Adversarial Open-World Person Re-Identification

Xiang Li[1], Ancong Wu[1] , and Wei-Shi Zheng[1,2,3]⋆[0000−0001−8327−0003]

[1] Sun Yat-sen University
{lixiang47,wuancong}@mail2.sysu.edu.cn
wszheng@ieee.org
[2] Inception Institute of Artificial Intelligence, United Arab Emirates
[3] Key Laboratory of Machine Intelligence and Advanced Computing, MOE

Abstract. In a typical real-world application of re-id, a watch-list (gallery set) of a handful of target people (*e.g.* suspects) to track around a large volume of non-target people are demanded across camera views, and this is called the open-world person re-id. Different from conventional (closed-world) person re-id, a large portion of probe samples are not from target people in the open-world setting. And, it always happens that a non-target person would look similar to a target one and therefore would seriously challenge a re-id system. In this work, we introduce a deep open-world group-based person re-id model based on adversarial learning to alleviate the attack problem caused by similar non-target people. The main idea is learning to attack feature extractor on the target people by using GAN to generate very target-like images (imposters), and in the meantime the model will make the feature extractor learn to tolerate the attack by discriminative learning so as to realize group-based verification. The framework we proposed is called the adversarial open-world person re-identification, and this is realized by our Adversarial PersonNet (APN) that jointly learns a generator, a person discriminator, a target discriminator and a feature extractor, where the feature extractor and target discriminator share the same weights so as to makes the feature extractor learn to tolerate the attack by imposters for better group-based verification. While open-world person re-id is challenging, we show for the first time that the adversarial-based approach helps stabilize person re-id system under imposter attack more effectively.

1 Introduction

Person re-identification (re-id), which is to match a pedestrian across disjoint camera views in diverse scenes, is practical and useful for many fields, such as public security applications and has gained increasing interests in recent years [3, 4, 6, 10, 11, 22, 34, 36, 37, 40, 42, 45]. Rather than re-identifying every person in a multiple camera network, a typical real-world application is to re-identify or track only a handful of target people on a watch list (gallery set), which is called

⋆ corresponding author

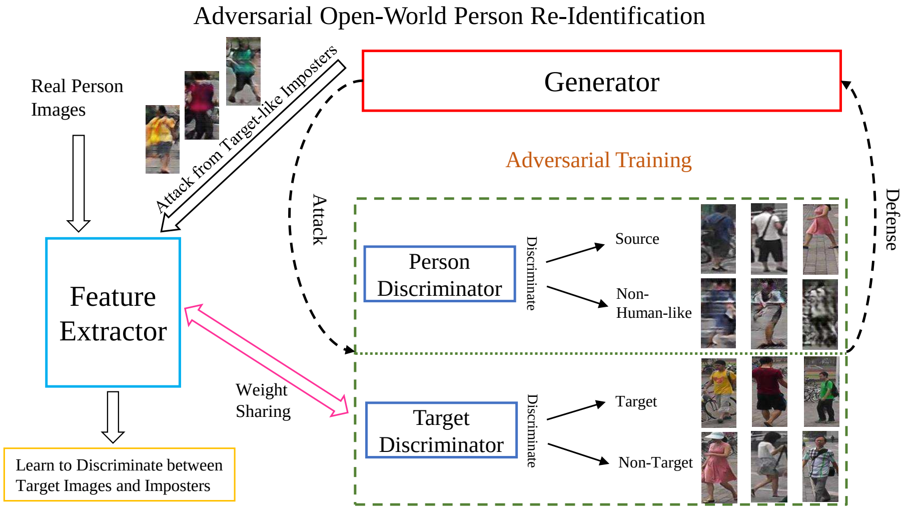

Fig. 1: Overview of adversarial open-world person re-identification. The goal for the generator is to generate target-like images, while we have two discriminators here. The person discriminator is to discriminate whether the generated images are from source dataset (*i.e.* being human-like). And the target discriminator is to discriminate whether the generated images are of target people. By the adversarial learning, we aim to generate images beneficial for training a better feature extractor for telling target person images apart from non-target ones.

the open-world person re-id problem [4,42,46]. While target people will reappear in the camera network at different views, a large volume of non-target people, some of which could be very similar to target people, would appear as well. This contradicts to the conventional closed-world person re-id setting that all probe queries are belonging to target people on the watch list. In comparison, the open-world person re-id is extremely challenging because both target and non-target (irrelevant) people are included in the probe set.

However, the majority of current person re-identification models are designed for the closed-world setting [6, 33, 36–38, 40, 43, 45] rather than the open-world one. Without consideration of discriminating target and non-target people during learning, these approaches are not stable and could often fail to reject a query image whose identity is not included in the gallery set. Zheng *et al.* [42] considered this problem and proposed open-world group-based verification model. Their model is based on hand-crafted feature and transfer-learning-based metric learning with auxiliary data, but the results are still far from solving this challenge. More importantly, the optimal feature representation and target-person-specific information for open-world setting have not been learned.

In this work, we present an adversarial open-world person re-identification framework for 1) learning features that are suitable for open-world person re-id, and 2) learning to attack the feature extractor by generating very target-like imposters and make person re-id system learn to tolerate it for better verification. An end-to-end deep neural network is designed to realize the above two objectives, and an overview of this pipeline is shown in Fig. 1. The feature learning and the adversarial learning are mutually related and learned jointly, meanwhile the generator and the feature extractor are learned from each other iteratively to enhance both the efficiency of generated images and the discriminability of the feature extractor. To use the unlabeled images generated, we further incorporate a label smoothing regularization for imposters (LSRI) for this adversarial learning process. LSRI allocates equal probabilities of being any non-target people and zero probabilities of being target people to the generated target-like imposters, and it would further improve the discrimination ability of the feature extractor for distinguishing real target people from fake ones (imposters).

While GAN has been attempted in Person re-id models recently in [8,43,44] for generating images adapted from source dataset so as to enrich the training dataset on target task. However, our objective is beyond this conventional usage. By sharing the weights between feature extractor and target discriminator (see Fig. 2), our adversarial learning makes the generator and feature extractor interact with each other in an end-to-end framework. This interaction not only makes the generator produce imposters look like target people, but also more importantly makes the feature extractor learn to tolerate the attack by imposters for better group-based verification.

In summary, our contributions are more on solving the open-world challenge in person re-identification. It is the first time to formulate the open-world group-based person re-identification under an adversarial learning framework. By learning to attack and learning to defend, we realize four progresses in a unified framework, including generating very target-like imposters, mimicking imposter attacking, discriminating imposters from target images and learning re-id feature to represent. Our investigation suggests that adversarial learning is a more effective way for stabilizing person re-id system undergoing imposters.

