Unsupervised Domain Adaptation in Person re-ID via k-Reciprocal Clustering and Large-Scale Heterogeneous Environment Synthesis

Devinder Kumar[1] Parthipan Siva[2] Paul Marchwica[2] Alexander Wong[1]
[1]University of Waterloo [2]Sportlogiq
{devinder.kumar,a28wong}@uwaterloo.ca, {parthipan,paul}@sportlogiq.com

Abstract

An ongoing major challenge in computer vision is the task of person re-identification, where the goal is to match individuals across different, non-overlapping camera views. While recent success has been achieved via supervised learning using deep neural networks, such methods have limited widespread adoption due to the need for large-scale, customized data annotation. As such, there has been a recent focus on unsupervised learning approaches to mitigate the data annotation issue; however, current approaches in literature have limited performance compared to supervised learning approaches as well as limited applicability for adoption in new environments. In this paper, we address the aforementioned challenges faced in person re-identification for real-world, practical scenarios by introducing a novel, unsupervised domain adaptation approach for person re-identification. This is accomplished through the introduction of: i) k-reciprocal tracklet Clustering for Unsupervised Domain Adaptation (ktCUDA) (for pseudo-label generation on target domain), and ii) Synthesized Heterogeneous RE-id Domain (SHRED) composed of large-scale heterogeneous independent source environments (for improving robustness and adaptability to a wide diversity of target environments). Experimental results across four different image and video benchmark datasets show that the proposed ktCUDA and SHRED approach achieves an average improvement of +5.7 mAP in re-identification performance when compared to existing state-of-the-art methods, as well as demonstrate better adaptability to different types of environments.

1. Introduction

Person re-identification (re-ID) attempts to match an individual from one camera view across other, non-overlapping camera views [15]. The most successful methods [45, 50, 21] leverage deep learning via a supervised learning approach. Such supervised learning driven approaches assume the availability of a large, manually-labelled dataset of individuals across multiple cameras in the deployment environment (referred as the target domain). This assumption inherently limits the widespread adoption of person re-ID because of the cost and logistics needed for manually annotating data from the target domain, which is not practical in many real-world scenarios.

Figure 1. Iterative adaptation to unlabelled target domain using the proposed ktCUDA approach. Result on test set after each iteration of adaptation on the unlabelled training set. Starting with direct knowledge transfer from the proposed SHRED source domain on the first row. Query on the left and the top-5 search result with green for correct match and blue for incorrect match. Image from Market-1501 [52] (left) and DukeMTMC-reID [16] (right).

To overcome the reliance on a large, manually-labelled dataset from the target domain, two approaches have been proposed in recent literature: a pure unsupervised approach [27], and the more popular unsupervised domain adaptation approach [2, 61, 11, 31, 49]. Both approaches rely on an unlabelled dataset from the target domain which is easily obtained by running tracking on the target domain. Furthermore, the unsupervised domain adaptation approach assumes the availability of a manually-labelled dataset from an independent source domain [11, 31, 49], whereas the pure unsupervised approach does not require a manually-labelled source domain dataset.

Without an independent source domain, the pure unsupervised approaches cannot function at all in a new target domain until they have learned the new environment. From

a practical point of view, this is undesirable as the system is not able to function at all upon deployment. The unsupervised domain adaption methods on the other hand are pre-trained on an independent source domain and can function upon deployment by directly transferring models learned on the source domain (we refer to this as direct transfer). Starting from the direct transfer results, the system simply gets better as it adapts to the target domain (Fig. 1). The ability for immediate usage upon deployment makes such an unsupervised domain transfer approach very attractive from a practical point of view, but only if direct transfer performance is good and unsupervised domain adaptation can further improve the performance of the system.

There are two key limitations to existing unsupervised domain adaptation approaches [2, 61, 11, 31, 49]. The first limitation is that the domain adaptation component of existing approaches either: i) only considers environmental style transfer between source and target domains [2, 61] and do not explicitly learn suitable features and distance metric for the target domain, or ii) directly transfer distance metric (typically Euclidean distance) [11, 31, 49] learned on the source domain to target domain for obtaining pseudo-labels on the target domain. Pseudo-labels are then used to learn suitable features and distance metric on the target domain. However, direct transfer of distance metric is not optimal due to differences in the source domain environment and target domain environment.

The second limitation of existing unsupervised domain adaptation approaches is that they rely heavily on a limited real-world source domain. Typically, a single independent environment is used as a source domain [27, 42, 38, 13, 60] which doesn't capture enough variations in environments needed for domain adaptation. Some methods have attempted to augment the source domain with thousands of synthetic data with varying illuminations [2], but other environmental variations outside of illumination are not captured. Finally, there are few works [49, 2, 34] that combine few different datasets in the source domain to obtain some variability. However, their performance before and after adaptation is generally noticeably lower as compared to the latest unsupervised person re-ID techniques [27].

In this work, we address the two aforementioned limitations of the current domain adaptation methods. First, we explore how to better leverage distance metrics that have been learned on the source domain to the target domain. Recently, k-reciprocal re-ranking [58] has become a popular post-processing step for all supervised re-ID methods, where the k-reciprocal nearest neighbours are ranked higher than neighbours that minimize a distance metric and result in better performance. It was shown in [58] to boost performance by $\sim 10\%$ on the mean average precision (mAP). Motivated by the effectiveness of such an approach within the realm of supervised re-ID, we propose a k-reciprocal tracklet Clustering method for Unsupervised Domain Adaptation (ktCUDA), where k-reciprocal neighbours are used to assign pseudo-labels to the target domain.

Second, we investigate the construction of a source domain that captures large environmental variations i.e., large number of identities and environmental conditions to ensure the best results for direct transfer of source domain to target domain. To this end, we constructed the Synthesized Heterogeneous RE-id Domain (SHRED), the largest source domain used in domain adaptation person re-ID literature. We show that the proposed SHRED performs very well for the direct transfer scenario. When combined with the proposed ktCUDA, we show that state-of-the-art performance can be achieved for unsupervised domain transfer on several test datasets.

The main contributions of this paper are:

- **ktCUDA**, a novel k-reciprocal tracklet clustering algorithm for obtaining unsupervised pseudo-labels on the target domain.
- **SHRED**, a synthesized large-scale heterogeneous source domain that captures a wide set of environmental variations.
- A comprehensive analysis using both image and video datasets to show the performance of the proposed ktCUDA and SHRED, with full experimental results for direct transfer of knowledge from source domain to target domain as well as experimental results after domain adaptation.

