Frustratingly Easy Person Re-Identification: Generalizing Person Re-ID in Practice

Jieru Jia[1]
12112059@bjtu.edu.cn
Qiuqi Ruan[1]
qqruan@bjtu.edu.cn
Timothy M. Hospedales[2]
t.hospedales@ed.ac.uk

[1] Institute of Information Science
Beijing Jiaotong University
Beijing, China

[2] Institute of Perception, Action and Behaviour
University of Edinburgh,
Edinburgh, UK

Abstract

Contemporary person re-identification (Re-ID) methods usually require access to data from the deployment camera network during training in order to perform well. This is because contemporary Re-ID models trained on one dataset do not generalise to other camera networks due to the domain-shift between datasets. This requirement is often the bottleneck for deploying Re-ID systems in practical security or commercial applications, as it may be impossible to collect this data in advance or prohibitively costly to annotate it. This paper alleviates this issue by proposing a simple baseline for domain generalizable (DG) person re-identification. That is, to learn a Re-ID model from a set of source domains that is suitable for application to unseen datasets out-of-the-box, without any model updating. Specifically, we observe that the domain discrepancy in Re-ID is due to style and content variance across datasets and demonstrate appropriate Instance and Feature Normalization alleviates much of the resulting domain-shift in Deep Re-ID models. Instance Normalization (IN) in early layers filters out style statistic variations and Feature Normalization (FN) in deep layers is able to further eliminate disparity in content statistics. Compared to contemporary alternatives, this approach is extremely simple to implement, while being faster to train and test, thus making it an extremely valuable baseline for implementing Re-ID in practice. With a few lines of code, it increases the rank 1 Re-ID accuracy by 11.8%, 33.2%, 12.8% and 8.5% on the VIPeR, PRID, GRID, and i-LIDS benchmarks respectively. Source codes are available at https://github.com/BJTUJia/person_reID_DualNorm.

1 Introduction

Person re-identification (Re-ID) aims to match pedestrian images captured by different cameras at different times and locations. Despite their goal of cross-camera matching, most contemporary Re-ID methods overfit to specific datasets (camera networks) in that, once trained on a given dataset, they perform poorly if applied to a different camera network. This prevents a single Re-ID system from being built that can successfully apply off-the-shelf to diverse scenarios. Contemporary research focuses on supervised Re-ID, where systems are

trained with annotated data from the deployment network [7, 51] – at a significant cost to scalability; or unsupervised domain adaptation (UDA) Re-ID [10, 18, 45, 56], which aims to alleviate the annotation cost by adapting a model trained on annotated source datasets (whose annotation cost is then amortised) to unannotated target datasets. While UDA methods reduce annotation costs, they still require data collection for the target network prior to model adaptive training. Since it may not be known in advance where a Re-ID model should be deployed, these are not ideal for real-world applications.

In this paper, we consider the most practically valuable problem setting of domain generalization (DG) Re-ID, in which we train using only labeled source dataset images, without touching any target network images – either labeled or unlabelled. During testing the model is applied to novel unseen datasets with no adaptation. This setting simulates the ideal scenario in which a strong learner is trained once in the lab, and can then be deployed to diverse camera networks in the wild with no further data collection or adaptive training required. This DG setting is the most practically relevant for real-world commercial/security applications. However due to its challenging nature, few contemporary deep-learning-based methods have attempted the DG setting, besides the recent DIMN [40]. DIMM is effective but requires a complicated meta-learning procedure for training and dynamic model-synthesis at testing-time that makes it relatively slow and cumbersome for practical applications. An easy-to-implement, fast, and stable DG Re-ID method is still needed in practice.