2 Related Work

Person Re-Identification: Since person re-identification targets to identify different people, better feature representations are studied by a great deal of recent research. Some of the research try to seek more discriminative/reliable hand-crafted features [10, 17, 22, 23, 25, 26, 37]. Except that, learning the best matching metric [5, 6, 14, 18, 27, 33, 40] is also widely studied for solving the cross-view change in different environments. With the rapid development of deep learning, learning to represent from images [1, 7, 9, 20] is attracted for person re-id, and in particular Xiao *et al.* [38] came up with domain guided drop out model for training CNN with multiple domains so as to improve the feature learning procedure. Also, recent deep approaches in person re-identification are found to unify feature learning and metric learning [1, 31, 36, 45]. Although these deep learning methods are expressive for large-scale datasets, they tend to be resistless for noises and incapable of distinguishing non-target people apart from the target ones, and thus becomes unsuitable for the open-world setting. In comparison, our deep model aims to model the effect of non-target people during training and optimize the person re-id in the open-world setting.

Towards Open-World Person Re-Identification: Although the majority of works on person re-id are focusing on the closed-world setting, a few works have been reported on addressing the open-world setting. The work of Candela *et al.* [4] is based on Conditional Random Field (CRF) inference attempting to build connections between cameras towards open-world person re-identification. But their work lacks the ability to distinguish very similar identities, and with some deep CNN models coming up, features from multiple camera views can be well expressed by joint camera learning. Wang *et al.* [34] worked out an approach by proposing a new subspace learning model suitable for open-world scenario. However, group-based setting and interference defense is not considered. Also, their model requires a large volume of extra unlabeled data. Zhu *et al.* [46] proposed a novel hashing method for fast search in the open-world setting. However, Zhu *et al.* aimed at large scale open-world re-identification and efficiency is considered primarily. Besides, noiseproof ability is not taken into account. The most correlated work with this paper is formulated by Zheng *et al.* [41,42], where the group-based verification towards open-world person re-identification was proposed. They came up with a transfer relative distance comparison model (t-LRDC), learning a distance metric and transferring non-target data to target data in order to overcome data sparsity. Different from the above works, we present the first end-to-end learning model to unify feature learning and verification modeling to address the open-world setting. Moreover, our work does not require extra auxiliary datasets to mimic attack of imposters, but integrates an adversarial processing to make re-id model learn to tolerate the attack.

Adversarial Learning: In 2014, Szegedy *et al.* [32] have found out that tiny noises in samples can lead deep classifiers to mis-classify, even if these adversarial samples can be easily discriminated by human. Then many researchers have been working on adversarial training. Seyed-Mohsen *et al.* [28] proposed DeepFool, using the gradient of an image to produce a minimal noise that fools deep networks. However, their adversarial samples are towards individuals and the relation between target and non-target groups is not modelled. Thus, it does not well fit into the group-based setting. Nicolas Papernot *et al.* [29] formulated a class of algorithms by using knowledge of deep neural networks (DNN) architecture for crafting adversarial samples. However, rather than forming a general algorithm for DNNs, our method is more specific for group-based person verification and the imposter samples generated are more effective to this scenario. Later, SafetyNet by Lu *et al.* [24] was proposed with an RBF-SVM in full-connected layer to detect adversarial samples. However, we perform the adversarial learning at feature level to better attack the learned features.

3 Adversarial PersonNet

3.1 Problem Statement

In this work, we concentrate on open-world person re-id by group-based verification. The group-based verification is to ensure a re-id system to identify whether a query person image comes from target people on the watch list. In this scenario, people out of this list/group are defined as non-target people.

Our objective is to unify feature learning by deep convolution networks and adversarial learning together so as to make the extracted feature robust and resistant to noise for discriminating between target people and non-target ones. The adversarial learning is to generate target-like imposter images to attack the feature extraction process and simultaneously make the whole model learn to distinguish these attacks. For this purpose, we propose a novel deep learning model called Adversarial PersonNet (APN) that suits open-world person re-id.

To better express our work under this setting in the following sections, we suppose that N_T target training images constitute a target sample set X_T sampled from C_T target people. Let $\boldsymbol{x}_i^T$ indicate the ith target image and $y_i^T \in Y_T$ represents the corresponding person/class label. The label set Y_T is denoted by $Y_T = \{y_1^T, ..., y_{N_T}^T\}$ and there are C_T target classes in total. Similarly, we are given a set of C_S non-target training classes containing N_S images, denoted as $X_S = \{\boldsymbol{x}_1^S, ..., \boldsymbol{x}_{N_S}^S\}$, where $\boldsymbol{x}_i^S \in X_S$ is the ith non-target image. y_i^S is the class of $\boldsymbol{x}_i^S$ and $Y_S = \{y_1^S, ..., y_{N_S}^S\}$. Note that there is no identity overlap between target people and non-target people. Under open-world setting, $N_S \gg N_T$. The problem is to better determine whether a person is on the target-list; that is for a given image $\boldsymbol{x}$ without knowing its class y, determine if $y \in Y_T$. We use $f(\boldsymbol{x}, \boldsymbol{\theta})$ to represent the extracted feature from image $\boldsymbol{x}$, and $\boldsymbol{\theta}$ is the weight of the feature extraction part of the CNN.

3.2 Learning to Attack by Adversarial Networks

Always, GANs are designed to generate images similar to those in the *source set*, which is constituted by both target and non-target image sets. A generator G and a discriminator D_p are trained adversarially. However the generator G normally only generates images looking like the ones in the source set and the discriminator D_p discriminates the generated images from the source ones. In this case, our source datasets are all pedestrian images, so we call such D_p the *person discriminator* in response to its ability of determining whether an image is of pedestrian-like images. D_p is trained by minimizing the following loss function:

$$L_{D_p} = -\frac{1}{m}\sum_{i=1}^{m}[\log D_p(\boldsymbol{x}) + \log{(1 - D_p(G(\boldsymbol{z})))}], \qquad (1)$$

where m is the number of samples, $\boldsymbol{x}$ represents image from source dataset and $\boldsymbol{z}$ is a noise randomly generated.

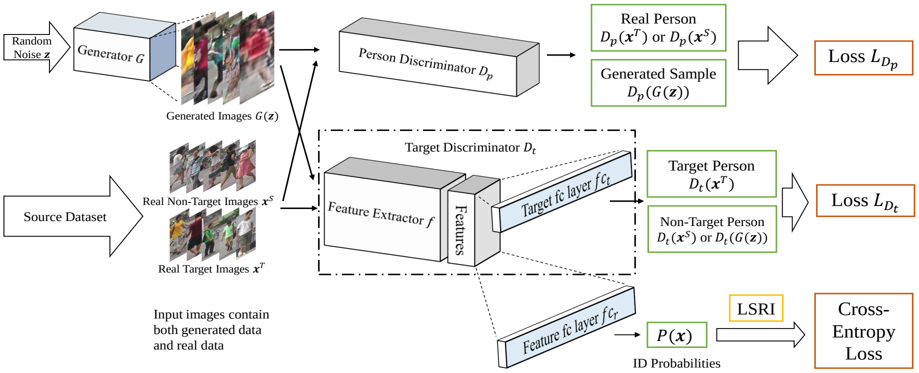
Random Noise $\boldsymbol{z}$
Generator G
Generated Images $G(\boldsymbol{z})$
Source Dataset
Real Non-Target Images $\boldsymbol{x}^S$
Real Target Images $\boldsymbol{x}^T$
Input images contain both generated data and real data
Person Discriminator D_p
Real Person $D_p(\boldsymbol{x}^T)$ or $D_p(\boldsymbol{x}^S)$
Generated Sample $D_p(G(\boldsymbol{z}))$
Loss L_{D_p}
Target Discriminator D_t
Feature Extractor f
Features
Target fc layer fc_t
Target Person $D_t(\boldsymbol{x}^T)$
Non-Target Person $D_t(\boldsymbol{x}^S)$ or $D_t(G(\boldsymbol{z}))$
Loss L_{D_t}
Feature fc layer fc_r
$P(\boldsymbol{x})$
ID Probabilities
LSRI
Cross-Entropy Loss