2. Related Works

Unsupervised domain adaptation can take the form of environmental style (such as illumination) transfer between source and target domains [2, 61, 60] or iterative clustering and training based on distance metric transfer (typically Euclidean distance) [11, 31, 49] between source and target domain. Our approach is an iterative clustering approach similar to [11, 31, 49]. However, unlike [11, 49], which uses distance metric directly for clustering, we use k-reciprocal neighbours. The concept of using k-reciprocal neighbours in clustering for domain adaptation has been used in [31]. But in [31], k-reciprocal neighbours are used to threshold potential cluster candidates then Euclidean distance is used during clustering. On the contrary, the proposed ktCUDA approach leverages k-reciprocal neighbour distance to perform spectral clustering without the reliance on distance metric during clustering.

3. Methodology

A common approach to unsupervised domain adaptation for person re-ID is to predict pseudo-labels for the unlabelled target domain using a deep convolutional neural network (DCNN) trained on the source domain and then fine-

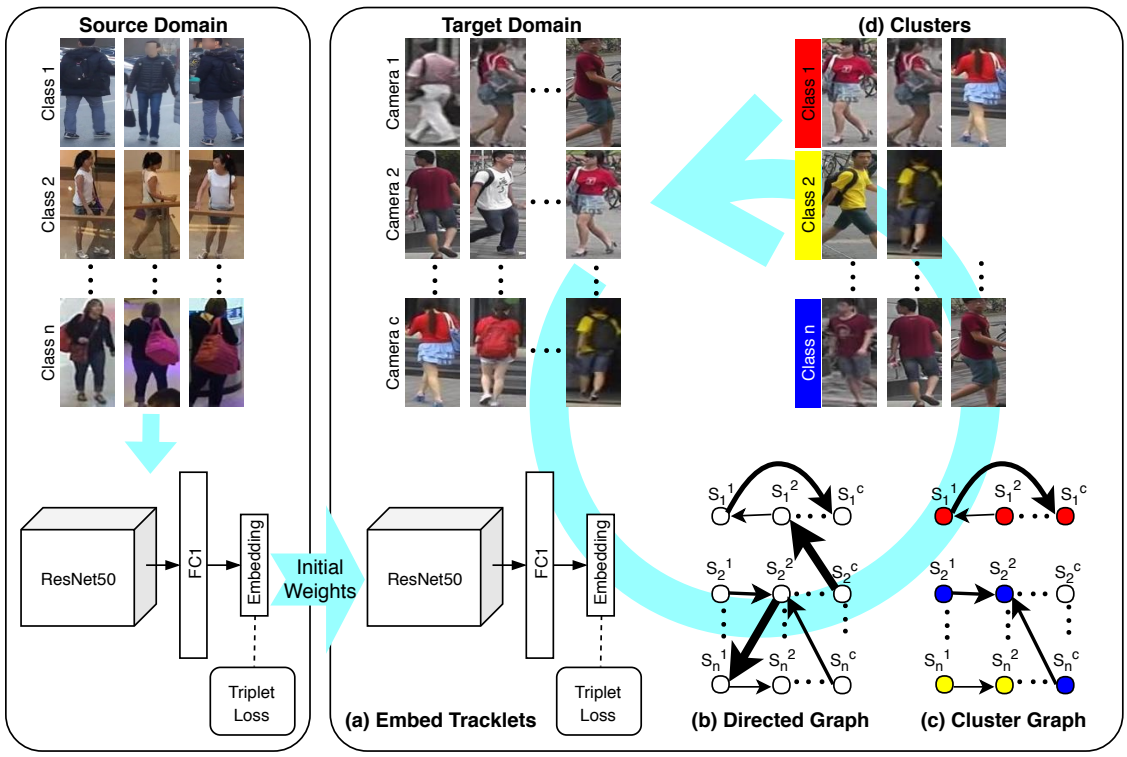

Figure 2. Overview of the proposed k-reciprocal tracklet Clustering for Unsupervised Domain Adaptation (ktCUDA) in person re-ID. Given the proposed Synthesized Heterogeneous RE-id Domain (SHRED) as source domain, a ResNet-50 model with fully connected and embedding layers (DCNN) is trained with triplet loss. Once trained, the weights are used to initialize an iterative training on an unlabelled target domain. (a) DCNN is used to embed target domain tracklets to an embedding space. (b) Tracklet embeddings are used to form a directed graph, with each node representing a tracklet and each weighted connection representing how much the two tracklets belong to the same cluster. (c) The directed graph is then thresholded based on weight to form clusters. (d) Tracklet images in each cluster formed in (c) are used to fine-tune the DCNN. This process (a)–(d) is repeated for I iterations.

tune the DCNN for the target domain using the pseudo-labels [11, 49]. Typically, pseudo-labels are obtained by clustering [11, 49] using distance metrics on the samples in the target domain. Two problems with existing clustering based approaches [11, 49] are:

- The heavy reliance on distance metrics [11, 31] in the target domain using an embedding learned for the source domain. This results in poor clusters due to environmental differences between source and target domains making distance metrics unreliable between the domains.
- The use of a source domain with low environmental variability [27, 42, 49] or the reliance on synthetic environmental variability [2] result in a poor initial embedding for clustering.

We discuss our ktCUDA approach to overcome the strong reliance on distance metrics in Section 3.1, and our SHRED approach to obtain the best source domain in Section 3.2.

3.1. Iterative Domain Adaptation

Motivated to overcome the limitation of strong reliance on distance metrics in existing approaches [11, 49], we introduce a novel k-reciprocal tracklet Clustering for Unsupervised Domain Adaptation (ktCUDA). It has been shown in previous literature that leveraging k-reciprocal nearest neighbours to re-rank person re-ID search results, instead of the raw distances ranking, can result in a $\sim 10\%$ boost in performance [58, 57, 26]. Based on this observation, the proposed ktCUDA approach leverages the k value in k-reciprocal nearest neighbours as the cost for joining tracklets into one cluster. This results in more accurate and robust clusters than using the raw distance between the two tracklets for the same reasons that re-ranking results in better person re-ID performance.

