Cluster Contrast for Unsupervised Person Re-Identification

Zuozhuo Dai [1*] Guangyuan Wang[1*] Weihao Yuan[1] Siyu Zhu[1] Ping Tan[12]
[1]Alibaba A.I. Labs [2]Simon Fraser University

Abstract

Unsupervised person re-identification (re-ID) attracts increasing attention due to its practical applications in industry. State-of-the-art unsupervised re-ID methods train the neural networks using a memory-based non-parametric softmax loss. They store the pre-computed instance feature vectors inside the memory, assign pseudo labels to them using clustering algorithm, and compare the query instances to the cluster using a form of contrastive loss. During training, the instance feature vectors are updated. However, due to the varying cluster size, the updating progress for each cluster is inconsistent. To solve this problem, we present Cluster Contrast which stores feature vectors and computes contrast loss in the cluster level. We demonstrate that the inconsistency problem for cluster feature representation can be solved by the cluster-level memory dictionary. By straightforwardly applying Cluster Contrast to a standard unsupervised re-ID pipeline, it achieves considerable improvements of 9.5%, 7.5%, 6.6% compared to state-of-the-art purely unsupervised re-ID methods and 5.1%, 4.0%, 6.5% mAP compared to the state-of-the-art unsupervised domain adaptation re-ID methods on the Market, Duke, and MSMT17 datasets. Code is available at https://github.com/alibaba/cluster-contrast.

1. Introduction

Deep unsupervised person Re-identification (re-ID) aims to train a neural network capable of retrieving a person of interest across cameras without any labeled data. This task attracts increasing attention recently due to the growing demands in practical video surveillance and the expensive labeling cost. There are mainly two approaches to address this problem. One is the purely unsupervised learning (USL) person re-ID, which generally exploits pseudo labels from the completely unlabeled data [8, 9, 11, 23, 39]. The other is the unsupervised domain adaptation person re-ID (UDA), which first pre-trains a model on the source labeled dataset, and then fine-tunes the model on the target unlabeled dataset [6, 22, 40, 42, 48, 55, 56]. Generally, the performance of UDA is superior to that of USL because of the introduction of the external source domain. However, UDA still suffers from the complex training procedure and requires that the difference between the source and target domain is not significant.

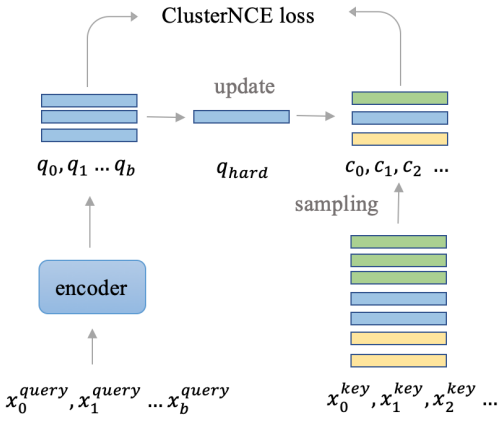


Figure 1: The Cluster Contrast computes the contrastive loss in cluster level. In the cluster level memory dictionary, the cluster feature is initialized through uniformly sample a random instance feature from the corresponding cluster and updated by the batch hard query instance. $x \in X$ is the training dataset. q is the query instance feature vector. c_k stands for the k-th cluster feature vector. Feature vectors with the same color belong to the same cluster.

In this paper, we focus on learning the person re-ID task without any labeled data (USL). Existing state-of-the-art USL methods [11, 39] exploit the memory dictionary together with pseudo labels from the clustering operation to train the neural network. At the beginning of each epoch, all the image features of the training data are extracted by the current neural network and we regard these image features stored in memory as the dictionary. Then, a clustering algorithm, like DBScan [7] or K-means [28] is employed to cluster image features and produce pseudo labels. Meanwhile, the cluster ID is assigned to each image as the person identity. Finally, the neural network is trained with a contrastive loss such as triplet loss [17, 33], InfoNCE loss [29], or other non-parametric classification loss [39] based on the memory dictionary. Although these pseudo-label-based methods achieve remarkable performance, there is still a large gap between USL and supervised methods, which limits its USL in practical applications.

To bridge the gap, we re-investigate the pseudo-label-based USL pipeline and argue that the memory dictionary

*Equal contribution.

storage mechanism, including the initialization, updating, and loss computation, is crucial to model optimization. The state-of-the-art approaches store the feature vector of every instance inside the memory dictionary and compute the multi-label non-parametric classification loss [39] or InfoNCE loss [29] with the query instance. Such instance-level contrastive loss functions have two main drawbacks. First, the feature updating progress of memory dictionary is inconsistent, since the distribution of training data is biased in the instance level. This problem is especially serious in large-scale re-ID datasets like MSMT17 [42]. Second, the cluster centroid is not an optimal feature representation of pseudo-labels in memory dictionary. Since the clustering approach inevitably introduces noisy labels, such an averaging operation of features within a cluster would contaminate its corresponding cluster feature representation as well.

To solve these two problems, we propose the Cluster Contrast. It builds a cluster-level memory dictionary in which each cluster is represented by a single feature vector and all cluster feature vectors are consequently updated in a consistent manner. More specifically, this cluster feature is initialized as the feature of an image randomly chosen from the cluster. During training, the cluster feature is updated by the batch hard query instance feature [17], which selects the most dissimilar query instance to cluster feature inside one mini-batch. Accordingly, we propose a cluster-level InfoNCE loss, denoted as ClusterNCE loss, based on the cluster-level memory dictionary, which takes much less GPU memory than the instance-level feature memory, and consequently allows our method to be trained on a large dataset as the same as InfoNCE loss. Since it is very expensive to cluster the whole dataset, for each epoch we only sample a small set of the original dataset for training and update this small set on the fly. Experiments demonstrate that our sampling-based mechanism can reach similar performance to the training with the full dataset.

In the experimental section, we demonstrate that the proposed purely unsupervised approach with Cluster Contrast surpasses all the existing unsupervised person re-ID methods by a large margin. Specifically, it achieves considerable improvements of 9.5%, 7.5%, 6.6% compared to the state-of-the-art purely unsupervised re-ID methods and 5.1%, 4.0%, 6.5% mAP compared to the state-of-the-art unsupervised domain adaptation re-ID methods on the Market, Duke, and MSMT17 datasets.