Fig. 2: Adversarial PersonNet structure. Two discriminators D_p and D_t accept samples from both datasets and generator G. Since D_t shares the same weights with feature extractor f, we represent them as the same cuboid in this figure.

Suppose that there is a pre-trained feature extractor for person re-id task and in an attempt to steer generator G to produce not only pedestrian-like but also feature attacking images towards this feature extractor, we design a paralleled discriminator D_t with the following definition:

$$D_t(\boldsymbol{x}) = fc_t(f(\boldsymbol{x}, \boldsymbol{\theta})). \tag{2}$$

The discriminator D_t is to determine whether an image will be regarded as target image by feature extractor. $f(\boldsymbol{x}, \boldsymbol{\theta})$ indicates that part of D_t has the same network structure as feature extractor f and shares the same weights $\boldsymbol{\theta}$ (Actually, the feature extractor can be regarded as a part of D_t.). fc_t means a full-connected layer following the feature extractor apart from the one connected to original CNN (with a fc layer used to pre-train the feature extractor). So D_t shares the same ability of target person discrimination with the feature extractor. To induce the generator G for producing target-like images for attacking and ensure the discriminator D_t to tell the non-target and generated imposters apart from the target ones, we formulate a paralleled adversarial training of G and D_t as

$$\begin{aligned}\min_{G}\max_{D_t} V_t(D_t, G) = {} & \mathbb{E}_{\boldsymbol{x}^T \sim X_T}[\log D_t(\boldsymbol{x}^T)] \\ & + \mathbb{E}_{\boldsymbol{x}^S \sim X_S}[\log(1 - D_t(\boldsymbol{x}^S))] \\ & + \mathbb{E}_{\boldsymbol{z} \sim p_{\boldsymbol{z}}(\boldsymbol{z})}[\log(1 - D_t(G(\boldsymbol{z})))].\end{aligned} \tag{3}$$

We train D_t to maximize $D_t(\boldsymbol{x})$ when passed by a target image but minimize it when passed by a non-target image or a generated imposter image by G. Notice that this process only trains the final fc_t layer of D_t without updating the feature extractor weights $\boldsymbol{\theta}$, to prevent the feature extractor from being affected by discriminator learning when the generated images are not good enough. We

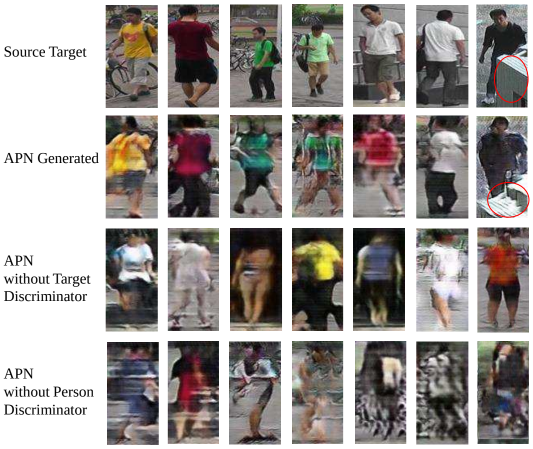
Source Target
APN Generated
APN without Target Discriminator
APN without Person Discriminator

Fig. 3: Examples of generated images. Although images produced by the generator are based on random noises, we can tell that the imposters generated by APN are very similar to targets. These similarities are mostly based on clothes, colors and postures (*e.g.* the fifth column). Moreover, surroundings are learned by APN as shown in the seventh column in the red circle.

call D_t the *target discriminator*. And we propose the loss function L_{D_t} for the training process of target discriminator D_t:

$$\begin{cases} L_{D_t} = -\frac{1}{m}\sum_{i=1}^{m}[\log Q_t(\boldsymbol{x}) + \log(1 - D_t(G(\boldsymbol{z})))], \\ Q_t(\boldsymbol{x}) = \begin{cases} D_t(\boldsymbol{x}), & \boldsymbol{x} \in X_T; \\ 1 - D_t(\boldsymbol{x}), & \boldsymbol{x} \in X_S. \end{cases} \end{cases} \tag{4}$$

We integrate the above into a standard GAN framework as follows:

$$\begin{aligned} &\min_G \max_{D_p} \max_{D_t} V'(D_p, D_t, G) = \\ &\mathbb{E}_{\boldsymbol{x}^T \sim X_T}[\log D_p(\boldsymbol{x}^T) + \log D_t(\boldsymbol{x}^T)] \\ &+ \mathbb{E}_{\boldsymbol{x}^S \sim X_S}[\log D_p(\boldsymbol{x}^S) + \log(1 - D_t(\boldsymbol{x}^S))] \\ &+ \mathbb{E}_{\boldsymbol{z} \sim p_{\boldsymbol{z}}(\boldsymbol{z})}[\log(1 - D_p(G(\boldsymbol{z}))) + \log(1 - D_t(G(\boldsymbol{z})))]. \end{aligned} \tag{5}$$

The collaboration of generator and couple discriminators is illustrated in Fig. 2. While GAN with only person discriminator will force the generator G to produce source-like person images, with the incorporation of the loss of target discriminator D_t, G is more guided to produce very much target-like imposter images. The target-like imposters, generated based on the discriminating ability of feature extractor, satisfy the usage of attacking the feature extractor. Examples of images generated by APN are shown in Fig. 3 together with the target images and the images generated by controlled groups (APN without target discriminator D_t and APN without person discriminator D_p) to indicate that our network indeed has the ability to generate target-like images. The generator G is trained to fool the target discriminator in the feature space, so that the generated adversarial images can attack the re-id system. While the target discriminator D_t is mainly to tell these attack apart from the target people so as to defend the re-id system.

3.3 Joint Learning of Feature Representation and Adversarial Modelling

We finally aim to learn robust person features that are tolerant to imposter attack for open-world group-based person re-id. For further utilizing the generated person images to enhance the performance, we jointly learn feature representation and adversarial modelling in a semi-supervised way.