Our goal is to train a model that generalizes well to unseen domains out-of-the-box – recognising new identities in a novel camera network. Our *DualNorm* solution explicitly addresses domain biases via normalization in deep feature extraction. We observe that the discrepancy between diverse Re-ID domains arises from two sources: (i) difference in style, *e.g.*, distinct color saturations and contrasts, lighting, resolutions, clothing styles, seasons; and (ii) content variation caused by differing typical distance to objects, focal length, viewpoint and typical poses. To alleviate these issues, we design a CNN architecture that jointly normalizes style and content statistics. Specifically, we use Instance Normalization (IN) layers at each bottleneck in shallow layers to capture and eliminate style variations. We next normalise different content statistics by performing Batch Normalization (BN)-based normalisation of extracted features. By jointly tackling style and content variances to explicitly address domain bias, extracted features are more universal and domain agnostic.

Our method is extremely simple compared to elaborate contemporary alternatives [18, 40, 56]. However, we argue that this should be considered a plus in practice where ease of implementation, stability, and efficiency are more valuable than elaborate methodologies. Furthermore, we point to a distinguished history of methods in vision and pattern recognition, where simple baselines that surpassed their more elaborate predecessors made significant impact [8, 20, 37].

To summarize, the main contributions of this paper are: (1) A strong baseline for domain generalization person Re-ID. Our method requires no information about the target domain during training, and works well as an out-of-the-box feature extractor for novel camera networks. It surpasses most existing (target domain)-supervised methods, and all existing methods that do not learn on target data. (2) We present a simple yet highly efficient way to address domain-shift through normalization layers: IN in shallow layers and BN in deep layers alleviate style and content statistic domain biases. (3) Our approach is very easy to implement (with just a few lines of code) and fast to run, but boosts DG Re-ID performance by a large margin for both MobileNet and ResNet backbones. Its simplicity, efficacy and efficiency make it appealing for Re-ID in practice.

2 Related Work

Domain adaption and generalization Unsupervised Domain Adaptation (UDA) alleviates domain-shift without recourse to *annotated* data in the target domain [5]. For example, by reducing the Maximum Mean Discrepancy (MMD) [15] between domains [48], or training an adversarial domain-classifier [43] to make different domains indistinguishable. In the Re-ID community, the UDA methods typically resort to image-synthesis [4, 10] or focus on source-target domain alignment [18, 45, 56]. While these methods are annotation efficient, they do require prior collection of target-domain data for training, while our method has no such requirement, making it more valuable in practice where the deployment network is not known at the time of model creation.

Compared to UDA[10, 18], Domain Generalisation (DG) methods [27] aim to create models that are robust-by-design to domain-shift between training and testing. These methods tend to leverage architectures specially designed for domain-shift robustness [13, 21], or propose meta-learning procedures for standard architectures [8, 22, 27]. Our method is in the former category, but only requires a minor modification of standard Re-ID architectures. In a Re-ID context, we are only aware of DIMN [40] as a contemporary attempt at the DG problem setting, which uses a meta-learning approach[1]. While DIMN is effective, it requires a complicated and slow meta-learning procedure for training, which limits its appeal to practitioners. Furthermore, DIMN uses dynamic model synthesis at runtime so it is not amenable to modifications for runtime scalability such as binarization, approximate nearest-neighbour search, and hashing. In contrast our carefully designed feature extractor is faster out-of-the-box, and can potentially be extended in all of these ways.

Normalization Batch Normalization (BN) [19] has become a key technique in CNN training, by standardizing input data or activations using statistics computed over examples in a mini-batch. Instance Normalization (IN) [44] performs BN-like computation over a single sample. Moreover, the IBN building block recently proposed in [34] enhances models' generalization ability by integrating IN and BN. A different way to combine BN and IN was put forward in [32]. Recently, some effort has been made in feature normalization [47, 54], mainly applying l_2-norm to the feature embeddings, constraining them to the unit circle.

Difference from previous works The normalization techniques we explore have been largely presented in existing works. The contribution of our work lies in the following aspects: (1) We explore the feasibility of IN in inverted residual bottlenecks in MobileNetV2 [38] to overcome the style variances in Re-ID. (2) Although BN layers after feature extraction are widely used, we explicitly utilize them as a means of feature normalization for domain-shift robustness. (3) We systematically demonstrate that, by integrating the two normalization techniques into a unified network with end-to-end learning, our DualNorm provides the strongest baseline for person Re-ID, while being simple and requiring no target domain data.