2. Related Work

Deep Unsupervised Person Re-ID. Deep unsupervised person re-ID can be summarized into two categories. The first category is pure unsupervised learning (USL) person re-ID [8, 9, 11, 23, 39], which trains model directory on unlabeled dataset. The second category is unsupervised domain adaptation (USA) re-ID, which utilizes transfer learning to improve unsupervised person re-ID [56, 55, 6, 42, 48, 40, 22]. State-of-the-art USL re-ID pipeline generally involves three stages: memory dictionary initialization, pseudo label generation, and neural network training. Previous works made great improvements on part or the whole pipeline. Specifically, Lin *et al*. [23] treat each individual sample as a cluster, and then gradually groups similar samples into one cluster to generate pseudo labels. MMCL [39] predicts quality pseudo labels comprising similarity computation and cycle consistency. It then trains the model as a multi-classification problem. SPCL [11] proposes a novel self-paced contrastive learning framework that gradually creates more reliable cluster to refine the hybrid memory dictionary containing both source-domain and target-domain dataset features. In this paper, we focus on USL re-ID method and the proposed Cluster Contrast aims to improve the memory dictionary and loss function part. Next we discuss related studies with respect to these two aspects.

Memory Dictionary. Contrast learning [13] can be thought of as training an encoder for a dictionary look-up task. Several recent studies [1, 16, 14, 18, 29, 36, 45, 58] on unsupervised visual representation learning present promising results through building dynamic dictionaries. The memory dictionary is updated consistently on the fly to facilitate contrastive unsupervised learning. Similar to unsupervised visual representation learning, state-of-the-art unsupervised person re-ID methods also build memory dictionaries for contrastive learning [46, 39, 10, 11]. During training, instance feature vectors in memory dictionary are updated by the features of query instances in the same cluster. Because the number of instances in every cluster is imbalanced, the updating progress of each cluster is inconsistent correspondingly, e.g. the cluster with fewer instances updates faster than the cluster with more instances. We propose a cluster-level memory dictionary which solves this problem by storing one single cluster feature vector instead of all instance feature vectors. Thus cluster features could be updated with a consistent speed.

Loss Functions. Another problem that comes to us is how to represent the salient cluster feature in a robust manner. In supervised person re-ID, the combination of triplet loss and identity loss is proved to be effective solutions to improving the re-ID performance [34, 38, 3, 2, 57, 51, 4, 12, 25]. An essential part of learning using the triplet loss is the hard example mining, as the easy examples with large instance number can overwhelm training and lead to degenerate models. To solve this problem, distance weighted sampling [43] selects more informative and stable examples for training and batch hard triplet loss [17] is presented to mine the hardest triplet in each batch. In unsupervised person re-ID, since there is no ground truth person identity and the pseudo labels are changing during training, people then use non-parametric classification loss such as In-

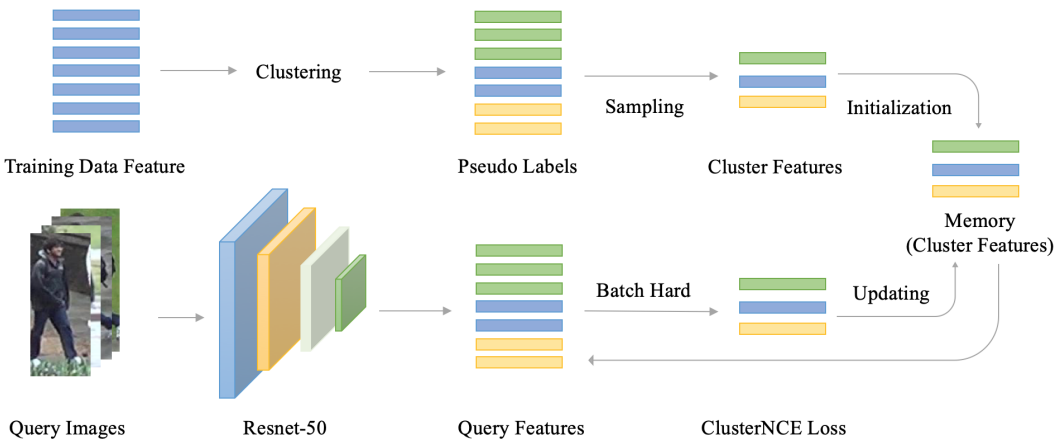


Figure 2: The unsupervised person re-ID pipeline. Feature vectors with the same color belong to the same cluster. The upper part is the memory initialization stage. Training data features are assigned pseudo labels by clustering algorithm. The lower part is the model training stage. Hard exampling method is used to select the hard query instance to update memory feature. The ClusterNCE loss computer contrastive loss between query features and all cluster features.

foNCE [29] as identity loss. Similar to InfoNCE, Tong *et al*. [46] designs an Online Instance Matching (OIM) loss with a memory dictionary scheme which compares query image to a memorized feature set of unlabelled identities. Wang and Zhang [39] introduce the memory-based non-parametric multi-label classification loss (MMCL), which treat USL re-ID as a multi-label classification problem. It uses hard negative class mining to solve the imbalanced class problem. In order to mitigate noisy pseudo labels, MMT [10] proposes the soft pseudo labels and a a novel soft softmax-triplet loss to support learning with soft pseudo triplet labels. SPCL [11] introduces a unified contrastive loss including both source domain dataset and target domain dataset. When SPCL is used in USL scenario, it stores each instance feature in dictionary and average them as the cluster feature for loss computation. However, the averaging operation would be very likely to introduce noisy labels due to the pseudo label generation. Inspired by the hard example mining [17], we select the hardest query feature vector from the current mini-batch to update the cluster feature, which is proved to be more informative than simply averaging all the instances within the same cluster.