Although the generated images look similar to target images, they are regarded as imposter samples, and we wish to incorporate unlabeled generated imposter samples. Inspired by the smoothing regularization [43], we modify the LSRO [43] in order to make it more suitable for group-based verification by setting the probability of an unlabeled generated imposter sample $G(\boldsymbol{z})$ belonging to an existing known class k as follows:

$$q_{LSRI}(k)(G(\boldsymbol{z})) = \begin{cases} \frac{1}{C_S}, & k \in Y_S; \\ 0, & k \in Y_T, \end{cases} \tag{6}$$

Compared to LSRO, we do not allocate a uniform distribution on each unlabeled data sample over all classes (including both target and non-target ones), but only allocate a uniform distribution on non-target classes. This is significant because we attempt to separate imposter samples from target classes. The modification is exactly for the defense towards the attack of imposter samples. By using this regularization, the generated imposters are more trending to be far away from target classes and have equal chances of being non-target. We call the modified regularization in Eq. (6) as label smoothing regularization for imposters (LSRI).

Hence for each input sample $\boldsymbol{x}_i$, we set its ground truth class distribution as:

$$q(k) = \begin{cases} 1, & k = y_i \text{ and } \boldsymbol{x}_i \in X_T \cup X_S; \\ 0, & k \neq y_i \text{ and } \boldsymbol{x}_i \in X_T \cup X_S, \text{ or } \boldsymbol{x}_i \in X_G \text{ and } k \in Y_T; \\ \frac{1}{C_S}, & \boldsymbol{x}_i \in X_G, \text{ and } k \in Y_S; \end{cases} \tag{7}$$

where y_i is the corresponding label of x_i, and we let $\boldsymbol{x}_i^G$ be the ith generated image and denote $X_G = \{\boldsymbol{x}_1^G, ..., \boldsymbol{x}_{N_G}^G\}$ as the set of generated imposter samples. With Eq. (7), we can now learn together with our feature extractor (*i.e.* weights $\boldsymbol{\theta}$). By such a joint learning, the feature learning part will become more discriminative between target and target-like imposter images.

3.4 Network Structure

We now detail the network structure. As shown in Fig. 2, our network consists of two parts: 1) learning robust feature representation, and 2) learning to attack by adversarial networks. For the first part, we train the feature extractor from source datasets and generated attacking samples. In this part, features are trained to be robust and resistant to imposter samples. LSRI is applied in this part to differentiate imposters from target people. Here, a full-connected layer fc_r is connected to feature extractor f at this stage, and we call it the *feature fc* *layer*. For the second part, as shown in Fig. 2, our learning attack by adversarial networks is a modification of DCGAN [36]. We combine modified DCGAN with couple discriminators to form an adversarial network. The generator G here is modified to produce target-like imposters specifically as an attacker. And the target discriminator D_t defends as discriminating target from non-target people. Of course, in this discriminator, a new fc layer is attached to the tail of feature extractor f, and we mark it fc_t, also called *target fc layer*, used to discriminate target from non-target images at the process of learning to attack by adversarial networks. By Eq. (2), D_t is the combination of f and target fc layer fc_t.

4 Experiments

4.1 Group-based Verification Setting

We followed the criterion defined in [42] for evaluation of open-world group-based person re-id. The performance of how well a true target can be verified correctly and how badly a false target can be verified as true incorrectly is indicated by true target rate (TTR) and false target rate (FTR), which are defined as follows:

$$\textbf{True Target Rate(TTR)} = \#TTQ/\#TQ, \ \textbf{False Target Rate(FTR)} = \#FNTQ/\#NTQ, \tag{8}$$

where TQ is the set of query target images from target people, NTQ is the set of query non-target images from non-target people, TTQ is the set of query target images that are verified as target people, and $FNTQ$ is the set of query non-target images that are verified as target people.

To obtain TTR and FTR, we follow the two steps below: 1) For each target person, there is a set of images S (single-shot or multi-shot) in gallery set. Given a query sample $\boldsymbol{x}$, the distance between sample $\boldsymbol{x}$ and a set S is the minimal distance between that sample and any target sample of that set; 2) Whether a query image is verified as a target person is determined by comparing the distance to a threshold r. By changing the threshold r, a set of TTR and FTR values can be obtained. A higher TTR value is preferred when FTR is small.

In our experiments, we conducted two kinds of verification as defined in [42] namely Set Verification (*i.e.* whether a query is one of the persons in the target set, where the target set contains all target people.) and Individual Verification (*i.e.* whether a query is the true target person. For each target query image, the target set contains only this target person). In comparison, Set Verification is more difficult. Although determining whether a person image belongs to a group of target people seems easier, it also gives more chances for imposters to cheat the classifier, producing more false matchings [42].

4.2 Datasets & Settings

We evaluated our method on three datasets including Market-1501 [39], CUHK01 [19], and CUHK03 [21]. For each dataset, we randomly selected 1% people as target people and the rest as non-target. Similar to [42], for target people, we separated images of each target people into training and testing sets by half. Since only four images are available in CUHK01, we chose one for training, two for gallery (reduce to one in single-shot case) and one for probe. Our division guaranteed that probe and gallery images are from diverse cameras for each person. For non-target people, they were divided into training and testing sets by half in person/class level to ensure there is no overlap on identity. In testing phase, two images of each target person in testing set were randomly selected to form gallery set, and the remaining images were selected to form query set. In the default setting, all images of non-target people in testing set were selected to form query set. The data split was kept the same for all evaluations on our and the compared methods. Specifically, the data split is summarized below:

CUHK01 CUHK01 contains 3,884 images of 971 identities from two camera views. In our experiment, 9 people were marked as target and 1,888 images of 472 people were selected to to form the non-target training set. The testing set of non-target people contains 1,960 images of 490 people.

CUHK03 CUHK03 is larger than CUHK01 and some images were automatically detected. A total of 1,360 identities were divided into 13 target people, 667 training non-target people and 693 testing non-target people. The numbers of training and testing non-target images were 6,247 and 6,563 respectively.

Market-1501 Market-1501 is a large-scale dataset containing a total of 32,668 images of 1,501 identities. We randomly selected 15 people as target and 728 people as non-target to form the training set containing a total of 12,433 images, and the testing non-target set contains 758 identities with 13,355 images.

Under the above settings, we evaluated our model together with selected popular re-id models. Since APN is based on ResNet-50 and our evaluations attempt to show our improvement on ResNet-50, metric learning methods such as t-LRDC [42], XICE [46], XQDA [22] and CRAFT [6] are also applied to the feature extracted by ResNet-50.