Our ktCUDA approach is illustrated in Fig. 2. We iteratively fine-tune a DCNN on the unlabelled target domain by automatically obtaining labels using ktCUDA. More specifically, the following strategy was taken:

1. Transform the target domain tracklets to the embed-

ding space using the DCNN (Fig. 2(a))
2. Cluster the tracklets using our k-reciprocal tracklet clustering approach (Fig. 2(b-c))
3. Use the clusters as the unsupervised labels for the tracklets and fine-tune our DCNN (Fig. 2(d))
4. Repeat steps 1 to 3 for I iterations

3.1.1 k-Reciprocal Tracklet Clustering

A tracklet is a short sequence of a tracked person in the video. Following findings of [51], we represent the tracklet by the average embedding vector of the person bounding box on each frame of the tracklet. When we refer to a tracklet, we will be referring to the average embedding vector. The embedding is obtained by a DCNN; in our case, it is the same model as used in [18] – a ResNet-50 model with two additional fully connected layers as illustrated in Fig. 2.

The unlabelled target domain

$$\boldsymbol{\mathcal{S}} = \{\mathrm{S}^1, \ldots, \mathrm{S}^c, \ldots, \mathrm{S}^N\}$$

is the set of all tracklet S^c from N cameras in the target domain and

$$\mathrm{S}^c = \{s_1^c, \ldots, s_t^c, \ldots, s_n^c\}$$

is the set of n tracklet from camera c and s_t^c is the t^{th} tracklet from camera c.

Given $\boldsymbol{\mathcal{S}}$, the goal is to find clusters (i.e. subsets of $\boldsymbol{\mathcal{S}}$) that represent a unique individual across multiple camera views. Two tracklets s_t^c and s_i^j with a small Euclidean distance $||s_t^c - s_i^j||$ will tend to be the same person if the DCNN was trained on the target domain using triplet loss. In this case, the DCNN was not trained on the target domain.

A stronger argument is that if s_t^c and s_i^j are k-reciprocal neighbours of each other (for a small k value) then the two tracklets will represent the same unique person [36]. Leveraging the idea of k-reciprocal nearest neighbours, we define a directed graph $\mathcal{G}$ where the weighted edges $\mathcal{E}$ represent k-reciprocal distance, the cost of assigning two tracklets to the same cluster. Clusters can then be formed on $\mathcal{G}$ to select tracklets representing a unique person.

Graph Construction – We define the k_1-nearest neighbours (i.e. the top-k_1 list) of s_t^c as the closest tracklets in the target domain $\boldsymbol{\mathcal{S}}$ excluding tracklets from camera c (i.e. cross-camera closest tracklets using Euclidean distance):

$$\text{top}(k_1, s_t^c) \in \boldsymbol{\mathcal{S}} \setminus \mathrm{S}^c \tag{1}$$

Using (1), we construct a directed graph $\mathcal{G} = \{\mathcal{V}, \mathcal{E}\}$ where vertices ($\mathcal{V}$) of the directed graph are representative of all the tracklets in the target domain: $s_t^c \in \boldsymbol{\mathcal{S}}$ (i.e. $\mathcal{V} = \boldsymbol{\mathcal{S}}$). Directed graph edges $e(s_t^c, s_i^j) \in \mathcal{E}$ are created from vertex s_t^c to all $s_i^j \in \text{top}(k_1, s_t^c)$. That is, we have k_1 directed edges starting from node s_t^c to its k_1-nearest neighbours (Fig. 2(a) illustrates a graph where $k_1 = 1$). Each edge $e(s_t^c, s_i^j)$ is given a weight, which we define as k-reciprocal distance:

$$e(s_t^c, s_i^j) = k = \arg\min_k s_t^c \in top(k, s_i^j). \tag{2}$$

In other words, we define $e(s_t^c, s_i^j)$, the distance between s_t^c and s_i^j, as the minimum k at which s_t^c and s_i^j are $k-$reciprocal neighbours of each other. For example if s_i^j is the $1-$nearest neighbour of s_t^c and s_t^c is the $5-$nearest neighbour of s_i^j then $e(s_t^c, s_i^j) = 5$.

Graph Clustering – Given the graph $\mathcal{G}$ we form a new graph $\mathcal{G}'$ by cutting edge connections using threshold K:

$$e(s_t^c, s_i^j) = \begin{cases} e(s_t^c, s_i^j) & \text{if } e(s_t^c, s_i^j) \leq K \\ \emptyset & \text{if } e(s_t^c, s_i^j) > K \end{cases} \tag{3}$$

where $e(s_t^c, s_i^j) = \emptyset$ means the connection between s_t^c and s_i^j has been removed (Fig. 2(b) illustrates the graph $\mathcal{G}$ and Fig. 2(c) illustrates the corresponding sparse graph $\mathcal{G}'$).

Due to these removal of connections, graph $\mathcal{G}'$ is a sparsely connected graph with a set of connected subgraphs $g' \subset \mathcal{G}'$. We define the cardinality of the connected subgraph g' as:

$$|g'| = \text{number of vertices in } g' \tag{4}$$

From the sparse graph $\mathcal{G}'$ we create a valid cluster set $\mathcal{C}$ as the set of connected subgraphs with number of nodes (a.k.a tracklets) greater than T:

$$\mathcal{C} = \{g'_{i=0,\ldots,m}\} \quad \forall \quad \begin{matrix} g'_i \in \mathcal{G}' \\ |g'_i| > T - 0.4 \end{matrix} \tag{5}$$

3.1.2 DCNN Fine-Tuning

All images in the tracklets of a single cluster (i.e. subgraph g') from the cluster set (5) are used as a unique class for the DCNN fine-tuning (Fig. 2(d)). During fine-tuning, all the layers of DCNN are re-trained with unsupervised cluster data using batch hard triplet loss as per [18]. For re-training, the weights from the previous iteration of domain adaptation are used as initialization.

3.2. Large-scale Heterogeneous Environment Synthesis

While the iterative process described in Section 3.1 allows us to adapt a DCNN to a target domain, we still need initial DCNN weights to start with. Typically, an initial DCNN is trained on an independent source domain. The source domain can either be a single independent dataset [27, 42, 38, 13, 60], a synthetic dataset [2], or a combination of few independent datasets [49, 2, 34].