3. Methodology

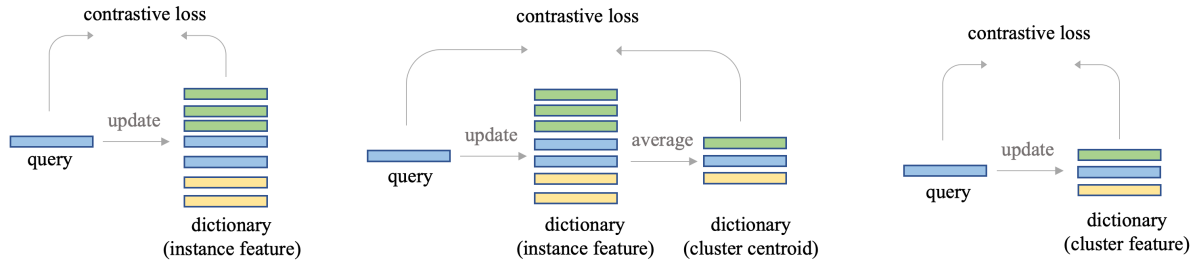
3.1. USL Person Re-ID as Contrastive Learning

State-of-the-art USL methods [39, 10, 11] generally follow the following procedures. First, ImageNet is applied to pre-train the neural network to extract feature vectors. Then, such approaches use a clustering algorithm, such as DBScan [7] or K-means [28], to generate pseudo labels. Finally, a form of contrastive loss is used to compute the loss values between the query instances and the memory dictionary. In Figure 2, the baseline pipeline together with the ClusterNCE loss is demonstrated. Specifically, we use a standard Resnet50 [15] as the backbone to extract semantic features of the input images. Similar to [11], we use DBScan to cluster training data features. DBScan algorithm requires two hyper-parameters, the maximum distance ϵ between two samples for one to be considered as in the neighborhood of the other, and the minimum number of samples $min\_samples$ in a neighborhood for an instance to be considered as a core instance. We set ϵ to 0.4 and min_sample to 4 in all our experiments. Then, The cluster ID is assigned to each training image as the pseudo label and the unclustered outlier images are discarded from training for simplicity.

In Figure 3, we compare the non-parametric loss functions of different approaches based on the memory dictionary. As shown in Figure 3 (a), the Memory Based Multi-classification Loss (MMCL) [39] computes the loss and updates the memory dictionary both in the instance level. In Figure 3 (b), SPCL [11] computes the loss in cluster level but updates the memory dictionary in the instance level. Figure 3 (c) demonstrates the proposed ClusterNCE loss which updates the feature vectors and computes the loss both in the cluster level. As elaborated in the following section, the contrastive loss design is vital to USL person re-ID methods and the proposed ClusterNCE loss outperforms the other two loss functions in the task of USL person re-ID.

3.2. Cluster Contrast

In this section, we demonstrate how to apply the Cluster Contrast in the existing USL pipeline. To demonstrate the effectiveness of our method, we keep the pipeline simple and follow previous methods as much as possible. As illustrated in Figure 2, the cluster feature is initialized through sampling a random instance feature from the correspond-



(a) multi-label classification loss (b) instance level InfoNCE loss (c) ClusterNCE loss

Figure 3: The comparison to existing memory based non-parametric classification loss.

ing cluster. During the model training stage, the query image features are compared to all cluster features with ClusterNCE loss. The training details are presented in Algorithm 1.

Require: Unlabeled training data X;
Require: Initialize the backbone encoder f_θ with ImageNet-pretrained ResNet-50;
Require: Temperature τ for Eq. 3;
Require: Momentum m for Eq. 2;
for n *in* $[1, num\_epochs]$ **do**
  Extract feature vectors X^{key} from X by f_θ;
  Clustering X^{key} into N clusters with DBScan;
  Initialize memory dictionary with Eq. 1;
  for i *in* $[1, num\_iterations]$ **do**
    Sample $P \times K$ query images from X;
    Computer ClusterNCE loss with Eq. 3;
    Update cluster feature with Eq. 2;
  end
end
Algorithm 1: USL pipeline with Cluster Contrast

We can see that the proposed Cluster Contrast involves the process of memory dictionary initialization, memory updating, and neural network training. In the following sections, we discuss about the three process in detail.

Memory Initialization. We store each cluster's feature $\{c_1, \cdots, c_N\}$ in the memory-based feature dictionary. Here, N stands for the number of clusters. Note that the clustering algorithm runs in every epoch so N is changing as the model trains. We use the feature of a random instance in the cluster to initialize the cluster feature, that is

$$c_i \leftarrow U(X_i) \tag{1}$$

where $U(\cdot)$ is a uniform sampling function and X_i denotes the i-th cluster set that contains all the instance feature vectors in cluster i.

Memory Updating. During training, we sample P person identities and a fixed number K of instances for each person identity. Consequently, we obtain a total number of $P \times K$ query images in the mini batch [17]. In contrast to the previous instance-level feature memory methods [10, 11], which just update all $P \times K$ query instance features in the instance level memory, we select the hardest instance for each person identity to momentum update the cluster feature as illustrated in Figure 3. Specifically, we select P hardest instance feature vectors from $P \times K$ query instances and update corresponding cluster feature vectors. For a certain cluster with person identity i, its feature vector is updated by:

$$\begin{aligned} q_{hard} &\leftarrow \arg\min_q q \cdot c_i, q \in Q^i \\ c_i &\leftarrow m \cdot c_i + (1-m) \cdot q_{hard}. \end{aligned} \tag{2}$$

Where, the batch hard instance q_{hard} is the instance with the minimum similarity to the cluster feature c^i. We measure the similarity with dot product. m is the momentum updating factor. Q^i is the instance features set with cluster id i in current mini batch.

Loss Function. Given a query instances q, we compare it to all the cluster features C in our cluster level feature memory using InfoNCE loss [29]:

$$L_q = -\log \frac{exp(q \cdot c^+)/\tau}{\sum_{i=0}^{K} exp(q, c_i)/\tau} \tag{3}$$

where c^+ is the positive cluster feature vector to query instance q and τ is a temperature hyper-parameter per [45]. The loss value is low when q is similar to its positive cluster feature c^+ and dissimilar to all other cluster features. It is a log loss of K-way softmax-based classifier that tries to classify q as c^+

Discussion. The Cluster Contrast has two main differences compared with the previous memory-based methods [11, 39]. First, it represents each cluster as a single feature vector. Second, it updates the cluster feature vector using the batch hard query instance feature vector.