3 Method

Setup For domain generalization person re-ID, we assume K labeled source datasets $D = \{D_1, D_2, \ldots D_K\}$. Each source $D_i = \left\{X^{(i)}, Y^{(i)}\right\}$ consists of image-label pairs, where $y_i \in$

[1] Of course classic feature-engineering approaches [12] are not tied to specific datasets, but these are not competitive with contemporary deep-learning based approaches.

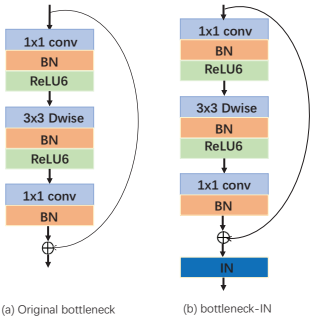

Figure 1: The structure of our MobileNetV2 bottlenecks with added Instance normalization (we use the block with stride=1 as an example).

$\{1,2,\ldots M_i\}$ and M_i is the number of identities in D_i [40]. Since the label spaces for K source sets are disjoint, we use their union as label space with $N = \sum_i M_i$ identities in total.

A Strong Cross-domain Baseline Following [40], we build a strong starting model by aggregating the labeled images from multiple source domains, and training a single model to discriminate all N identities. During the testing phase, the learnt model is used as an off-the-shelf feature extractor on previously unseen identities. We choose lightweight MobileNetV2 [38] as backbone, since it has significantly fewer parameters compared to other CNN architectures like ResNet [16]. We keep the default structure of MobileNetV2 except for changing the dimension of last classification layer (FC) to be N, i.e., the total number of identities. We use the cross-entropy (CE) loss to calculate the loss of all source domains:

$$L_{CE} = -\frac{1}{n_s}\sum_{i=1}^{n_s} \log p\left(y_i|x_i\right) \tag{1}$$

where n_s is the number of images in a mini-batch and $p\left(y_i|x_i\right)$ is the predicted probability that the source image x_i belongs to identity y_i. It is worth mentioning the mini-batch scheduling: each mini-batch contains samples randomly selected from all source domains. The model is trained from scratch, without pre-training on ImageNet [9].

Testing During testing phase, given an input image from target domain, we extract the 1280-d pooling layer activations as output features. Then, we use Euclidean distance to perform person retrieval in the target set. Next, we will discuss how to improve this naïve approach to achieve better generalization through normalizing dataset-specific biases.

3.1 Style Normalization

The structure of our bottleneck with IN layer is shown in Fig. 1. In modern deep networks, Batch Normalization (BN) [19] has become a standard operator with its ability to stabilize and accelerate training. On the other hand, Instance Normalization (IN) [42], which is mainly used in style transfer tasks [32, 34], normalizes feature responses at the instance-level with the statistics of a single sample. It is assumed that the styles of images are encoded in the first-order statistics, i.e. the mean and variance of a convolutional feature map [32]. Therefore, by normalizing the output of the original inverted residual bottlenecks, we filter out instance specific style variances, which makes the learned features more dataset-agnostic.

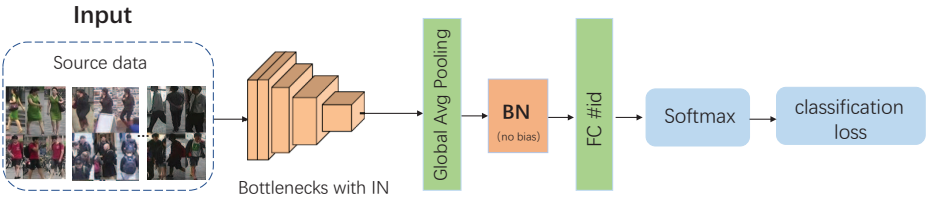

Figure 2: Structure of our DualNorm network. Input images are randomly sampled from multiple source sets, and go forward through the stacked bottlenecks with IN. After global average pooling, a BN with bias removed is adopted to normalize the features. They are then fed into a classifier with FC and softmax layer.