Table 1. Composition of proposed SHRED source domain variants

Dataset	SHRED 1	SHRED 2	SHRED 3	# IDs	# Images	# Cameras
3DPeS [3]	✓	✓	✓	164	951	8
Airport [22]	✓	✓	✓	1381	8660	6
CUHK02 [28]	✓	✓	✓	1816	7264	10
CUHK03 [30]	✓	✓		1467	14097	10
DukeMTMC-reID [56]	✓		✓	1404	32948	8
End-to-End [44]	✓	✓	✓	11934	34574	N/A
GRID [32]	✓	✓	✓	250	500	8
iLIDS-VID [41]	✓	✓	✓	300	42459	2
MSMT17 [42]	✓	✓	✓	3060	126142	15
VIPeR [17]	✓	✓	✓	632	1264	2
Market-1501* [52]		✓	✓	1501	32668	6

SHRED 1 is used to test on Market-1501, MARS and PRID datasets (22,408 IDs)
SHRED 2 is used to test on DukeMTMC-reID dataset (22,505 IDs)
SHRED 3 is used to test on CUHK03 dataset (22,442 IDS)

An important consideration to keep in mind when considering adaptation from source domain to target domain is that we need embedding learned on the source domain to be as invariant as possible to environmental conditions such as lighting, background, etc. As such using a single independent environment [27, 42, 38, 13, 60] in the source domain is not ideal because network thus obtained will be too specific to the source domain. The use of synthetic source domain [2] can achieve invariance but only to the variables introduced in the generation of the synthetic data. Ideally, we would want the source domain to be created with data from many different actual environments as possible. With the nearly 30 different source domains that has been used since 2007 for person re-ID research [1], it is possible to construct a source domain that has a wide variety of environmental variations.

Motivated to capture a wide of a set of environmental variations as possible, we construct a Synthesized Heterogeneous RE-id Domain (SHRED) from existing re-ID source domains under the following constraints:

- We avoid the use of source domains that have overlaps to ensure no one individual takes on two identities in the source domain. Some examples of source domains with overlap include CUHK02 [28]– CUHK01 [29], DukeMTMC-reID [56]– DukeMTMC4ReID [16] and Market-1501 [52]– MARS [51].
- We avoid any gait domains such as [54] in our SHRED source domain because they are staged in a studio environment with uniform background.
- We avoid source domains with less than or equal to 200 identities because they will be dwarfed by the larger source domains. Such small source domains include: Shinpuhkan [23], RAiD [8], V47 [40], HDA Person [12], WARD [35], CAVIAR4ReID [7], MPR Drone [25], RPIfield [53], PKU-Reid [33], QMUL iLIDS [55], SAIVT-SoftBio [4], ETH 1,2,3 [37].
- For each selected source domain, we combine data from training, validation and testing to ensure we have the largest possible variation in the source domain.
- For each source domain, we eliminate any individuals who doesn't appear in more than one camera as we want to ensure that the embedding learned from the source domain is for cross-camera comparison.

Based on the above constraints we are left with 12 source domains: 3DPeS [3], iLIDS-VID [41], VIPeR [17], PRID 2011 [19], GRID [32], CUHK03 [30], Market-1501 [52], DukeMTMC-reID [56], CUHK02 [28], MSMT17 [42], Airport [22], and End-to-End Deep Learning for Person Search [44]. Of the 12 source domains, Market-1501 is a common source domain used for testing and it also overlaps with MARS video source domain which is another common source domain used for testing. As such, we leave Market-1501 out of the proposed SHRED to allow for testing on a large video and image datasets. Finally, PRID 2011 was excluded from our SHRED source domain because it only has 200 individuals appearing in multiple camera views.

The resulting SHRED [1] source domain contains a heterogeneous mix of 10 different domains, with the details of the domain makeup shown in Table 1. As stated in our selection constraints, the number of images used will be less than that originally reported for the respective domains because distractor identities or identities that don't appear in multiple cameras are removed in pre-processing.

Note that our proposed SHRED source domain contains DukeMTMC-reID and CUHK03. When we report results for DukeMTMC-reID, we remove it from our source domain and replace it with Market-1501 (i.e. SHRED 2 from Table 1). Similarly, when reporting results for CUHK03, we remove it and replace it with Market-1501 (i.e. SHRED 3 from Table 1).

4. Experiment Setup

The efficacy of leveraging the proposed ktCUDA and SHRED is investigated through a series of experiments across different image and video benchmark datasets. The experimental setup in this paper is described below.

Datasets – We leverage three image datasets – Market-1501, CUHK03 and DukeMTMC-reID – to evaluate the proposed domain adaptation approach in a single-shot retrieval setting. In addition, we also use two video datasets – MARS and PRID – to evaluate the proposed approach in a multi-shot setting. When testing on Market-1501, MARS, and PRID, the source domain consists of SHRED 1 (Table 1). When testing on CUHK03, the source domain consists of SHRED 3 (Table 1). When testing on DukeMTMC-reID, the source domain consists of SHRED 2 (Table 1).

[1] List of images in SHRED 1, 2, 3 as well as the DCNN weights trained on SHRED 1, 2, and 3 will be released.

Table 2. Direct transfer (SHRED) and unsupervised domain adaptation (SHRED+ktCUDA) performance on benchmark re-ID datasets compared to published methods. 1st/2nd/3rd best results are in **red**/**blue**/**cyan**. Multisource domain method in magenta.

Methods	Market -1501 [52]		MARS[51]		CUHK03[30]		Duke MTMC-reID[56]		PRID[19]			Avg.	
	R1	mAP	R1	mAP	R1	mAP	R1	mAP	R1	R5	R20	**R1**	**mAP***
AML[47]	44.7	18.4	-	-	31.4	-	-	-	-	-	-	-	-
PTGAN [42]	38.6	-	-	-	24.8	-	27.4	-	-	-	-	-	-
PUL [11]	44.7	20.1	-	-	-	-	30.4	16.4	-	-	-	-	-
SPGAN+LMP [10]	58.1	26.9	-	-	-	-	46.4	26.2	-	-	-	-	-
TJ-AIDL [39]	58.2	26.5	-	-	-	-	44.3	23.0	-	-	-	-	-
HHL [60]	62.2	31.4	-	-	-	-	**46.9**	**27.2**	-	-	-	-	-
TFusion [60]	60.8	-	-	-	-	-	-	-	-	-	-	-	-
UnKISS [24]	-	-	22.3	10.6	-	-	-	-	58.1	81.9	96.0	-	-
SMP [31]	-	-	23.9	10.5	- -	-	-	-	**80.9**	**95.6**	**99.4**	-	-
DGM+MLAPG [46]	-	-	24.6	11.8	-	-	-	-	73.1	92.5	**99.0**	-	-
DGM+IDE [46]	-	-	36.8	21.3	-	-	-	-	56.4	81.3	96.4	-	-
RACE [48]	-	-	43.2	24.5	-	-	-	-	50.6	79.4	91.8	-	-
DAL [6]	-	-	**46.8**	21.4	-	-	-	-	**85.3**	**97.0**	**99.6**	-	-
TAUDL [27]	**63.7**	**41.2**	43.8	**29.1**	**44.7**	**31.2**	**61.7**	**43.5**	49.4	78.7	98.9	**52.7**	**36.3**
JSTL[43]	44.7	18.4	-	-	33.2	-	-	-	-	-	-	-	-
CAMEL [49]	54.5	26.3	-	-	**39.4**	-	-	-	-	-	-	-	-
SyRI [2]	**65.7**	-	-	-	-	-	-	-	43.0	-	-	-	-
SHRED	53.9	**32.4**	**53.3**	**33.6**	28.5	26.1	40.9	24.5	76.4	94.4	98.9	**50.6**	**29.2**
SHRED+ktCUDA	**68.6**	**49.4**	**57.2**	**36.0**	**44.4**	**41.6**	**58.7**	**40.9**	**84.3**	**96.6**	98.9	**62.6**	**42.0**
GCS [5](*Sup.*)	93.5	81.6	-	-	88.8	97.2	84.9	69.5	-	-	-	-	-
HDLF [50](*Sup.*)	-	-	86.4	79.3	-	-	-	-	95.7	99.1	-	-	-