We argue that through representing each cluster as a single feature vector, we can speed up feature updating and

Dataset	Object	#train IDs	#train images	#test IDs	#query images	#total images	#cameras
MSMT17	Person	1,041	32,621	3,060	11,659	126,441	15
PersonX	Person	410	9,840	856	5,136	45,792	6
Market-1501	Person	751	1 2,936	750	3,368	32,668	6
DukeMTMC-reID	Person	702	16,522	702	2,228	36,441	8
VeRi-776	Vehicle	575	37,746	200	1,678	51,003	20

Table 1: Statistics of datasets used in the paper.

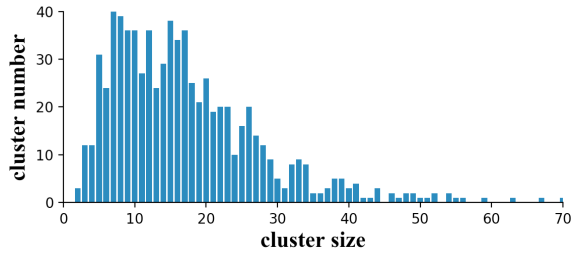


Figure 4: The cluster size follows a normal distribution in Market1501 dataset.

eliminate the feature inconsistency in the memory dictionary. As shown in Figure 4, the cluster size follows a normal distribution with a mean size of 15. Most of the cluster has size ranged from 5 to 30, and there are still many clusters have a large size up to 70. As a result, in instance-level memory, the updating process for each cluster is slow and be varied up to $10\times$. In a typical 8G memory GPU card, the batch size can be set to $16 \times 16 = 256$ at most, and only 50% instance feature vectors could be updated in an iteration. A natural solution is to limit the cluster size so that all instances can be updated, which indeed solves the problem as shown Section 4.3. For simplicity, we find that a single feature vector is enough for cluster representation, and it could also decrease the GPU memory usage and training time.

We can see that both the multi-label classification loss [39] shown in Figure 3 (a) and instance-level InfoNCE loss [11] shown in Figure 3 (b) calculate the averaged classification loss between query feature vector and each instance feature vector for each cluster. Simply averaging would always introduce noisy feature vectors for cluster feature representation because the pseudo labels inevitably contain noise. Besides, the averaged feature represents the most common feature inside one cluster, lacking the hard example mining for query instances. However, select the hardest pair in one cluster would also introduce outliers. Therefore, similar to batch hard triplet loss [17], we select the hardest query instance inside one batch to update the cluster feature vector. The batch hard instance can be considered a moderate instance since it is the hardest within a small subset of the data rather than the hardest in the whole cluster. We show that the batch hard instance performs much better than averaging instance features in experiment section 4.3.

4. Experiment

4.1. Datasets and Implementation

Datasets We evaluate our proposed method on four large-scale person re-identification (re-ID) benchmarks: Market-1501 [52], DukeMTMC-reID [32], MSMT17 [42], PersonX [35], and one vehicle ReID datasets, VeRi-776 [26]. The Market-1501, DukeMTMC-reID, and MSMT17 are widely used real-world person re-identification tasks. The PersonX is synthesized based on Unity [31], which contains manually designed obstacles such as random occlusion, resolution, and lighting differences. To show the robustness of our method, we also conduct vehicle re-identification experiments on the widely used real scene VeRi-776 datasets. The details of these datasets are summarized in Table 1.

Implementation Details We adopt ResNet-50 [15] as the backbone of the feature extractor and initialize the model with the parameters pre-trained on ImageNet [5]. After layer-4, we remove all sub-module layers and add global average pooling (GAP) followed by batch normalization layer [19] and L2-normalization layer, which will produce 2048-dimensional features. During testing, we take the features of the global average pooling layer to calculate the distance. For the beginning of each epoch, we use DBSCAN [7] for clustering to generate pseudo labels.

The input image is resized 256 x 128 for Market-1501 and PersonX datasets, and 224 x 224 for MSMT17 and VeRi-776. For training images, we perform random horizontal flipping, padding with 10 pixels, random cropping, and random erasing [54]. Each mini-batch contains 256 images of 16 pseudo person identities (16 instances for each person). We adopt Adam optimizer to train the re-ID model with weight decay 5e-4. The initial learning rate is set to 3.5e-4, and is reduced to 1/10 of its previous value every 20 epoch in a total of 50 epoch. As with the cluster method of [11] paper, we use DBSCAN and Jaccard distance [53] to cluster with k nearest neighbors, where k = 30. For DBSCAN, the maximum distance d between two samples is set as 0.4 and the minimal number of neighbors in a core point is set as 4.

4.2. Comparison with State-of-the-arts

Compared with SOTA USL Methods. We first compare our method to State-of-the-arts USL methods which is the main focus of our method. From Table 2, we can

Methods	Market-1501				
	source	mAP	top-1	top-5	top-10
BUC [23]	None	38.3	66.2	79.6	84.5
SSL [24]	None	37.8	71.7	83.8	87.4
MMCL [39]	None	45.5	80.3	89.4	92.3
MMCL [39]	Duke	60.4	84.4	92.8	95.0
HCT [49]	None	56.4	80.0	91.6	95.2
CycAs [41]	None	64.8	84.8	-	-
AD-Cluster++ [50]	Duke	68.3	86.7	94.4	96.5
UGA [44]	None	70.3	87.2	-	-
MMT [10]	MSMT17	75.6	89.3	95.8	97.5
SPCL [11]	None	73.1	88.1	95.1	97.0
SPCL [11]	MSMT17	77.5	89.7	96.1	97.6
Ours/ResNet-50	None	**82.6**	**93.0**	**97.0**	**98.1**
Ours/IBN-ResNet-50	None	**84.1**	**93.2**	**97.6**	**98.2**