According to the findings in [34], the appearance divergence caused by style variances mostly lies in early layers and adding IN in early layers can effectively reduce the domain-shift due to style differences. On the other hand, applying IN to deep layers may cause severe loss of discriminative information and degrade the classification performance. Therefore, we only add IN for the shallow layers of the network.

3.2 Content Normalization

Summary The feature norm methods in [47, 54] help feature robustness by introducing smoothness with l_2-norm, which is also the fundamental effect of BN, as explained in [39]. Hence, we exploit BN as an alternative feature norm approach. The exploitation of BN in our overall framework is shown in Fig. 2. In this case, BN is applied to the penultimate layer after feature encoding.

Details Let F and G denote the feature extraction module and classifier layer respectively. The input images $\{x_n\}_{n=1,...N}$ are passed through F to acquire the unnormalized feature $F(x_n) \in R^{B\times C\times H\times W}$, where H and W indicate spatial location, C is the number of channels and B denotes the number of examples in the mini-batch. With a global average pooling layer, $F(x_n)$ is transformed to $f(x_n) \in R^{B\times C\times 1\times 1}$. Next, BN normalizes each channel of $f(x_n)$ using the mean μ and variance σ^2 computed over the mini-batch:

$$BN(f(x_n)) = \gamma\left(\frac{f(x_n)-\mu}{\sigma}\right)+\beta \tag{2}$$

This causes the feature for a particular instance to depend on other instances in the mini-batch and helps regularize the network training. Another advantage of this BatchNorm is that it makes gradients more stable and enables larger learning rates, thus yielding faster convergence and better generalization.

Note that different to vanilla BN, we fix the bias to $\beta = 0$ following [30]. This is because the softmax loss is angular and the hypersphere is nearly symmetric about the origin of the coordinate axis. By removing the bias parameter, the features are constrained around the origin of the coordinate axis. Accordingly, the bias of the following FC layer is also removed.

Discussion By using the proposed DualNorm mechanism, we are able to push the learned feature space towards domain-invariant and better generalizable to novel domains. More specifically, we assume that the domain discrepancy in Re-ID mainly stems from difference

in image style and content. IN in early layers can explicitly normalizes domain-specific image styles, like color, lighting, clothing style *et al*. BN after feature extractor is able to normalize the disparity in content statistics caused by differing camera angle, body size, viewpoint, pose *et al*. Using the style and content normalisations outlined above, our feature extractor alleviates much of the domain-specific biases that cause the domain-shift problem when porting re-id models across datasets. Therefore, the learned model is more likely to be domain agnostic and own better generalization ability.

The added normalization layers only introduce negligible additional parameters, taking up a very small percentage of full network parameters. The additional parameters are learned jointly with the other parameters of the network in an end-to-end manner, requiring very limited extra computation cost and GPU memory. Despite its simplicity, our method achieves the new state-of-the-art cross-domain re-ID accuracy. Thus it makes a major contribution towards making person Re-ID easy to implement and deploy.

4 Experiments

4.1 Datasets and Settings

Datasets To evaluate the proposed method, we follow [40] and combine existing large-scale Re-ID datasets to form the source domains and test the performance on several smaller target datasets. This is to reflect the most desirable practical scenario where a practitioner would use the largest available existing datasets to train a deep Re-ID model, and hope that it can be applied off-the-shelf to a novel camera network. To be specific, we exploit Market-1501 [52], DukeMTMC-reID [55], CUHK02 [24], CUHK03 [25] and PersonSearch [42] as the source datasets, with a a total of 18,530 training identities and 121,765 training images in all. We denote this large scale re-ID dataset collection as 'MS' (multi-source).