* PRID is excluded from average **mAP** because mAP is not a standard used to evaluate PRID [19].

Table 3. Direct transfer (SHRED) and unsupervised domain adaptation (SHRED+ktCUDA) re-ranked (rr) [59] results.

Methods	Market-1501[52]		MARS[51]		CUHK03[30]		DukeMTMC-reID[56]	
	$R1_{rr}$	mAP_{rr}	$R1_{rr}$	mAP_{rr}	$R1_{rr}$	mAP_{rr}	$R1_{rr}$	mAP_{rr}
SHRED	57.4	43.7	MSMT1753.4	41.1	37.6	37.7	47.0	38.0
SHRED + ktCUDA	71.3	60.5	58.6	45.4	49.0	51.3	63.5	55.1

In [27, 31] for Market-1501, all images of an individual per camera are treated as a single tracklet and for MARS, a single tracklet per individual per camera is manually selected. For our experiments, we use the sequence ID in the Market-1501 dataset for tracklets, and for MARS we make no manual selection. As such, we have a harder and more realistic scenario of multiple tracklets of individuals per camera.

For all datasets, we follow the same test gallery-query split as in [27]. All evaluation are done using the evaluation code provided with the datasets. For datasets without evaluation code, Market-1501 evaluation code was used.

Implementation Detail – Three parameters of our k-reciprocal tracklet Clustering for Unsupervised Domain Adaptation (ktCUDA) algorithm are: i) the number of domain adaptation iteration I, ii) the $k-$reciprocal distance threshold K (3), and iii) the subgraph cardinality threshold T (5). For all our experiments, we do at-least $I = 2$ round of adaptation and only go above if the performance increases in the next round. We do early stopping only if number of cluster exceeds a soft upper bound on expected number unique individuals of 850. As the largest of the datasets contain around 700-750 identities, we number larger than that was chosen and hence 850 was picked. For deciding the values for K and T, we chose the minimum number of cameras and cardinality that makes the domain iteration viable. Therefore, for datasets (Market-1501,MARS, and DukeMTMC-reID) with camera networks larger than two (that is an individual could potentially appear in more than 2 cameras), we set $K = 2$ and $T = 2$. For datasets (CUHK03 and PRID) with two camera network, we set $K = 1$ and $T = 1$ because we can't expect clusters larger than two since only two cameras exist in the network.

Network Architecture – All experiments were performed using the modified ResNet-50 network introduced in [18], which has an additional 1024 dimensional fully connected layer and a 128 dimensional embedding layer (see Fig. 2).

Training – Training on the source domain is initialized with pre-trained ImageNet [9] weights. Domain adaptation is initialized with weights trained on the source domain.

We keep the same training parameters provided by [18] with the exception of the number of iteration. We vary this based on our training data. For the source domain where we have much larger number of data due to the combination of several dataset, we set the number of iterations to $50,000$. For domain adaptation we use $25,000$ iteration for all datasets except PRID where we use $6,000$ iteration since it has far fewer images.

5. Results and Discussion

To compare the proposed ktCUDA and SHRED, we use the common Cumulative Matching Characteristic (CMC) and mean Average Precision (mAP) metrics. We evaluate against the state-of-the-art methods for domain adaption (where unlabelled target domain is used for training) and direct transfer (where target domain data is not used at all).

Table 4. Comparison of SHRED direct transfer results with state-of-the-art unsupervised direct transfer methods on Market-1501. 1st/2nd/3rd best results are in **red**/**blue**/**cyan**. Multisource domain method in magenta.

Methods	Source Domain	R1	mAP
TFusion[20]	GRID	20.7	-
TFusion[20]	VIPeR	24.7	-
TFusion[20]	CUHK01	29.4	-
PTGAN[42]	CUHK03	27.8	-
HHL[60]	CUHK03	42.2	20.3
PTGAN[42]	DukeMTMC-reID	33.5	-
HHL[60]	DukeMTMC-reID	44.6	20.6
T&P[38]	DukeMTMC-reID	46.8	19.1
One-Shot[14]	DukeMTMC-reID	50.6	**23.7**
TJAIDL[39]	DukeMTMC-reID	**57.1**	**26.2**
SyRI[2]	CUHK03 + DukeMTMC-reID	44.7	-
SyRI[2]	CUHK03 + DukeMTMC-reID+SyRI	**54.3**	-
ktCUDA	SHRED	**53.9**	**32.4**

Table 5. Domain adaptation (ktCUDA) and direct transfer (SHRED) comparison for Market-1501. 1st/2nd/3rd best results are in **red**/**blue**/**cyan**.Multisource domain method in magenta.

Methods	Source Domain	Direct Transfer		Domain Adapt.	
		R1	mAP	R1	mAP
HHL[60]	CUHK03	42.2	20.3	56.8	29.8
PTGAN[42]	CUHK03	–	–	27.8	–
ktCUDA	CUHK03	33.5	15.5	57.5	**35.2**
PTGAN[42]	DukeMTMC-reID	33.5	-	38.6	-
SPGAN+LMP[10]	DukeMTMC-reID	43.1	17.0	58.1	26.9
HHL[60]	DukeMTMC-reID	44.6	**20.6**	62.2	31.4
TJAIDL[39]	DukeMTMC-reID	**57.1**	**26.2**	58.2	26.5
ktCUDA	DukeMTMC-reID	40.3	17.6	56.0	32.6
TAUDL[27]*	None	-	-	**63.7**	**41.2**
CAMEL	7set*	41.4	14.1	54.5	26.3
SyRI	CUHK03+ DukeMTMC-reID+SyRI	**44.7**	–	**65.7**	–
ktCUDA	SHRED	**53.9**	**32.4**	**68.6**	**49.4**

7set*: VIPeR, CUHK01, CUHK03, PRID, 3DPeS, i-LIDS and Shinpuhkan.