(a) Experiments on Market-1501 datasets

Methods	DukeMTMC-reID				
	source	mAP	top-1	top-5	top-10
BUC [23]	None	27.5	47.4	62.6	68.4
SSL [24]	None	28.6	52.5	63.5	68.9
MMCL [47]	None	51.4	72.4	82.9	85.0
AD-Cluster++ [50]	Market	54.1	72.6	82.5	85.5
MMCL [39]	Market	51.4	72.4	82.9	85.0
HCT [49]	None	50.7	69.6	83.4	87.4
UGA [44]	None	53.3	75.0	-	-
CycAs [41]	None	60.1	77.9	-	-
SPCL [11]	None	65.3	81.2	90.3	92.2
MMT [10]	Market	65.1	78.9	88.8	92.5
SPCL [11]	Market	68.8	82.9	90.1	92.5
Ours/ResNet-50	None	**72.8**	**85.7**	**92.0**	**93.5**
Ours/IBN-ResNet-50	None	**74.2**	**85.8**	**92.1**	**94.2**

(b) Experiments on DukeMTMC-reID datasets

Methods	MSMT17				
	source	mAP	top-1	top-5	top-10
ECN [56]	Duke	10.2	30.2	41.5	46.8
MMCL [47]	None	11.2	35.4	44.8	49.8
TAUDL [20]	None	12.5	28.4	-	-
UTAL [21]	None	13.1	31.4	-	-
SPCL [11]	None	19.1	42.3	55.6	61.2
UGA [44]	None	21.7	49.5	-	-
MMT [10]	Market	24.0	50.1	63.5	69.3
CycAs [41]	None	26.7	50.1	-	-
SPCL [11]	Market	26.8	53.7	65.0	69.8
Ours/ResNet-50	None	**33.3**	**63.3**	**73.7**	**77.8**
Ours/IBN-ResNet-50	None	**41.1**	**69.1**	**79.3**	**83.1**

(c) Experiments on MSMT17 datasets

Methods	PersonX				
	source	mAP	top-1	top-5	top-10
SPCL [11]	None	72.3	88.1	96.6	98.3
MMT [10]	Market	78.9	90.6	96.8	98.2
SPCL [11]	Market	78.5	91.1	97.8	99.0
Ours/ResNet-50	None	**84.8**	**94.5**	**98.4**	**99.2**
Ours/IBN-ResNet-50	None	**87.1**	**95.6**	**98.9**	**99.5**

(d) Experiments on PersonX datasets

Methods	VeRi-776				
	source	mAP	top-1	top-5	top-10
SPCL [11]	None	36.9	79.9	86.8	89.9
MMT [10]	VehicleID	35.3	74.6	82.6	87.0
SPCL [11]	VehicleID	38.9	80.4	86.8	89.6
Ours/ResNet-50	None	**42.5**	**87.7**	**91.4**	**93.1**
Ours/IBN-ResNet-50	None	**43.6**	**89.8**	**92.7**	**94.4**

(e) Experiments on VeRi-776 datasets

Table 2: Comparison with state-of-the-art methods on the object re-ID, including unsupervised methods and unsupervised domain adaptation methods. "None" represents the pure unsupervised method. Other value represents the source-domain dataset in unsupervised domain adaptation method.

Batch size	#Instance	mAP	Rank-1
64	4	65.5	82.7
128	8	71.5	86.7
256	16	73.1	87.3
416	26	**75.1**	**89.0**

Table 3: The impact of instance number to baseline method on Market1501 dataset.

Cluster Size	#Instance	Fraction	mAP	Rank-1
20	4	0.2	67.6	84.4
8	4	0.5	71.4	85.9
5	4	0.8	73.2	87.5
4	4	1.0	**76.5**	**89.0**

Table 4: The impact of cluster size to baseline method on Market1501 dataset.

Labels	#Cluster	mAP	Rank-1
Ground Truth	751	85.1	93.8
Splitting	851	83.3	93.6
Merging	651	78.2	91.0

Table 5: The impact of splitting and merging incorrect instances to baseline method on Market1501 dataset.

see that our method is significantly better than all existing unsupervised methods, which proves the effectiveness of our method. Based on the same pipeline, the mAP of our method surpasses the state-of-the-art USL method SPCL [11] by 9.5%, 6.6%, 7.7%, 12.5% and 5.6% on Market-1501 [52], MSMT17 [42], DukeMTMC-reID [32], PersonX [35], and VeRi-776 [26] respectively. Some qualitative results are shown in Figure 5.

Compared with SOTA UDA Methods. We also compare our method with State-of-the-arts UDA re-ID methods. The UDA methods can make full use of the labeled source domain datasets, so they usually achieve better results than the USL re-ID methods. The ClusterNCE loss could also be easily generalized to UDA re-ID methods. However, since it does not focus on how to make use of the labeled source domain dataset, the labeled source dataset does not help much. Table 2 show that the mAP of our pure unsupervised re-ID method still outperforms the SOTA UDA method [11] by up to 10% even though they use more training data.

Compared with IBN-ResNet-50 and ResNet-50 backbones. Instance-batch normalization (IBN) [30] combines the advantages of IN [37] learning appearance invari-

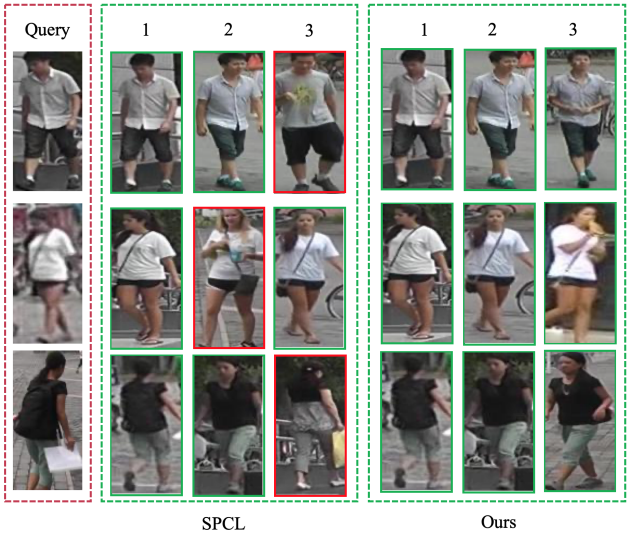


Figure 5: The comparison of top-3 ranking list between SPCL [11] and our method on Market-1501 dataset. The correct results are highlighted by green borders and the incorrect results by red borders.

ance features and BN [19] learning content-related information. It has been proved effective in object re-ID methods in supervised [27] learning tasks. We evaluate our method with IBN-ResNet as the backbone of our framework, which is formed by replacing all BN layers in ResNet-50 with IBN layers. As shown in Table 6, using IBN-ResNet can further improve performance.