The target datasets include VIPeR[14], PRID[17], QMUL GRID [29] and i-LIDS [53]. For the convenience of comparing with prior reported results, we use the standard evaluation protocols on the testing datasets. More specifically, we follow the single-shot setting with the number of probe/gallery images set as: VIPeR: 316/316; PRID: 100/649; GRID: 125/900; i-LIDS: 60/60 respectively. Note that for i-LIDS, two images per identity are randomly selected as gallery and probe respectively, following the settings in [40]. For all the testing datasets, the average results over 10 random splits are reported.

Implementation Details We use MobileNetV2 [38] with width multiplier of 1.0 as the backbone network. In terms of input images, we keep the aspect ratio and resize them to 256×128. Random cropping and random flipping are applied as data augmentation. We implement our model with PyTorch [35] and train it on a single Titan X GPU. The model is trained from scratch with initial learning rate as 0.01 for all the layers. The SGD optimizer is exploited in a total of 150 epochs and the learning rate is divided by 10 after 100 epochs. The mini-batch size is set to 64. The only trick we use is label smoothing [30, 55], which is helpful to prevent the model from overfitting to training IDs.

Competitors We compare our proposed DualNorm method with a variety of alternatives. **'DG' Methods:** In terms of domain generalisation approaches, we compare with the domain aggregation baseline **AGG** (our method prior to adding normalisation layers) and state-of-the-art method DIMN [40]. **'U' Methods:** For unsupervised methods, we compare with UDA methods MMFA [28], TJ-AIDL [45], SyRI[4], and a representative selection of other

Method	Type	Source	VIPeR	PRID	GRID	i-LIDS
MMFA [28]	U	Market	39.1	35.1	-	-
MMFA [28]	U	Duke	36.3	34.5	-	-
TJ-AIDL[45]	U	Market	38.5	26.8	-	-
TJ-AIDL[45]	U	Duke	35.1	34.8	-	-
SyRI[4]	U	R+S[1]	43.0	43.0	-	56.5
CAMEL [49]	U	JSTL [46]	30.9	-	-	-
UMDL[36]	U	Comb [2]	31.5	24.2	-	49.3
SSDAL [41]	U	PETA [11]	37.9	20.1	19.1	-
OneShot [2]	S	Target	34.3	41.4	-	51.2
NFST [50]	S	Target	51.2	40.9	-	-
Ensembles [33]	S	Target	45.9	17.9	-	50.3
DSPSL [23]	S	Target	-	-	-	55.2
MTDnet [6]	S	Target	47.5	32.0	-	58.4
ImpTrpLoss [7]	S	Target	47.8	22.0	-	60.4
GOG+XQDA [31]	S	Target	49.7	**68.4**	24.7	-
JLML [26]	S	Target	50.2	-	37.5	-
SSM [1]	S	Target	53.7	-	27.2	-
SpindleNet [51]	S	Target	53.8	67.0	-	66.3
AGG (DIMN)[40]	DG	MS	42.9	38.9	29.7	69.2
AGG (Ours)	DG	MS	42.1	27.2	28.6	66.3
DIMN [40]	DG	MS	51.2	39.2	29.3	70.2
DualNorm(Ours)	DG	MS	**53.9**	60.4	**41.4**	**74.8**

Table 1: Comparison against state-of-the-art on VIPeR, PRID, GRID and i-LIDS (Rank 1 accuracy). 'U' and 'S' denote (U)nsupervised and (S)upervised use of the target dataset for training. 'DG' methods do not touch the target during training. '-': unreported result.

unsupervised alternatives. We also compare a representative selection of recent supervised (**'S'**) methods – that use target images and labels. It is important to note that the UDA, and S methods are not fair competitors in that they use more information about the target domain than ours. We include them not as direct competitors, but to contextualise our results.