5.1. Domain Adaptation

The result of the proposed k-reciprocal tracklet Clustering for Unsupervised Domain Adaptation (ktCUDA) algorithm can be found in Table 2 (indicated as **SHRED + ktCUDA**) with comparison to existing state-of-the-art approaches. It can be clearly observed that the proposed ktCUDA approach is the state-of-the art method for Market-1501 (**+8.2** mAP), MARS (**+6.9** mAP) and CUHK03 (**+10.4** mAP) datasets based on mAP amongst the tested methods. We also get competitive performance to state-of-the-art methods on DukeMTMC-reID and PRID datasets.

In Table 2, we present the average rank-1 (**R1**) and mean average precision (**mAP**) across all five test datasets as summary metrics. Based on the average performance, ktCUDA is **+9.9 R1** and **+5.7 mAP** better the current state-of-the-art. Finally, the efficacy of ktCUDA is shown by the observation that it is the only method that is consistently ranked as the best or competitive second best method on all five test datasets.

For the sake of completeness, we also present the re-ranked [59] results in Table 3.

Comparison to multi-source domain methods – While the general performance of SHRED+ktCUDA is consistently in the top two across all test sets it is worth looking at its performance relative to other multi-source domain methods (highlighted in magenta in Table 2). Comparing to CAMEL, SHRED+ktCUDA uses 10 datasets versus CAMEL which uses 7 datasets and SHRED+ktCUDA outperforms CAMEL method. However, SHRED+ktCUDA has $\sim$ $250k$ images in the source domain compared to CAMEL's $\sim$ $45k$ images. Comparing to SyRI, which uses more than 1.6 million synthetic images and $\sim$ $45k$ real world images, SHRED+ktCUDA still out performs SyRI, thus motivating the need for real-world diverse images over synthetic images.

5.2. Direct Transfer (SHRED without ktCUDA)

It can be observed that the proposed SHRED source domain is quite effective across all test datasets as seen in Table 2 (indicated as **SHRED**). In particular, the performance on MARS dataset stands out. For MARS, our direct transfer results are **+4.5** mAP better than state-of-the-art domain transfer methods, even when these methods use unlabelled MARS data in the training.

A comparison of the proposed SHRED source domain for direct transfer with domain transfer methods on the Market-1501 dataset based on previously published results in literature can be found in Table 4. As expected, we can see the proposed SHRED source domain outperforms existing source domains on mAP by a large margin.

Considering synthetic dataset augmentation (SyRI [2]) results in Table 4, we observe that its Rank-1 result is slightly higher than the proposed SHRED source domain. Unfortunately this analysis is not conclusive without mAP. However, [2] also report Rank-1 result for single-shot re-ID on PRID dataset as 15%. Our direct transfer for PRID single-shot re-ID gets a rank-1 accuracy of 22%. Therefore, while synthetic data augmentation is good for giving some variability, real data from multiple sources is ultimately better.

5.3. Domain adaptation boost

For the three best source domain direct transfer results in Table 4, TJ-AIDL [39], SyRI [2] and the proposed ktCUDA, we look at the improvement achieved by domain transfer over the direct transfer results in Table 5. From Table 5 we observe, of the methods with best direct transfer results, the proposed method has best domain adaptation boost for mAP. Of particular importance is the SyRI method which uses a much larger source domain than SHRED+ktCUDA and has similar direct transfer accuracy as SHRED+ktCUDA. From the same starting point, SHRED+ktCUDA was able to achieve higher Rank-1 result than SyRI showing that ktCUDA is very effective strategy for domain adaption.

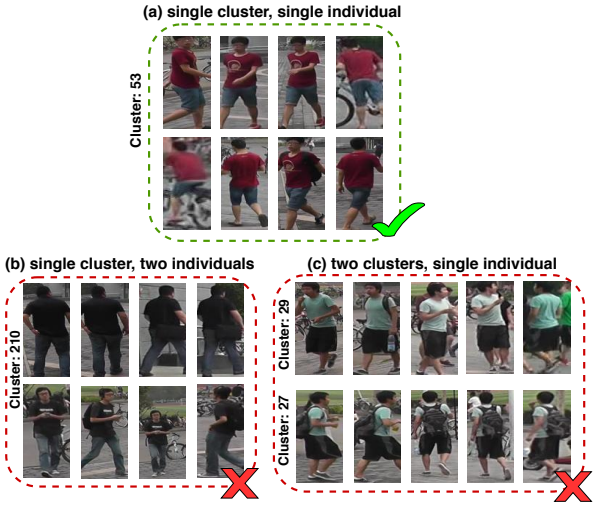

Figure 3. Different types of clusters arising from the proposed ktCUDA algorithm - (a) Good cluster (GC): A cluster containing single individual who does not appear in any other clusters, (b) Mixed cluster (MC): a cluster with two or more different individuals and (c) Divided clusters (DC): an individual is split across two or more different clusters. Best viewed in color.

This shows that while a heterogeneous source domain is very effective at giving a good initialization, the proposed ktCUDA is also well-suited for adapting to a new domain.

We test our proposed ktCUDA approach with DukeMTMC-reID as the source domain and Market-1501 as the target domain as well in Table 5. This tests how well ktCUDA works for domain adaptation without using our proposed SHRED as the source domain. We can see that the proposed ktCUDA approach outperforms existing domain adaptation methods that use DukeMTMC-reID as the source domain. Furthermore, when combined with SHRED the proposed ktCUDA approach can get a significant boost over existing state-of-the-art methods.

5.4. k-Reciprocal Tracklet Clusters

To further evaluate ktCUDA, we take a closer look at our k-reciprocal tracklet clustering. We note that k-reciprocal clustering results in three main types of clusters: Good Clusters (GC) containing only a single individual who does not appear in any other clusters, mixed clusters (MC) where multiple different individuals are in a single cluster and divided clusters (DC) where a single individual appears in multiple clusters. An example of the three types of clusters are shown in Fig. 3 (a)-(c). (Note there is also a third error type which is a mix of MC and DC.)