4.3. Ablation Studies

Int this section, we study the effectiveness of various component in Clusetr Contrast method. We define the USL pipeline with instance-level meomry dictionary (Figure 3 (b)) as the **baseline method**.

Memory Updating Consistency In section 3.2, we argue that compared to instance-level memory, the cluster-level memory could update cluster feature more consistently. As shown in Figure 3 (b), the instance-level memory dictionary maintains the feature of each instance of the dataset. In every training iteration, each instance feature in the mini-batch will be updated to its own memory dictionary. Since the cluster size is unbalancedly distributed, only a small fraction of the instance features could be updated in a large cluster when all instances in a small cluster are updated. There are two natural solutions to increase the fraction of the updated instance features.

The simplest solution is to increase the batch size. As the batch size increases, more instance features could be updated inside one cluster. We conduct the batch size experiments on the baseline and present the results in Table 3. It is shown that the performance of the baseline increases as the batch size increases. However, the batch size reaches its upper limit of 512 due to the GPU memory.

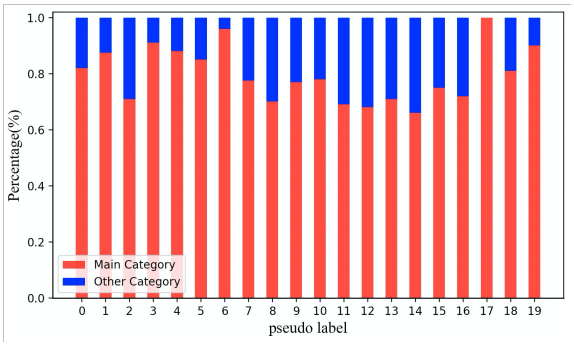


Figure 6: We randomly select 20 categories from the Market1501 clustering results and calculate the percentage of different categories using ground truth labels.

To deal with the limitation of the GPU memory, we restrict the cluster size. Specifically, we fix the cluster size to a constant number so in every iteration a fixed fraction of the instance features could be updated. In this way, the instance feature vectors can be updated consistently with a small batch size. The results in Table 4 demonstrate that the performance of the baseline increases with the rising of the fraction of the updated instance features, until it reaches the peak performance when the cluster size is the instance number K so that all instance feature vectors inside one cluster could be updated in a single iteration. In sum, we propose the Cluster Contrast, which uses a single feature vector to represent the cluster so it can be updated in every iteration.

Update policy	Market-1501			
	mAP	top-1	top-5	top-10
Average	78.7	90.9	96.0	96.5
Random	80.9	91.7	96.7	97.5
Easy	75.8	89.7	95.2	96.9
Hard	**82.6**	**93.0**	**97.0**	**98.1**

Table 6: Comparison of different cluster feature updating polices on our method.

Cluster Feature Representation As shown in Figure 3 (b), the instance-level memory averages all instance feature vectors to represent the cluster feature. However, in USL re-ID, the pseudo label generation stage would inevitably introduce the outlier instances. In Figure 6, we count the proportions of different real categories being clustered into the same category on the Market-1501 dataset. It shows there still around 20% noisy instances when model training in finished. These incorrect instance feature vectors are averaged along with correct instance feature vectors to compute the cluster feature centroid, which harmful to the cluster representation. As Ge *et al*. [11] states that merging an instance to the wrong cluster does more harm than good, we study the impact of merging and splitting wrong instances on the

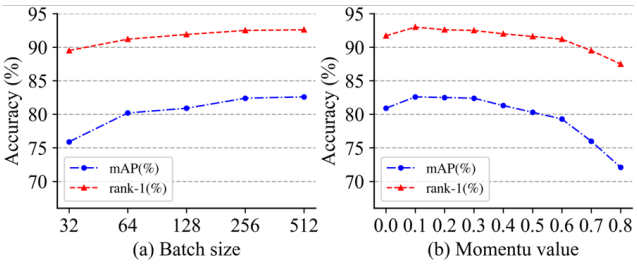


Figure 7: The impact of (a) batch size and (b) momentum value on Market-1501 datasets

baseline method. In Table 5 we randomly merge two ground truth clusters or split one cluster to two clusters. It shows that comparing to using ground-truth label, merging two different cluster degrade more performance than splitting one cluster. In the batch hard strategy, if the correct instance is selected, updating cluster feature with the instance feature would no double improve the final performance. However, if the incorrect instance is selected in minority cases, then there are two clusters selecting the instances which actually have the same ground truth label. This is exactly the splitting case in Table 5. It is still better than always using the incorrect instance to compute the cluster centroid which is the merging case.

In Table 6, we evaluate different cluster feature updating designs. "Average" means we average the query instance vectors in the current batch to update the corresponding cluster feature. "Random" means we randomly select a query instance and "Easy" means we select the most similar query instance to cluster feature. "Hard" is the batch hard method used in our method. It shows the batch hard example works best for person re-ID.

Momentum Values. We use the momentum update strategy to update cluster features in memory dictionary. As shown in Equation 2, the momentum value m controls the update speed of cluster memory. The larger the value of m, the slower the cluster memory update. We conducted experiments on the Market-1501 dataset to explore the influence of different m values on our method. As shown in Figure 7 (b), smaller m (less than 0.3) performs better than large m (greater than 0.7). Therefore, we can conclude that faster cluster memory updating(i.e., relatively small momentum) benefits re-ID.

Batch Size. To explore the impact of different batch sizes on our method, we use different batch sizes from 32 to 512 to train our method. From the experimental results shown in Figure 7 (a), we can see that a large batch size will achieve better results. Comparing to the baseline method in Table 3, the performance of our method remains stable over a wide range of batch sizes from 64 to 512.

d	Market-1501		Veri-776	
	mAP	top-1	mAP	top-1
0.3	78.1	90.6	25.6	58.9
0.4	**82.6**	**93.0**	36.8	78.0
0.5	81.3	92.2	39.4	82.5
0.6	80.9	92.0	39.7	83.3
0.7	77.9	90.6	**42.5**	**87.8**
0.8	67.2	84.8	38.9	80.0

Table 7: Performances of our method with different of d, where d represents the maximum distance between two sample in DBSCAN algorithm.

