dataset MSMT17. Our method outperforms existing methods under both unsupervised and unsupervised transfer settings by a large margin. We achieve the rank-1 accuracy of 56.4%, about 11% higher than the recent NRMT [40], which adopts extra DukeMTMC-ReID for training. Those above experiments clearly demonstrate the superior performance of the proposed method.

Discussion Our method uses pre-defined clustering numbers in both stages, thus the clustering number is a critical parameter for pseudo label generation. The clustering number can also be adaptively determined by setting a similarity threshold. Generalizable strategies for determining the clustering number for different datasets will be studied in our future work.

5. Conclusion

This paper proposes a intra-inter camera similarity method for unsupervised person ReID which iteratively optimizes Intra-Inter Camera similarity through generating intra- and inter-camera pseudo-labels. The intra-camera training stage is proposed to train a multi-branch CNN using generated intra-camera pseudo-labels. Based on the classification score produced by each classifier trained at intra-camera training stage, a more robust inter-camera similarity can be calculated. Then the network can be trained with the pseudo-label generated by performing clustering across cameras with this inter-camera similarity. Moreover, AIBN is introduced to boost the generalization ability of the network. Extensive experimental results demonstrate the effectiveness of the proposed method in unsupervised person ReID.

Acknowledgement This work is supported in part by Peng Cheng Laboratory, in part by Natural Science Foundation of China under Grant No. U20B2052, 61936011, 61620106009, in part by The National Key Research and Development Program of China under Grant No. 2018YFE0118400, in part by Beijing Natural Science Foundation under Grant No. JQ18012.

References

[1] Woong-Gi Chang, Tackgeun You, Seonguk Seo, Suha Kwak, and Bohyung Han. Domain-specific batch normalization for unsupervised domain adaptation. In *CVPR*, 2019. 3

[2] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *CVPR*, 2009. 5

[3] Weijian Deng, Liang Zheng, Qixiang Ye, Guoliang Kang, Yi Yang, and Jianbin Jiao. Image-image domain adaptation with preserved self-similarity and domain-dissimilarity for person re-identification. In *CVPR*, 2018. 1

[4] Qi Dou, Daniel Coelho de Castro, Konstantinos Kamnitsas, and Ben Glocker. Domain generalization via model-agnostic learning of semantic features. In *NeurIPS*, 2019. 2, 4

[5] Hehe Fan, Liang Zheng, Chenggang Yan, and Yi Yang. Unsupervised person re-identification: Clustering and fine-tuning. *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)*, 14(4):1–18, 2018. 1, 3

[6] Yang Fu, Yunchao Wei, Guanshuo Wang, Yuqian Zhou, Honghui Shi, and Thomas S Huang. Self-similarity grouping: A simple unsupervised cross domain adaptation approach for person re-identification. In *ICCV*, 2019. 1, 7, 8

[7] Yixiao Ge, Dapeng Chen, and Hongsheng Li. Mutual mean-teaching: Pseudo label refinery for unsupervised domain adaptation on person re-identification. In *ICLR*, 2020. 2, 3, 8

[8] Arthur Gretton, Kenji Fukumizu, Zaid Harchaoui, and Bharath K Sriperumbudur. A fast, consistent kernel two-sample test. In *NeurIPS*, 2009. 2

[9] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016. 5

[10] Alexander Hermans, Lucas Beyer, and Bastian Leibe. In defense of the triplet loss for person re-identification. *arXiv preprint arXiv:1703.07737*, 2017. 5

[11] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *ICML*, 2015. 3, 5

[12] Xin Jin, Cuiling Lan, Wenjun Zeng, and Zhibo Chen. Global distance-distributions separation for unsupervised person re-identification. In *ECCV*, 2020. 1, 3

[13] Jianing Li and Shiliang Zhang. Joint visual and temporal consistency for unsupervised domain adaptive person re-identification. In *ECCV*, 2020. 7, 8

[14] Shengcai Liao, Yang Hu, Xiangyu Zhu, and Stan Z Li. Person re-identification by local maximal occurrence representation and metric learning. In *CVPR*, 2015. 8

[15] Shan Lin, Haoliang Li, Chang-Tsun Li, and Alex Chichung Kot. Multi-task mid-level feature alignment network for unsupervised cross-dataset person re-identification. In *BMVC*, 2018. 1, 2, 8

[16] Yutian Lin, Xuanyi Dong, Liang Zheng, Yan Yan, and Yi Yang. A bottom-up clustering approach to unsupervised person re-identification. In *AAAI*, 2019. 1, 3, 8

[17] Xiaobin Liu and Shiliang Zhang. Domain adaptive person re-identification via coupling optimization. In *Proceedings of the 28th ACM International Conference on Multimedia*, pages 547–555, 2020. 3

[18] Xiaobin Liu, Shiliang Zhang, and Ming Yang. Self-guided hash coding for large-scale person re-identification. In *MIPR*, pages 246–251. IEEE, 2019. 1

[19] H. Luo, W. Jiang, Y. Gu, F. Liu, X. Liao, S. Lai, and J. Gu. A strong baseline and batch normalization neck for deep person re-identification. *IEEE Transactions on Multimedia*, 22(10):2597–2609, 2020. 1, 6

[20] Laurens van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of machine learning research*, 9(Nov):2579–2605, 2008. 1

[21] Xingang Pan, Ping Luo, Jianping Shi, and Xiaoou Tang. Two at once: Enhancing learning and generalization capacities via ibn-net. In *ECCV*, 2018. 3, 6, 7

[22] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011. 6

[23] Ergys Ristani, Francesco Solera, Roger Zou, Rita Cucchiara, and Carlo Tomasi. Performance measures and a data set for multi-target, multi-camera tracking. In *ECCV*, 2016. 1, 2, 5, 7