4.3 Implementation Details

In our APN, we used ResNet-50 [12] as the feature extractor in the target discriminator. The generator and person discriminator is based on DCGAN [30]. At the first step of our procedure, we pre-trained the feature extractor using auxiliary datasets, 3DPeS [2], iLIDS [35], PRID2011 [13] and Shinpuhkan [15]. These datasets were only used in the pre-training stage for the feature extractor. In pre-training, we used stochastic gradient descent with momentum 0.9. The learning rate was 0.1 at the beginning and multiplied by 0.1 every 10 epochs. Then, the adversarial part of APN was trained using ADAM optimizer [16] with parameters $\beta_1 = 0.5$ and $\beta_2 = 0.99$. Using the target dataset for evaluation, the person discriminator D_p and generator G were pre-trained for 30 epochs. Then, the target discriminator D_t together with the person discriminator D_p, and the generator G were trained jointly for $k_1 = 15$ epochs, where G is optimized twice in each iteration to prevent losses of discriminators from going to zero. Finally,

Table 1: Comparison with typical person re-identification: TTR (%) against FTR

Evaluation	Market-1501						CHUK01						CUHK03					
FTR	0.1%	1%	5%	10%	20%	30%	0.1%	1%	5%	10%	20%	30%	0.1%	1%	5%	10%	20%	30%
Dataset	Set Verification																	
APN	9.01	**22.32**	**46.78**	**63.34**	**73.82**	81.12	**16.67**	**33.33**	**88.89**	**88.89**	**94.44**	**100**	**22.18**	**41.14**	**52.02**	**58.87**	**73.68**	**80.24**
ResNet-50 [12]	3.43	20.79	43.35	57.43	71.24	79.83	0	11.11	33.33	55.56	72.22	83.33	9.27	18.95	30.65	39.52	47.98	58.47
DCGAN+LSRO [43]	6.77	20.60	42.06	58.80	72.49	80.11	3.14	22.22	66.67	72.22	88.89	94.44	11.75	23.02	35.15	43.77	50.04	63.46
ResNet+t-LRDC [42]	3.00	18.88	42.06	51.07	65.24	75.54	5.56	5.56	16.67	44.44	72.22	72.22	7.74	19.35	33.01	38.81	49.18	59.45
ResNet+XICE [46]	3.28	15.75	41.63	51.07	69.53	78.11	0	11.11	16.67	44.44	72.22	81.09	5.77	13.18	20.16	37.19	45.91	57.43
ResNet+XQDA [22]	3.86	21.89	44.64	62.23	74.68	81.12	0	5.56	22.22	44.44	72.22	83.33	6.05	15.32	27.02	36.29	46.37	59.68
ResNet+CRAFT [6]	1.29	15.02	34.76	46.35	64.81	79.83	5.56	5.56	27.78	55.56	72.22	88.89	9.27	20.97	31.45	39.92	47.58	59.27
JSTL-DGD [38]	6.67	18.03	35.62	60.52	73.09	79.39	**16.67**	27.53	44.44	66.67	77.78	94.44	19.35	34.67	45.22	53.64	58.25	68.23
DeepFool [28]	**10.78**	21.89	45.05	59.23	69.53	**85.41**	**16.67**	22.22	33.33	44.44	61.11	77.78	19.35	31.45	42.74	46.37	56.05	65.32
Dataset	Individual Verification																	
APN	**32.71**	63.18	80.32	87.68	95.54	**97.30**	**41.67**	**72.22**	**97.22**	**100**	**100**	**100**	**43.13**	**58.58**	**76.14**	**83.09**	**90.08**	**93.65**
ResNet-50 [12]	30.15	61.23	77.09	85.47	91.32	97.12	33.33	44.44	72.22	86.11	97.22	**100**	39.19	53.22	70.76	77.55	86.11	92.86
DCGAN+LSRO [43]	28.14	62.77	77.29	84.09	92.96	95.11	36.11	67.56	72.22	88.89	**100**	**100**	40.72	51.18	74.91	80.58	86.36	92.06
ResNet+t-LRDC [42]	10.70	25.41	44.31	57.26	66.29	80.32	5.56	16.67	20.14	55.56	80.56	91.67	16.57	32.86	45.73	53.22	68.23	83.68
ResNet+XICE [46]	11.71	39.64	53.10	68.74	77.29	90.21	36.11	48.51	79.23	81.72	91.67	97.22	23.51	39.19	51.18	64.93	77.85	82.54
ResNet+XQDA [22]	31.03	59.10	74.51	79.39	87.63	91.42	11.11	38.89	69.44	83.33	88.89	97.22	31.89	45.36	59.15	64.38	76.44	84.87
ResNet+CRAFT [6]	22.66	62.08	78.53	83.74	91.42	97.16	11.11	33.33	63.89	80.56	88.89	97.22	30.43	43.35	57.29	68.65	80.88	84.08
JSTL-DGD [38]	27.60	52.84	70.15	77.00	87.10	90.29	25.00	50.00	86.11	91.67	97.22	**100**	37.97	50.45	67.58	76.19	82.54	87.57
DeepFool [28]	31.30	**64.43**	**83.31**	**91.89**	**95.85**	96.73	33.82	48.51	79.23	95.71	98.15	**100**	32.86	53.22	68.23	75.10	83.68	89.88

the feature extractor was trained again for $k_2 = 20$ epochs with a lower learning rate starting from 0.001 and multiplied by 0.1 every 10 epochs. The above procedure was executed repeatedly as an adversarial process.

4.4 Comparison with Open-world Re-id Methods

Open-world re-id is still under studied, and t-LRDC [42] and XICE [46] are two represented existing methods designed for the open-world setting in person re-id. Since the original works of these two existing open-world methods use traditional hand-crafted features, which are not comparable with deep learning models, we applied these two methods to ResNet-50 features for better comparison. The results are reported in Table 1. Our APN outperformed t-LRDC and XICE in all cases, and the margin is especially large on CUHK03. Compared to t-LRDC and XICE, our APN is an end-to-end learning framework and takes adversarial learning into account for feature learning, so that APN is more tolerant to the attack of samples of non-target people.

4.5 Comparison with Closed-world Re-id Methods

We compared our method with related popular re-id methods developed for closed-world person re-identification. We mainly evaluated ResNet-50 [12], XQDA [22], CRAFT [6], and JSTL-DGD [38] for comparison. These methods were all evaluated by following the same setting as our APN, where the deep features extracted by ResNet-50 were applied for all non-deep-learning methods. As shown in Table 1, these approaches optimal for closed-world scenario cannot well adapt to the open-world setting. In all cases, the proposed APN achieved overall better performance, especially when tested on Set Verification and when FTR is 1% as compared to the others. On Market-1501, APN obtained 4.29 more matching rate than the second place JSTL-DGD, a favorable deep model for re-id,