Sampe rate	Veri-776			
	mAP	top-1	top-5	top-10
0.3	37.2	82.5	88.7	91.0
0.5	40.4	84.7	89.6	92.0
1.0	42.5	87.8	91.4	94.4

Table 8: In the large scale Veri776 dataset, we sample a fraction of dataset and study the impact of sampling rate on final performance.

Impact of Clustering Hyper-parameters. The maximum distance d between two samples is a hyper-parameter of the DBSCAN algorithm, which will affect the final number of clusters. If the d value is chosen too small then a larger part of the data will be considered as outliers. If it is chosen too large then the clusters will merge and a majority of the data points will be in the same clusters. Eventually, it will affect the performance of our algorithm. We analyze the influence of d on performance on the Market-1501 and Veri-776 datasets. From Table 7, we can see that the value of d has a relatively large impact on the results, and the values of d for the optimal results are different under different datasets.

Training on Large Datasets. Similar to InfoNCE loss, the ClusterNCE loss could be also used in sampling-based contrastive learning. For large-scale datasets like Veri776, it is too expansive to clustering every instance in every epoch. Therefore, we sample a fraction of the dataset and do clustering at the beginning of each epoch. Table 8 show that our method could still achieve good performance when only use 30% of training data.

5. Conclusion

In this paper, we present the Cluster Contrast mechanism, which stores feature vectors and computes contrast loss in cluster level memory dictionary. It solves two problems exists in the instance level memory dictionary. First, it unifies the cluster feature updating progress regardless the cluster size or dataset size. Second, it uses a more robust cluster feature representation to compute the contrastive loss. Experiments show that the simple USL pipeline with Cluster Contrast surpassing all existing USL and UDA re-ID methods.

References

[1] Philip Bachman, R Devon Hjelm, and William Buchwalter. Learning representations by maximizing mutual information across views. *arXiv preprint arXiv:1906.00910*, 2019. 2

[2] Guangyi Chen, Chunze Lin, Liangliang Ren, Jiwen Lu, and Jie Zhou. Self-critical attention learning for person re-identification. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9637–9646, 2019. 2

[3] Zuozhuo Dai, Mingqiang Chen, Xiaodong Gu, Siyu Zhu, and Ping Tan. Batch dropblock network for person re-identification and beyond. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3691–3701, 2019. 2

[4] Zuozhuo Dai, Mingqiang Chen, Siyu Zhu, and Ping Tan. Batch feature erasing for person re-identification and beyond. *arXiv preprint arXiv:1811.07130*, 1(2):3, 2018. 2

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

[6] 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 *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 994–1003, 2018. 1, 2

[7] Martin Ester, Hans-Peter Kriegel, Jörg Sander, Xiaowei Xu, et al. A density-based algorithm for discovering clusters in large spatial databases with noise. In *Kdd*, volume 96, pages 226–231, 1996. 1, 3, 5

[8] 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, 2

[9] 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 *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6112–6121, 2019. 1, 2

[10] Yixiao Ge, Dapeng Chen, and Hongsheng Li. Mutual mean-teaching: Pseudo label refinery for unsupervised domain adaptation on person re-identification. *arXiv preprint arXiv:2001.01526*, 2020. 2, 3, 4, 6

[11] Yixiao Ge, Feng Zhu, Dapeng Chen, Rui Zhao, and hongsheng Li. Self-paced contrastive learning with hybrid memory for domain adaptive object re-id. In H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 11309–11321. Curran Associates, Inc., 2020. 1, 2, 3, 4, 5, 6, 7

[12] Jianyuan Guo, Yuhui Yuan, Lang Huang, Chao Zhang, Jin-Ge Yao, and Kai Han. Beyond human parts: Dual part-aligned representations for person re-identification. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3642–3651, 2019. 2

[13] Raia Hadsell, Sumit Chopra, and Yann LeCun. Dimensionality reduction by learning an invariant mapping. In *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*, volume 2, pages 1735–1742. IEEE, 2006. 2

[14] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9729–9738, 2020. 2

[15] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 3, 5

[16] Olivier Henaff. Data-efficient image recognition with contrastive predictive coding. In *International Conference on Machine Learning*, pages 4182–4192. PMLR, 2020. 2

[17] Alexander Hermans, Lucas Beyer, and Bastian Leibe. In defense of the triplet loss for person re-identification. *arXiv preprint arXiv:1703.07737*, 2017. 1, 2, 3, 4, 5

[18] R Devon Hjelm, Alex Fedorov, Samuel Lavoie-Marchildon, Karan Grewal, Phil Bachman, Adam Trischler, and Yoshua Bengio. Learning deep representations by mutual information estimation and maximization. *arXiv preprint arXiv:1808.06670*, 2018. 2

[19] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *International conference on machine learning*, pages 448–456. PMLR, 2015. 5, 7

[20] Minxian Li, Xiatian Zhu, and Shaogang Gong. Unsupervised person re-identification by deep learning tracklet association. In *Proceedings of the European conference on computer vision (ECCV)*, pages 737–753, 2018. 6

[21] Minxian Li, Xiatian Zhu, and Shaogang Gong. Unsupervised tracklet person re-identification. *IEEE transactions on pattern analysis and machine intelligence*, 42(7):1770–1782, 2019. 6

[22] 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. *arXiv preprint arXiv:1807.01440*, 2018. 1, 2

[23] Yutian Lin, Xuanyi Dong, Liang Zheng, Yan Yan, and Yi Yang. A bottom-up clustering approach to unsupervised person re-identification. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 8738–8745, 2019. 1, 2, 6

[24] Yutian Lin, Lingxi Xie, Yu Wu, Chenggang Yan, and Qi Tian. Unsupervised person re-identification via softened similarity learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3390–3399, 2020. 6

[25] Jinxian Liu, Bingbing Ni, Yichao Yan, Peng Zhou, Shuo Cheng, and Jianguo Hu. Pose transferrable person re-identification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4099–4108, 2018. 2