[24] Baochen Sun and Kate Saenko. Deep coral: Correlation alignment for deep domain adaptation. In *ECCV*, 2016. 2

[25] Yifan Sun, Liang Zheng, Yi Yang, Qi Tian, and Shengjin Wang. Beyond part models: Person retrieval with refined part pooling (and a strong convolutional baseline). In *ECCV*, 2018. 1

[26] Eric Tzeng, Judy Hoffman, Trevor Darrell, and Kate Saenko. Simultaneous deep transfer across domains and tasks. In *ICCV*, 2015. 2, 4

[27] Dmitry Ulyanov, Andrea Vedaldi, and Victor Lempitsky. Improved texture networks: Maximizing quality and diversity in feed-forward stylization and texture synthesis. In *CVPR*, 2017. 3, 5

[28] Dongkai Wang and Shiliang Zhang. Unsupervised person re-identification via multi-label classification. In *CVPR*, 2020. 1, 3, 8

[29] Jingya Wang, Xiatian Zhu, Shaogang Gong, and Wei Li. Transferable joint attribute-identity deep learning for unsupervised person re-identification. In *CVPR*, 2018. 1, 7, 8

[30] Longhui Wei, Xiaobin Liu, Jianing Li, and Shiliang Zhang. Vp-reid: Vehicle and person re-identification system. In *ICMR*, ICMR '18, page 501–504, New York, NY, USA, 2018. Association for Computing Machinery. 1

[31] Longhui Wei, Shiliang Zhang, Wen Gao, and Qi Tian. Person transfer gan to bridge domain gap for person re-identification. In *CVPR*, 2018. 1, 2, 5, 7, 8

[32] Ancong Wu, Wei-Shi Zheng, and Jian-Huang Lai. Unsupervised person re-identification by camera-aware similarity consistency learning. In *ICCV*, 2019. 1, 2, 8

[33] Hong-Xing Yu, Wei-Shi Zheng, Ancong Wu, Xiaowei Guo, Shaogang Gong, and Jian-Huang Lai. Unsupervised person re-identification by soft multilabel learning. In *CVPR*, 2019. 1, 7, 8

[34] Kaiwei Zeng, Munan Ning, Yaohua Wang, and Yang Guo. Hierarchical clustering with hard-batch triplet loss for person re-identification. In *CVPR*, 2020. 8

[35] Yunpeng Zhai, Shijian Lu, Qixiang Ye, Xuebo Shan, Jie Chen, Rongrong Ji, and Yonghong Tian. Ad-cluster: Augmented discriminative clustering for domain adaptive person re-identification. In *CVPR*, 2020. 1, 3, 8

[36] Yunpeng Zhai, Qixiang Ye, Shijian Lu, Mengxi Jia, Rongrong Ji, and Yonghong Tian. Multiple expert brainstorming for domain adaptive person re-identification. In *ECCV*, 2020. 1, 3, 8

[37] Xinyu Zhang, Jiewei Cao, Chunhua Shen, and Mingyu You. Self-training with progressive augmentation for unsupervised cross-domain person re-identification. In *ICCV*, 2019. 3

[38] Xuan Zhang, Hao Luo, Xing Fan, Weilai Xiang, Yixiao Sun, Qiqi Xiao, Wei Jiang, Chi Zhang, and Jian Sun. Alignedreid: Surpassing human-level performance in person re-identification. *arXiv preprint arXiv:1711.08184*, 2017. 1

[39] Ying Zhang, Tao Xiang, Timothy M Hospedales, and Huchuan Lu. Deep mutual learning. In *CVPR*, 2018. 3

[40] Fang Zhao, Shengcai Liao, Guo-Sen Xie, Jian Zhao, Kaihao Zhang, and Ling Shao. Unsupervised domain adaptation with noise resistible mutual-training for person re-identification. In *ECCV*, 2020. 2, 3, 8

[41] Liang Zheng, Liyue Shen, Lu Tian, Shengjin Wang, Jingdong Wang, and Qi Tian. Scalable person re-identification: A benchmark. In *ICCV*, 2015. 1, 2, 5, 7, 8

[42] Zhun Zhong, Liang Zheng, Donglin Cao, and Shaozi Li. Re-ranking person re-identification with k-reciprocal encoding. In *CVPR*, 2017. 7, 8

[43] Zhun Zhong, Liang Zheng, Shaozi Li, and Yi Yang. Generalizing a person retrieval model hetero-and homogeneously. In *ECCV*, 2018. 8

[44] Zhun Zhong, Liang Zheng, Zhiming Luo, Shaozi Li, and Yi Yang. Invariance matters: Exemplar memory for domain adaptive person re-identification. In *CVPR*, 2019. 8

[45] Zhun Zhong, Liang Zheng, Zhedong Zheng, Shaozi Li, and Yi Yang. Camera style adaptation for person re-identification. In *CVPR*, 2018. 1

[46] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. Unpaired image-to-image translation using cycle-consistent adversarial networkss. In *ICCV*, 2017. 2

[47] Xiangping Zhu, Xiatian Zhu, Minxian Li, Pietro Morerio, Vittorio Murino, and Shaogang Gong. Intra-camera supervised person re-identification. *arXiv preprint arXiv:2002.05046*, 2021. 3

[48] Zijie Zhuang, Longhui Wei, Lingxi Xie, Tianyu Zhang, Hengheng Zhang, Haozhe Wu, Haizhou Ai, and Qi Tian. Rethinking the distribution gap of person re-identification with camera-based batch normalization. In *ECCV*, 2020. 3

[49] Yang Zou, Xiaodong Yang, Zhiding Yu, BVK Kumar, and Jan Kautz. Joint disentangling and adaptation for cross-domain person re-identification. In *ECCV*, 2020. 1, 2, 8