Table 2: Different generated imposter sources

Dataset	Market-1501						CUHK01						CUHK03					
FTR	0.1%	1%	5%	10%	20%	30%	0.1%	1%	5%	10%	20%	30%	0.1%	1%	5%	10%	20%	30%
Evaluation	Set Verification																	
APN	**9.01**	**22.32**	**46.78**	**63.34**	**73.82**	**81.12**	**16.67**	**33.33**	**88.89**	**88.89**	**94.44**	**100**	**22.18**	41.14	**52.02**	**58.87**	**73.68**	**80.24**
APN w/o D_t	8.58	20.60	43.78	58.80	70.82	77.68	5.56	27.78	77.78	77.78	**94.44**	94.44	17.34	**43.15**	49.19	55.24	67.34	78.23
APN w/o D_p	7.13	18.21	39.65	56.65	68.49	75.54	3.14	17.34	50.00	66.67	83.33	94.44	16.53	41.14	47.98	54.03	65.32	75.41
APN w/o WS	3.43	19.30	40.52	50.18	65.24	73.22	0	13.85	37.12	53.20	77.78	86.33	16.53	32.49	39.01	46.15	55.24	64.31
No Imposter	8.58	20.60	45.05	56.65	69.53	78.97	0	22.22	77.78	77.78	88.89	94.44	16.53	38.71	48.39	54.03	69.76	78.23
Evaluation	Individual Verification																	
APN	**32.71**	**63.18**	**80.32**	**87.68**	**95.54**	**97.30**	**41.67**	**72.22**	**97.22**	**100**	**100**	**100**	**43.13**	**58.58**	**76.14**	**83.09**	**90.08**	**93.65**
APN w/o D_t	**32.71**	61.13	75.68	85.25	90.17	95.11	35.79	70.02	95.66	**100**	**100**	**100**	41.23	55.33	75.50	81.23	87.89	92.06
APN w/o D_p	28.62	53.37	75.68	83.81	88.05	94.65	33.21	68.76	94.44	**100**	**100**	**100**	37.18	55.33	69.64	79.44	85.08	89.21
APN w/o WS	25.32	50.74	72.14	81.26	87.63	93.10	21.42	57.69	79.44	96.51	**100**	**100**	32.86	47.29	68.23	72.15	79.42	90.10
No Imposter	26.20	57.03	77.11	84.63	91.48	93.71	38.89	69.72	94.44	**100**	**100**	**100**	40.87	56.57	75.50	82.09	89.29	92.06

Table 3: Number of shots on Set Verification

Method	APN						ResNet-50					
FTR	0.1%	1%	5%	10%	20%	30%	0.1%	1%	5%	10%	20%	30%
Dataset	Market-1501											
single-shot	**5.15**	**15.45**	**41.63**	**51.93**	**64.38**	**72.10**	3.00	13.73	32.19	42.49	61.37	66.09
multi-shot	**9.01**	**22.32**	**46.78**	**63.34**	**73.82**	**81.12**	3.43	20.79	43.35	57.43	71.24	79.83
Dataset	CUHK01											
single-shot	**11.11**	**33.33**	**66.67**	**77.78**	**88.89**	**100**	0	22.22	38.89	**77.78**	**88.89**	94.44
multi-shot	**16.67**	**33.33**	**88.89**	**88.89**	**94.44**	**100**	0	22.22	77.78	77.78	88.89	94.44
Dataset	CUHK03											
single-shot	**20.97**	**38.31**	**45.56**	**52.42**	**63.31**	**69.76**	16.53	36.29	44.35	50.63	61.69	66.53
multi-shot	**22.18**	**41.14**	**52.02**	**58.87**	**73.68**	**80.24**	16.53	38.71	48.39	54.03	69.76	78.23

Table 4: Number of shots on Individual Verification

Method	APN						ResNet-50					
FTR	0.1%	1%	5%	10%	20%	30%	0.1%	1%	5%	10%	20%	30%
Dataset	Market-1501											
single-shot	**20.20**	**55.07**	**76.50**	**85.00**	**91.42**	**94.28**	15.60	39.19	62.00	71.73	82.11	88.57
multi-shot	**32.71**	**63.18**	**80.32**	**87.68**	**95.54**	**97.30**	30.15	61.23	77.09	85.47	91.32	97.12
Dataset	CUHK01											
single-shot	**39.61**	**72.22**	**91.67**	**100**	**100**	**100**	31.11	44.44	68.45	85.14	97.22	**100**
multi-shot	**41.67**	**72.22**	**97.22**	**100**	**100**	**100**	33.33	44.44	72.22	86.11	97.22	**100**
Dataset	CUHK03											
single-shot	**38.69**	**53.27**	**70.44**	**80.56**	**88.89**	**92.86**	38.39	50.84	66.96	77.38	85.31	90.87
multi-shot	**43.13**	**58.58**	**76.14**	**83.09**	**90.08**	**93.65**	39.19	53.32	70.76	77.55	86.11	92.86

when FTR is 1% on Set Verification, and as well outperformed JSTL-DGD on all conditions on Individual Verification. On CUHK01, APN gained 5.8 more matching rate as compared to JSTL-DGD when FTR is 1% and 45 matching rate more when FTR is 5% on Set Verification. The compared closed-world models were designed with the assumption that the same identities hold between gallery and probe sets, while the relation between target and non-target people is not modelled. Meanwhile our APN is designed for the open-world group-based verification for discriminating target from non-target people.

4.6 Comparison with Related Adversarial Generation

We compared our model with fine-tuned ResNet-50 with adversarial samples generated by DeepFool [28], which is also a method using extra generated samples. DeepFool produced adversarial samples to fool the network by adding noise computed by gradient. As shown in Table 1, our APN performed much better than DeepFool especially on CUHK01 and CUHK03. DeepFool cannot adapt to open-world re-id well because the adversarial samples generated are produced with a separate learning from the classifier learning and thus the relation between the generated samples and target set is not modelled for group-based verification, while in our APN we aim to generate target-like samples so as to make adversarial learning facilitate learning better features.

We also evaluated ResNet-50 trained with samples generated by DCGAN and using LSRO as done in [43]. And APN outperformed it in all cases. The work of [43] only used generated samples to enlarge the dataset and the group-based verification for open-world re-id is not taken into consideration.

Table 5: Different target proportion of Market-1501 on Set Verification (TP. stands for Target Proportion)

Method	APN						ResNet-50					
FTR	0.1%	1%	5%	10%	20%	30%	0.1%	1%	5%	10%	20%	30%
TP. 0.5%	**25.35**	**43.74**	**66.39**	**77.89**	**86.54**	**93.79**	14.48	37.96	61.25	73.50	82.85	90.62
TP. 1%	**9.01**	**22.32**	**46.78**	**59.23**	**73.82**	**81.12**	3.43	18.79	40.35	57.43	70.24	77.83
TP. 3%	**5.66**	**17.31**	**35.11**	**46.76**	**59.55**	**69.58**	4.50	15.34	31.63	44.50	58.22	67.55
TP. 5%	**5.38**	**15.77**	**31.36**	**42.11**	**56.43**	**68.09**	5.02	12.31	30.08	39.02	53.97	64.99