[26] Xinchen Liu, Wu Liu, Huadong Ma, and Huiyuan Fu. Large-scale vehicle re-identification in urban surveillance videos. In *2016 IEEE International Conference on Multimedia and Expo (ICME)*, pages 1–6. IEEE, 2016. 5, 6

[27] Hao Luo, Youzhi Gu, Xingyu Liao, Shenqi Lai, and Wei Jiang. Bag of tricks and a strong baseline for deep person re-identification. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pages 0–0, 2019. 7

[28] James MacQueen et al. Some methods for classification and analysis of multivariate observations. In *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability*, volume 1, pages 281–297. Oakland, CA, USA, 1967. 1, 3

[29] Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018. 1, 2, 3, 4

[30] Xingang Pan, Ping Luo, Jianping Shi, and Xiaoou Tang. Two at once: Enhancing learning and generalization capacities via ibn-net. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 464–479, 2018. 6

[31] John Riccitiello. John riccitiello sets out to identify the engine of growth for unity technologies (interview). *VentureBeat. Interview with Dean Takahashi. Retrieved January*, 18(3), 2015. 5

[32] Ergys Ristani, Francesco Solera, Roger Zou, Rita Cucchiara, and Carlo Tomasi. Performance measures and a data set for multi-target, multi-camera tracking. In *European conference on computer vision*, pages 17–35. Springer, 2016. 5, 6

[33] Florian Schroff, Dmitry Kalenichenko, and James Philbin. Facenet: A unified embedding for face recognition and clustering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 815–823, 2015. 1

[34] Jifei Song, Yongxin Yang, Yi-Zhe Song, Tao Xiang, and Timothy M Hospedales. Generalizable person re-identification by domain-invariant mapping network. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 719–728, 2019. 2

[35] Xiaoxiao Sun and Liang Zheng. Dissecting person re-identification from the viewpoint of viewpoint. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 608–617, 2019. 5, 6

[36] Yonglong Tian, Dilip Krishnan, and Phillip Isola. Contrastive multiview coding. *arXiv preprint arXiv:1906.05849*, 2019. 2

[37] Dmitry Ulyanov, Andrea Vedaldi, and Victor Lempitsky. Improved texture networks: Maximizing quality and diversity in feed-forward stylization and texture synthesis. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6924–6932, 2017. 6

[38] Cheng Wang, Qian Zhang, Chang Huang, Wenyu Liu, and Xinggang Wang. Mancs: A multi-task attentional network with curriculum sampling for person re-identification. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 365–381, 2018. 2

[39] Dongkai Wang and Shiliang Zhang. Unsupervised person re-identification via multi-label classification. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10981–10990, 2020. 1, 2, 3, 4, 5, 6

[40] Jingya Wang, Xiatian Zhu, Shaogang Gong, and Wei Li. Transferable joint attribute-identity deep learning for unsupervised person re-identification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2275–2284, 2018. 1, 2

[41] Zhongdao Wang, Jingwei Zhang, Liang Zheng, Yixuan Liu, Yifan Sun, Yali Li, and Shengjin Wang. Cycas: Self-supervised cycle association for learning re-identifiable descriptions. *arXiv preprint arXiv:2007.07577*, 2020. 6

[42] Longhui Wei, Shiliang Zhang, Wen Gao, and Qi Tian. Person transfer gan to bridge domain gap for person re-identification. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 79–88, 2018. 1, 2, 5, 6

[43] Chao-Yuan Wu, R Manmatha, Alexander J Smola, and Philipp Krahenbuhl. Sampling matters in deep embedding learning. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2840–2848, 2017. 2

[44] Jinlin Wu, Yang Yang, Hao Liu, Shengcai Liao, Zhen Lei, and Stan Z Li. Unsupervised graph association for person re-identification. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8321–8330, 2019. 6

[45] Zhirong Wu, Yuanjun Xiong, Stella X Yu, and Dahua Lin. Unsupervised feature learning via non-parametric instance discrimination. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3733–3742, 2018. 2, 4

[46] Tong Xiao, Shuang Li, Bochao Wang, Liang Lin, and Xiaogang Wang. Joint detection and identification feature learning for person search. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017. 2, 3

[47] Tong Xiao, Shuang Li, Bochao Wang, Liang Lin, and Xiaogang Wang. Joint detection and identification feature learning for person search. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3415–3424, 2017. 6

[48] 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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2148–2157, 2019. 1, 2

[49] Kaiwei Zeng, Munan Ning, Yaohua Wang, and Yang Guo. Hierarchical clustering with hard-batch triplet loss for person re-identification. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13657–13665, 2020. 6

[50] 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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9021–9030, 2020. 6

[51] Zhizheng Zhang, Cuiling Lan, Wenjun Zeng, and Zhibo Chen. Densely semantically aligned person re-identification.

In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 667–676, 2019. 2

[52] Liang Zheng, Liyue Shen, Lu Tian, Shengjin Wang, Jingdong Wang, and Qi Tian. Scalable person re-identification: A benchmark. In *Proceedings of the IEEE international conference on computer vision*, pages 1116–1124, 2015. 5, 6

[53] Zhun Zhong, Liang Zheng, Donglin Cao, and Shaozi Li. Re-ranking person re-identification with k-reciprocal encoding. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1318–1327, 2017. 5

[54] Zhun Zhong, Liang Zheng, Guoliang Kang, Shaozi Li, and Yi Yang. Random erasing data augmentation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 13001–13008, 2020. 5

[55] Zhun Zhong, Liang Zheng, Shaozi Li, and Yi Yang. Generalizing a person retrieval model hetero-and homogeneously. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 172–188, 2018. 1, 2

[56] Zhun Zhong, Liang Zheng, Zhiming Luo, Shaozi Li, and Yi Yang. Invariance matters: Exemplar memory for domain adaptive person re-identification. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 598–607, 2019. 1, 2, 6

[57] Sanping Zhou, Fei Wang, Zeyi Huang, and Jinjun Wang. Discriminative feature learning with consistent attention regularization for person re-identification. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8040–8049, 2019. 2

[58] Chengxu Zhuang, Alex Lin Zhai, and Daniel Yamins. Local aggregation for unsupervised learning of visual embeddings. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6002–6012, 2019. 2