Interestingly, of the two types of errors –mixed clusters and divided clusters– we find that the presence of divided clusters doesn't negatively impact triplet loss fine-tuning. If we plot the distance between divided clusters (a.k.a. intra person) and distance between clusters with different individuals (a.k.a. inter person) before and after fine-tuning (Fig. 4), we see both distances increase but the inter person distances increases more than intra person distance. Meaning even with the presence of divided clusters, the triplet loss is able to separate different individuals because triplet loss is not directly forcing different individuals closer. However, mixed clusters do present a problem as that will force different individuals closer.

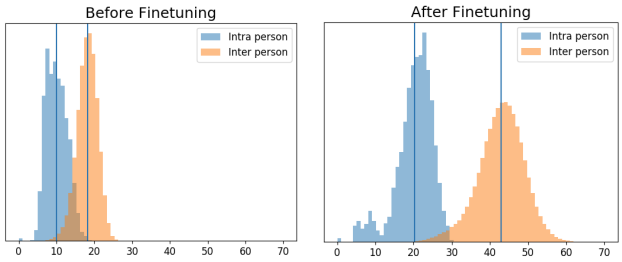

Figure 4. Separation of clusters with same vs different individuals. Distance between clusters with same individuals (divided clusters Fig. 3) shown as *Intra person* and distance between clusters with different individuals shown as *Inter person*. Plots shown before triplet loss fine-tuning (left) and after fine-tuning (right). During fine-tuning it can be seen that the *Inter person* clusters are pushed further away than *Intra person* clusters.

6. Conclusion

In this work, we presented new strategies for unsupervised person re-ID using unlabelled data from a target domain. Our method addressed the two main limitations of the current domain adaptation approaches: first, using source domain distance metrics for pseudo-labelling in target domain and second, relying heavily on limited source domain data. The two problems were addressed by the proposed k-reciprocal tracklet Clustering for Unsupervised Domain Adaptation (ktCUDA) method and the proposed comprehensive Synthesized Heterogeneous RE-id Domain (SHRED), respectively. Addressing these issues allowed the presented ktCUDA method to become more scalable for real-world applications. Extensive evaluation was done on image and video person re-ID benchmark datasets to validate the effectiveness of the proposed ktCUDA in outperforming other state-of-the art unsupervised domain adaptation methods in person re-ID.

References

[1] Re-id datasets. `https://github.com/NEU-Gou/awesome-reid-dataset`. Accessed: 2019-03-18.

[2] S. Bak, P. Carr, and J.-F. Lalonde. Domain adaptation through synthesis for unsupervised person re-identification. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 189–205, 2018.

[3] D. Baltieri, R. Vezzani, and R. Cucchiara. 3dpes: 3d people dataset for surveillance and forensics. In *Proceedings of the*

1st International ACM Workshop on Multimedia access to 3D Human Objects, pages 59–64, Scottsdale, Arizona, USA, Nov. 2011.

[4] A. Bialkowski, S. Denman, P. Lucey, S. Sridharan, and C. B. Fookes. A database for person re-identification in multi-camera surveillance networks. In *Digital Image Computing : Techniques and Applications (DICTA 2012)*, pages 1–8, Esplanade Hotel, Fremantle, WA, 2012. IEEE.

[5] D. Chen, D. Xu, H. Li, N. Sebe, and X. Wang. Group consistent similarity learning via deep crf for person re-identification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8649–8658, 2018.

[6] Y. Chen, X. Zhu, and S. Gong. Deep association learning for unsupervised video person re-identification. *CoRR*, abs/1808.07301, 2018.

[7] D. S. Cheng, M. Cristani, M. Stoppa, L. Bazzani, and V. Murino. Custom pictorial structures for re-identification. In *British Machine Vision Conference (BMVC)*, 2011.

[8] A. Das, A. Chakraborty, and A. K. Roy-Chowdhury. Consistent re-identification in a camera network. In *European Conference on Computer Vision*, pages 330–345. Springer, 2014.

[9] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009.

[10] W. Deng, L. Zheng, Q. Ye, G. Kang, Y. Yang, and J. Jiao. Image-image domain adaptation with preserved self-similarity and domain-dissimilarity for person re-identification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 994–1003, 2018.

[11] H. Fan, L. Zheng, C. Yan, and Y. Yang. Unsupervised person re-identification: Clustering and fine-tuning. *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)*, 14(4):83, 2018.

[12] D. Figueira, T. Matteo, N. Athira, N. Jacinto, and A. Bernardino. The hda+ data set for research on fully automated re-identification systems. In *European Conference on Computer Vision*, pages 241–255. Springer, 2014.

[13] Y. Fu, Y. Wei, G. Wang, J. Li, X. Zhou, H. Shi, and T. Huang. One shot domain adaptation for person re-identification. *arXiv preprint arXiv:1811.10144*, 2018.

[14] Y. Fu, Y. Wei, G. Wang, J. Li, X. Zhou, H. Shi, and T. Huang. One shot domain adaptation for person re-identification. *CoRR*, abs/1811.10144, 2018.

[15] S. Gong, C. Marco, C. L. Chen, and T. M. Hospedales. "the re-identification challenge.". In *Person re-identification*, pages 1–20. Springer, 2014.

[16] M. Gou, S. Karanam, W. Liu, O. Camps, and R. J. Radke. Dukemtmc4reid: A large-scale multi-camera person re-identification dataset. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 10–19, 2017.

[17] D. Gray, S. Brennan, and H. Tao. Evaluating appearance models for recognition, reacquisition, and tracking. In *PETS*, 2007.

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

[19] M. Hirzer, C. Beleznai, P. M. Roth, and H. Bischof. Person Re-Identification by Descriptive and Discriminative Classification. In *Proceedings of the Scandinavian Conference on Image Analysis*, 2011.

[20] L. Jianming, W. Chen, Q. Li, and C. Yang. Unsupervised cross-dataset person re-identification by transfer learning of spatial-temporal patterns. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7948–7956, 2018.

[21] E. B. M. G. M. E. K. Kalayeh, Mahdi M. and M. Shah. Human semantic parsing for person re-identification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1062–1071, 2018.

[22] S. Karanam, M. Gou, Z. Wu, A. Rates-Borras, O. Camps, and R. J. Radke. A systematic evaluation and benchmark for person re-identification. In *Proceedings of the IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 523–536, 2018.

[23] Y. Kawanishi, Y. Wu, M. Mukunoki, and M. Minoh. Shinpuhkan2014: A multi-camera pedestrian dataset for tracking people across multiple cameras. In *Proceedings of the 20th Korea-Japan Joint Workshop on Frontiers of Computer Vision*, 2014.