4.2 Comparisons Against State-of-the-Art

The rank 1 accuracy on VIPeR, PRID, GRID and i-LIDS are listed in Table 1. Models in the Unsupervised (U) setting are trained with some source data and then adapted to the target datasets using unlabeled images form their training split, while models in the Supervised (S) setting are trained with data and labels from the target dataset's training split. From Table 1, the following observations can be made: (1) The naïve aggregation solution provides a strong baseline. The use of a large amount of source data provides a stronger baseline than most existing unsupervised adaptation methods, even though the target data is not used at all. (2) Our approach significantly and consistently outperforms most Unsupervised and Supervised competitors, despite using neither images nor labels from the target domain. The ability to outperform recent fully-supervised approaches such as SpindleNet is noteworthy since we

[1] 'R' include CUHK03+Duke while 'S' means Synthetic images.

[2] Four out of five datasets VIPeR, PRID, CUHK01, i-LIDS and CAVIAR are combined as the source datasets when the fifth one is chosen as the target.

Components	VIPeR	PRID	GRID	i-LIDS
Baseline	42.1	27.2	28.6	66.3
+IN	52.7	54.1	39.7	70.5
+FN	48.1	47.2	31.3	71.2
DualNorm	53.9	60.4	41.4	74.8

Table 2: Ablation study on the impact of different components for cross-domain Re-ID.

use neither the target data nor labels that they use. This shows that sufficient source data volume, and appropriate normalisation for debiasing can eventually alleviate the need for target-domain data in achieving state-of-the-art performance. This is a crucially important practical result, because it means that a model can be trained once and then re-deployed to different locations without re-training – which is crucial for making Re-ID deployment practical. (3) Our approach significantly and consistently outperforms DIMN [40], the only purpose-designed DG competitor. This is despite the fact that our method is significantly simpler and more computationally efficient.

4.3 Additional Analysis

The effect of IN and FN There are two important components in our framework: IN in early bottlenecks to alleviate style variance and FN on the penultimate layer to normalize content. To evaluate the contribution of each component, we separately add IN and FN to the backbone network and compare the performance in Table 2. We can see that both IN and FN contribute notably to the improvement of overall performance. Among them, adding IN in early layers seems more beneficial, and combining two normalization techniques leads to a further performance gain, proving that they are complementary. Overall, our approach boosts the rank 1 rates of **AGG** by 11.8%, 33.2%, 12.8%, 8.5% respectively, only with extra normalization layers that introduce few paremeters.

On the location and amount of IN We apply FN at one fixed location (after feature pooling), while the best place to apply IN is unclear. We thus investigate where to apply IN. We denote the initial convolution block in MobileNetV2 [38] as 'Conv1'. The following groups of inverted residual bottlenecks are referred to as 'Conv2_x-Conv8_x' respectively.

Table 3 gives performance of IN layers added to various locations in the original MobileNetV2. Note that for all the experiments here, the feature norm is removed to shed light on IN's influence. It can be seen that: (1) adding more IN layers in early layers increases the performance more, but IN applied to deep layers has a detrimental effect. This indicates that a moderate amount of IN operations in shallow layers suffices. (2) Comparing '1-6' and '2-6', we can see the instance normalization on Conv1 is critical since it directly takes the activation of the first convolution layer as input, where appearance differences are intrinsic.

Comparison with other feature norm approaches To investigate the effect of the added BN as feature normalization, we compare it with other recently proposed feature norm methods in Table 4. Here, IN is added to Conv1-6 for all the experiments and we follow the default parameter settings in each paper for the compared methods. We observe that alternative methods such as HAFN [47] tend to deteriorate the overall performance. The possible reasons for the performance drop are: i) the additional hyper-parameters in [47, 54] require careful manual tuning; ii) the l_2-norm of the feature embeddings may hurt the generalization capacity and don't cooperate well with IN in early layers. On the contrary, our approach

Groups	VIPeR	PRID	GRID	i-LIDS
None	42.1	27.2	28.6	66.3
1-3	43.9	37.2	29.4	67.1
1-4	48.0	42.3	31.9	69.4
1-5	49.8	47.4	37.1	69.7
1-6	52.7	54.1	39.7	70.5
2-6	48.2	45.1	30.9	67.1
1-7	46.3	42.9	34.8	66.0
1-8	42.3	37.3	29.2	65.1

Table 3: Design-space analysis on where to add IN layers in MobileNetV2 convolution blocks (rank 1 accuracy).