Table 6: LSRI vs. LSRO

Evaluation	Set Verification						Individual Verification					
FTR	0.1%	1%	5%	10%	20%	30%	0.1%	1%	5%	10%	20%	30%
Dataset	Market-1501											
LSRI	**9.01**	**22.32**	**46.78**	**63.34**	**73.82**	**81.12**	**32.71**	**63.18**	**80.32**	**87.68**	**95.54**	**97.30**
LSRO [43]	8.15	20.60	45.92	57.51	72.53	79.83	28.42	61.66	78.80	84.61	90.56	92.54
Dataset	CUHK01											
LSRI	**16.67**	**33.33**	**88.89**	**88.89**	**94.44**	**100**	**41.67**	**72.22**	**97.22**	**100**	**100**	**100**
LSRO [43]	**16.67**	27.78	77.78	**88.89**	88.89	94.44	36.11	68.23	**97.22**	**100**	**100**	**100**
Dataset	CUHK03											
LSRI	**22.18**	**41.14**	**52.02**	**58.87**	**73.68**	**80.24**	**43.13**	**58.58**	**76.14**	**83.09**	**90.08**	**93.65**
LSRO [43]	17.34	39.52	49.60	55.24	69.76	79.03	39.01	52.53	71.50	80.13	88.49	91.27

4.7 Further Evaluation of Our Method

Effect of Generated Imposters. We compared to the case without using the generated imposters. We trained our network in the same way without inputting the generated images in the training of feature extractor. The results are shown in the rows indicated by "No Imposters" in Table 2. It can be observed that, training with the imposters generated by APN can achieve large improvement as compared to the case without it, because these imposters are target-like and can improve the discriminating ability of the features. In details, on Set Verification, APN outperformed an average of 2.15 matching rate on Market-1501 and 3.23 matching rate on CUHK03, and for CUHK01 APN outperformed 16.67% when FTR is 0.1%. On Individual Verification, APN outperformed an average of 4.43 matching rate on Market-1501 and has better performance on all other cases.

Effect of Weight Sharing. The weight sharing between the target discriminator and the feature extractor aims to ensure that the generator can learn from the feature extractor and generate more target-like attack samples. Without the sharing, there is no connection between generation and feature extraction. Taking Individual Verification on Market for instance, ours degrades from 63.18% to 50.74% (no sharing indicated by "APN w/o WS") when FTR=1% in Table 2.

Effect of Person Discriminator and Target Discriminator. Our APN is based on GAN consisting of generator, person discriminator D_p and target discriminator D_t. To further evaluate them, we compared with APN without person discriminator (APN w/o D_p) and APN without target discriminator (APN w/o D_t). APN without target discriminator can be regarded as two independent components DCGAN and feature extraction network. To fairly compare these cases, LSRI was also applied as in APN for the generated samples. The results are reported in Table 2. It is obvious that our full APN is the most effective one among the compared cases. Sometimes generating imposters by APN without person discriminator D_p or target discriminator D_t even degrade the performance as compared to the case of no imposter. When target discriminator is discarded, although person-like images can be generated, they are not similar to target people and thus are not serious attacks to the features for group-based verification. In the case without person discriminator, the generator even fails to generate person-like images (see Fig. 3) so that the performance is largely degraded. This indicates that the person discriminator plays an important role in generating person-like images, and the target discriminator is significant for helping the generator to generate better target-like imposters, so that the feature extractor can benefit more from distinguishing these imposters.

LSRI vs LSRO. We verified that the modification of LSRO, namely LSRI in Eq. (6) is more suitable for optimizing the open-world re-id. The performance of comparing our LSRI with the original LSRO is reported in Table 6. It shows that the feature extractor is more likely to correctly discriminate target people under the same FTR using LSRI on. It is proved that our modification LSRI is more appropriate for open-world re-id scenario, since the imposters are allocated equal probabilities of being non-target for group-based towards modelling, so they are more likely to be far away from target person samples, leading to more discriminative feature representation for target people, while in LSRO, the imposters are allocated equal probabilities of being non-target as well as target.

Effect of target proportion. The evaluation results on different target proportion are reported in Table 5. We used different percentages of people marked as target. This verification was conducted on Market-1501, and we used original ResNet-50 for comparison. While TTR declines with the growth of target proportion due to more target people to verify, our APN can still outperformed the original ResNet-50 in all cases.

Effect of the Number of Shots. The performance under multi-shot and single-shot settings were also compared in our experiments. For multi-shot setting, we randomly selected two images of each target person as gallery set, while for single-shot setting, we only selected one. As shown in Table 3 and Table 4, on both single-shot and multi-shot settings, our APN outperformed ResNet-50 on all conditions of Market-1501, CUHK01, and CUHK03. Especially on Set Verification, for CUHK01, when FTR is 0.1%, APN outperformed ResNet-50 11.11% under single-shot setting and 16.67% under multi-shot setting.

5 Conclusion

For the first time, we demonstrate how adversarial learning can be used to solve the open-world group-based person re-id problem. The introduced adversarial person re-id enables a mutually related and cooperative progress among learning to represent, learning to generate, learning to attack, and learning to defend. In addition, this adversarial modelling is also further improved by a label smoothing regularization for imposters under semi-supervised learning.

Acknowledgment

This work was supported partially by the National Key Research and Development Program of China (2016YFB1001002) and the NSFC(61522115, 61472456), Guangdong Programme (2016TX03X157), and the Royal Society Newton Advanced Fellowship (NA150459). The corresponding author for this paper is Wei-Shi Zheng.