[24] F. M. Khan and F. Bremond. Unsupervised data association for metric learning in the context of multi-shot person re-identification. In *2016 13th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS)*, pages 256–262, 2016.

[25] R. Layne, T. M. Hospedales, and S. Gong. Investigating open-world person re-identification using a drone. In *European Conference on Computer Vision*, pages 225–240. Springer, 2014.

[26] K. Li, Z. Ding, K. Li, Y. Zhang, and Y. Fu. Support neighbor loss for person re-identification. In *2018 ACM Multimedia Conference on Multimedia Conference*, pages 1492–1500. ACM, 2018.

[27] M. Li, X. Zhu, and S. Gong. Unsupervised person re-identification by deep learning tracklet association. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 737–753, 2018.

[28] W. Li and X. Wang. Locally aligned feature transforms across views. In *CVPR*, 2013.

[29] W. Li, R. Zhao, and X. Wang. Human reidentification with transferred metric learning. In *ACCV*, 2012.

[30] W. Li, R. Zhao, T. Xiao, and X. Wang. Deepreid: Deep filter pairing neural network for person re-identification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 152–159, 2014.

[31] D. W. Liu, Zimo and H. Lu. Stepwise metric promotion for unsupervised video person re-identification. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2429–2438, 2017.

[32] C. C. Loy, C. Liu, and S. Gong. Person re-identification by manifold ranking. In *ICIP*, 2013.

[33] L. Ma, H. Liu, L. Hu, C. Wang, and Q. Sun. Orientation driven bag of appearances for person re-identification. *CoRR*, abs/1605.02464, 2016.

[34] P. Marchwica, M. Jamieson, and P. Siva. An evaluation of deep cnn baselines for scene-independent person re-identification. In *2018 15th Conference on Computer and Robot Vision (CRV)*, pages 297–304. IEEE, 2018.

[35] N. Martinel and C. Micheloni. Re-identify people in wide area camera network. In *CVPRW*, 2012.

[36] D. Qin, S. Gammeter, L. Bossard, T. Quack, and L. V. Gool. Hello neighbor: Accurate object retrieval with k-reciprocal nearest neighbors. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 777–784, 2011.

[37] W. R. Schwartz and L. S. Davis. Learning discriminative appearance-based models using partial least squares. In *SIBGRAPI*, pages 322–329. IEEE Computer Society, 2009.

[38] L. Song, C. Wang, L. Zhang, B. Du, Q. Zhang, C. Huang, and X. Wang. Unsupervised domain adaptive re-identification: Theory and practice. *arXiv preprint arXiv:1807.11334*, 2018.

[39] J. Wang, X. Zhu, S. Gong, and W. 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.

[40] S. Wang, M. Lewandowski, J. Annesley, and J. Orwell. Re-identification of pedestrians with variable occlusion and scale. In *Proceedings of the IEEE International Conference on Computer Vision Workshops*, pages 1876–1882, 2011.

[41] T. Wang, S. Gong, X. Zhu, and S. Wang. Person re-identification by discriminative selection in video ranking. In *Proceedings of the IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 2501–2514, 2016.

[42] L. Wei, Z. Shiliang, G. Wen, and Q. 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.

[43] T. Xiao, H. Li, W. Ouyang, and X. Wang. Learning deep feature representations with domain guided dropout for person re-identification. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1249–1258, 2016.

[44] T. Xiao, S. Li, B. Wang, L. Lin, and X. Wang. End-to-end deep learning for person search. *CoRR*, abs/1604.01850, 2016.

[45] J. Yang, X. Shen, T. Xinmei, L. Houqiang, H. Jianqiang, and X.-S. Hua. Local convolutional neural networks for person re-identification. In *ACM Multimedia Conference on Multimedia Conference*, pages 1074–1082. ACM, 2018.

[46] A. J. M. L. Z. J. L. Ye, Mang and P. C. Yuen. Dynamic label graph matching for unsupervised video re-identification. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 5142–5150, 2017.

[47] J. Ye, Z. Zhao, and H. Liu. Adaptive distance metric learning for clustering. In *2007 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–7. IEEE, 2007.

[48] X. L. Ye, Mang and P. C. Yuen. Robust anchor embedding for unsupervised video person re-identification in the wild. In *European Conference on Computer Vision*, pages 170–186. Springer, 2018.

[49] H.-X. Yu, A. Wu, and W.-S. Zheng. Cross-view asymmetric metric learning for unsupervised person re-identification. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 994–1002, 2017.

[50] C. T. Zeng, Mingyong and Z. Wu. Person re-identification with hierarchical deep learning feature and efficient xqda metric. In *ACM Multimedia Conference on Multimedia Conference*, pages 1838–1846. ACM, 2018.

[51] L. Zheng, Z. Bie, Y. Sun, J. Wang, C. Su, S. Wang, and Q. Tian. Mars: A video benchmark for large-scale person re-identification. In *European Conference on Computer Vision*, pages 868–884. Springer, 2016.

[52] L. Zheng, L. Shen, L. Tian, S. Wang, J. Wang, and Q. Tian. Scalable person re-identification: A benchmark. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1116–1124, 2015.

[53] M. Zheng, S. Karanam, and R. J. Radke. Rpifield: A new dataset for temporally evaluating person re-identification. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, June 2018.

[54] S. Zheng, K. Huang, and T. Tan. Evaluation framework on translation-invariant representation for cumulative foot pressure image. In *International Conference on Image Processing(ICIP)*, Brussels, Belgium, 2011.

[55] W.-S. Zheng, S. Gong, and T. Xiang. Associating groups of people. In *BMVC*, 2009.

[56] Z. Zheng, L. Zheng, and Y. Yang. Unlabeled samples generated by gan improve the person re-identification baseline in vitro. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3754–3762, 2017.

[57] Z. Zheng, L. Zheng, and Y. Yang. Pedestrian alignment network for large-scale person re-identification. *IEEE Transactions on Circuits and Systems for Video Technology*, 2018.

[58] Z. Zhong, Z. Liang, C. Donglin, and L. Shaozi. Re-ranking person re-identification with k-reciprocal encoding. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1318–1327, 2017.

[59] Z. Zhong, L. Zheng, D. Cao, and S. Li. Re-ranking person re-identification with k-reciprocal encoding. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1318–1327, 2017.

[60] Z. Zhong, L. Zheng, S. Li, and Y. Yang. Generalizing a person retrieval model hetero-and homogeneously. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 172–188, 2018.

[61] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2223–2232, 2017.