Methods	VIPeR	PRID	GRID	i-LIDS
None	52.7	54.1	39.7	70.5
+ HAFN [47]	47.9	51.2	34.0	64.5
+ IAFN [47]	50.1	53.8	35.6	65.8
+ Ring loss [54]	46.6	49.1	33.8	61.7
+ FN (Ours)	53.9	60.4	41.4	74.8

Table 4: Comparison with other feature norm approaches (cross-domain rank1 accuracy).

yields consistently better results, validating the efficacy of our FN choice.

Compatibility with other backbone networks To demonstrate the generality of our method, we apply it to the same Re-ID problem using another prevalent CNN architecture, ResNet50 [16]. The model is initialized with parameters pre-trained on ImageNet [9]. The initial learning rate is set to 0.05 for classification layer and 0.005 for other base layers, decayed by 0.1 after 40 epochs. The model is trained for 70 epochs in total. Similar to MobileNetV2, the initial convolution block in ResNet50 is referred to as 'Conv1' and the following groups of residual bottlenecks are denoted as 'Conv2_x-Conv5_x' respectively. By default, for experiments on ResNet50, IN is added to the last block of 'Conv2_x' to 'Conv4_x', with the rest of network unchanged. Also, the FN is applied to the penultimate layer after feature encoding.

From Table 5, the following observations can be made: (1) A consistent improvement over vanilla ResNet50 can be achieved with our DualNorm solution, demonstrating its ability to overcome domain-specific biases and enhance the performance of Re-ID DG problems. (2) We evaluate the closely related IBN-a and IBN-b model [34] for our task, with the network architectures in their original paper. It can be seen that they can greatly boost the cross-domain accuracy, but not as significantly as our DualNorm strategy, which considers and alleviates both style and content domain-biases. (3) As for the design-space analysis on the location and number of IN layers, similar to the result on MobileNetV2 in Table 2, we can see that more IN in early layers is beneficial, but IN applied to deep layers deteriorates performance. Note that IN added in '1-3' is exactly the same as ResNet50-IBN-b in [34]. (4) Comparing '1-4' with '2-4', '1-3' with '2-3', we can see that opposite to the results in Table 2, instance normalization on Conv1 in ResNet50 has a detrimental effect. (5) Finally, see that IN and FN together remarkably increase the cross-domain accuracy, confirming their utility in eliminating statistics disparities across domains and their complementary strength. (6) Though ResNet50 [16] gives higher accuracy than MobileNetV2, we still exploit MobileNetV2 as our default backbone network due to its lower number of

Models	VIPeR	PRID	GRID	i-LIDS
ResNet50 [16]	48.5	20.3	29.0	71.3
ResNet50-IBN-a [34]	52.1	39.3	36.2	70.4
ResNet50-IBN-b [34]	54.9	48.5	40.9	68.8
ResNet50+IN (1-2)	51.6	38.2	37.2	76.5
ResNet50+IN (1-3)	54.9	48.5	40.9	68.8
ResNet50+IN (2-3)	55.5	50.2	42.8	70.7
ResNet50+IN (1-4)	55.8	48.8	36.7	73.8
ResNet50+IN (1-5)	24.6	15.5	10.6	32.8
ResNet50+IN (2-4)	56.5	54.9	39.2	74.4
ResNet50+FN	50.9	60.0	38.8	76.7
ResNet50+DualNorm	**59.4**	**69.6**	**43.7**	**78.2**

Table 5: Cross-domain Re-ID performance and design-space analysis on where to add IN layers using ResNet50 backbone (rank 1 accuracy).

Dataset	model	rank-1	rank-5	rank-10	mAP
Market-1501	MobileNetV2	77.2	89.9	93.8	53.9
Market-1501	MobileNetV2+DualNorm	82.6	91.7	95.3	57.2
DukeMTMC-reID	MobileNetV2	65.0	79.8	84.1	44.1
DukeMTMC-reID	MobileNetV2+DualNorm	71.2	82.5	86.3	48.3

Table 6: Within-dataset results on Market-1501 and DukeMTMC-reID dataset.

parameters and flexibility to avoid pre-training on ImageNet [9].