References

1. Ahmed, E., Jones, M.J., Marks, T.K.: An improved deep learning architecture for person re-identification. In: CVPR. IEEE Computer Society (2015)
2. Baltieri, D., Vezzani, R., Cucchiara, R.: 3dpes: 3d people dataset for surveillance and forensics. In: Proceedings of the 1st International ACM Workshop on Multimedia access to 3D Human Objects. pp. 59–64. Scottsdale, Arizona, USA (Nov 2011)
3. Bedagkar-Gala, A., Shah, S.K.: A survey of approaches and trends in person re-identification. Image Vision Comput. **32**(4), 270–286 (2014)
4. Cancela, B., Hospedales, T.M., Gong, S.: Open-world person re-identification by multi-label assignment inference. In: Valstar, M.F., French, A.P., Pridmore, T.P. (eds.) BMVC. BMVA Press (2014)
5. Chen, Y.C., Zheng, W.S., Lai, J.H., Yuen, P.C.: An asymmetric distance model for cross-view feature mapping in person reidentification. IEEE Trans. Circuits Syst. Video Techn. **27**(8), 1661–1675 (2017)
6. Chen, Y.C., Zhu, X., Zheng, W.S., Lai, J.H.: Person re-identification by camera correlation aware feature augmentation. CoRR **abs/1703.08837** (2017)
7. Cheng, D., Gong, Y., Zhou, S., Wang, J., Zheng, N.: Person re-identification by multi-channel parts-based cnn with improved triplet loss function. In: CVPR. IEEE Computer Society (2016)
8. Deng, W., Zheng, L., Ye, Q., Kang, G., Yang, Y., Jiao, J.: Image-image domain adaptation with preserved self-similarity and domain-dissimilarity for person reidentification. In: CVPR. p. 6. IEEE Computer Society (2018)
9. Ding, S., Lin, L., Wang, G., Chao, H.: Deep feature learning with relative distance comparison for person re-identification. CoRR **abs/1512.03622** (2015)
10. Farenzena, M., Bazzani, L., Perina, A., Murino, V., Cristani, M.: Person re-identification by symmetry-driven accumulation of local features. In: CVPR. IEEE Computer Society (2010)
11. Gong, S., Cristani, M., Yan, S., Loy, C.C. (eds.): Person Re-Identification. Advances in Computer Vision and Pattern Recognition, Springer (2014)
12. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: CVPR. IEEE Computer Society (2016)
13. Hirzer, M., Beleznai, C., Roth, P.M., Bischof, H.: Person re-identification by descriptive and discriminative classification. In: SCIA. Springer (2011)
14. Hirzer, M., Roth, P.M., Köstinger, M., Bischof, H.: Relaxed pairwise learned metric for person re-identification. In: ECCV. Lecture Notes in Computer Science (2012)
15. Kawanishi, Y., Wu, Y., Mukunoki, M., Minoh, M.: Shinpuhkan2014: A multi-camera pedestrian dataset for tracking people across multiple cameras
16. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. CoRR **abs/1412.6980** (2014)
17. Kviatkovsky, I., Adam, A., Rivlin, E.: Color invariants for person reidentification. IEEE Trans. Pattern Anal. Mach. Intell. **35**(7), 1622–1634 (2013)
18. Köstinger, M., Hirzer, M., Wohlhart, P., Roth, P.M., Bischof, H.: Large scale metric learning from equivalence constraints. In: CVPR. IEEE Computer Society (2012)
19. Li, W., Zhao, R., Wang, X.: Human reidentification with transferred metric learning. In: ACCV (2012)
20. Li, W., Zhao, R., Xiao, T., Wang, X.: Deepreid: Deep filter pairing neural network for person re-identification. In: CVPR. IEEE Computer Society (2014)

21. Li, W., Zhao, R., Xiao, T., Wang, X.: Deepreid: Deep filter pairing neural network for person re-identification. In: CVPR (2014)
22. Liao, S., Hu, Y., Zhu, X., Li, S.Z.: Person re-identification by local maximal occurrence representation and metric learning. In: CVPR. pp. 2197–2206. IEEE Computer Society (2015)
23. Liu, C., Gong, S., Loy, C.C.: On-the-fly feature importance mining for person re-identification. Pattern Recognition **47**(4), 1602–1615 (2014)
24. Lu, J., Issaranon, T., Forsyth, D.A.: Safetynet: Detecting and rejecting adversarial examples robustly. CoRR **abs/1704.00103** (2017)
25. Ma, B., Su, Y., Jurie, F.: Bicov: a novel image representation for person re-identification and face verification. In: BMVC (2012)
26. Ma, B., Su, Y., Jurie, F.: Covariance descriptor based on bio-inspired features for person re-identification and face verification. Image Vision Comput. **32**(6-7), 379–390 (2014). https://doi.org/10.1016/j.imavis.2014.04.002
27. Mignon, A., Jurie, F.: Pcca: A new approach for distance learning from sparse pairwise constraints. In: CVPR. IEEE Computer Society (2012)
28. Moosavi-Dezfooli, S.M., Fawzi, A., Frossard, P.: Deepfool: A simple and accurate method to fool deep neural networks. In: CVPR. IEEE Computer Society (2016)
29. Papernot, N., McDaniel, P.D., Jha, S., Fredrikson, M., Celik, Z.B., Swami, A.: The limitations of deep learning in adversarial settings. CoRR **abs/1511.07528** (2015)
30. Radford, A., Metz, L., Chintala, S.: Unsupervised representation learning with deep convolutional generative adversarial networks. CoRR **abs/1511.06434** (2015)
31. Subramaniam, A., Chatterjee, M., Mittal, A.: Deep neural networks with inexact matching for person re-identification. In: Lee, D.D., Sugiyama, M., von Luxburg, U., Guyon, I., Garnett, R. (eds.) NIPS (2016)
32. Szegedy, C., Zaremba, W., Sutskever, I., Bruna, J., Erhan, D., Goodfellow, I.J., Fergus, R.: Intriguing properties of neural networks. CoRR **abs/1312.6199** (2013)
33. Tao, D., Jin, L., Wang, Y., Yuan, Y., Li, X.: Person re-identification by regularized smoothing kiss metric learning. IEEE Trans. Circuits Syst. Video Techn. **23**(10), 1675–1685 (2013)
34. Wang, H., Zhu, X., Xiang, T., Gong, S.: Towards unsupervised open-set person re-identification. In: ICIP. IEEE (2016)
35. Wang, T., Gong, S., Zhu, X., Wang, S.: Person re-identification by discriminative selection in video ranking. IEEE transactions on pattern analysis and machine intelligence **38**(12), 2501–2514 (2016)
36. Wu, L., Shen, C., van den Hengel, A.: Personnet: Person re-identification with deep convolutional neural networks. CoRR **abs/1601.07255** (2016)
37. Wu, S., Chen, Y.C., Li, X., Wu, A., You, J., Zheng, W.S.: An enhanced deep feature representation for person re-identification. CoRR **abs/1604.07807** (2016)
38. Xiao, T., Li, H., Ouyang, W., Wang, X.: Learning deep feature representations with domain guided dropout for person re-identification. In: CVPR. IEEE Computer Society (2016)
39. Zheng, L., Shen, L., Tian, L., Wang, S., Wang, J., Tian, Q.: Scalable person re-identification: A benchmark. In: ICCV (2015)
40. Zheng, W.S., Gong, S., Xiang, T.: Person re-identification by probabilistic relative distance comparison. In: CVPR. IEEE Computer Society (2011)
41. Zheng, W.S., Gong, S., Xiang, T.: Transfer re-identification: From person to set-based verification. In: CVPR. IEEE Computer Society (2012)
42. Zheng, W.S., Gong, S., Xiang, T.: Towards open-world person re-identification by one-shot group-based verification. IEEE Trans. Pattern Anal. Mach. Intell. **38**(3), 591–606 (2016)

43. Zheng, Z., Zheng, L., Yang, Y.: Unlabeled samples generated by gan improve the person re-identification baseline in vitro. CoRR **abs/1701.07717** (2017)
44. Zhong, Z., Zheng, L., Zheng, Z., Li, S., Yang, Y.: Camera style adaptation for person re-identification. In: CVPR. pp. 5157–5166. IEEE Computer Society (2018)
45. Zhu, J., Zeng, H., Liao, S., Lei, Z., Cai, C., Zheng, L.: Deep hybrid similarity learning for person re-identification. CoRR **abs/1702.04858** (2017)
46. Zhu, X., Wu, B., Huang, D., Zheng, W.S.: Fast open-world person re-identification. IEEE Transactions on Image Processing **PP**(99), 1–1 (2017). https://doi.org/10.1109/TIP.2017.2740564