Within-dataset experiments The previous experiments show that our method significantly improves cross-domain generalisation in Re-ID. Finally we evaluate our method in single-dataset re-ID tasks. The results for popular datasets Market-1501 [52] and DukeMTMC-reID [53] are shown in Table 6 with the train/test split and protocol in the original papres. We still train the models from scratch with 300 epochs and learning rate starts from 0.01, decayed by 0.1 after 100 epochs. We can see that our method boosts within-dataset performance as well. The increase in rank 1 is 5.4% and 6.2% on Market-1501 and DukeMTMC-reID respectively, not as significant as cross-dataset experiments since the bias between train/test within one dataset is much less dramatic. Nevertheless, a performance improvement is still observed, indicating that our strategy can handle intra-domain variations as well.

5 Conclusion

We proposed a normalization based domain generalization (DG) baseline for person Re-ID. With a combination of Instance and Feature Normalization, style and content biases between domains are alleviated, promoting generalization and transferability of Deep Re-ID. Experiments demonstrate that our approach surpasses contemporary supervised, unsupervised, and purpose-designed DG methods for multi-source cross-domain Re-ID. We believe this fast and simple but effective method provides a strong baseline that will be invaluable to engineers implementing Re-ID in practice as well as a baseline for future research.

Acknowledgements This work was supported by China Scholarship Council, EPSRC grant EP/R026173/1, National Natural Science Foundation of China (61772067, 61471032, 61472030) and NVIDIA Corporation GPU donation.

References

[1] S. Bai, X. Bai, and Q. Tian. Scalable person re-identification on supervised smoothed manifold. In *CVPR*, 2017.

[2] S. Bak and P. Carr. One-shot metric learning for person re-identification. In *CVPR*, 2017.

[3] Y. Balaji, S. Sankaranarayanan, and R. Chellappa. Metareg: Towards domain generalization using meta-regularization. In *NIPS*, 2018.

[4] S. Bąk, P. Carr, and J.-F. Lalonde. Domain adaptation through synthesis for unsupervised person re-identification. In *ECCV*, 2018.

[5] X. Chang, Y. Yang, T. Xiang, and T. Hospedales. Disjoint label space transfer learning with common factorised space. In *AAAI*, 2019.

[6] W. Chen, X. Chen, J. Zhang, and K. Huang. A multi-task deep network for person re-identification. In *AAAI*, 2017.

[7] D. Cheng, Y. Gong, S. Zhou, J. Wang, and N. Zheng. Person re-identification by multi-channel parts-based cnn with improved triplet loss function. In *CVPR*, 2016.

[8] H. Daumé. Frustratingly easy domain adaptation. In *ACL*, 2007.

[9] J. Deng, W. Dong, R. Socher, L. Li, K. Li, and F. Li. Imagenet: A large-scale hierarchical image database. In *CVPR*, 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 reidentification. In *CVPR*, 2018.

[11] Y. Deng, P. Luo, C. C. Loy, and X. Tang. Pedestrian attribute recognition at far distance. In *ACMMM*, 2014.

[12] M. Farenzena, L. Bazzani, A. Perina, V. Murino, and M. Cristani. Person re-identification by symmetry-driven accumulation of local features. In *CVPR*, 2010.

[13] M. Ghifary, W. B. Kleijn, M. Zhang, and D. Balduzzi. Domain generalization for object recognition with multi-task autoencoders. In *ICCV*, 2015.

[14] D. Gray and H. Tao. Viewpoint invariant pedestrian recognition with an ensemble of localized features. In *ECCV*, 2008.

[15] A. Gretton, K. Borgwardt, M. Rasch, B. Schölkopf, and A. J. Smola. A kernel method for the two-sample-problem. In *NIPS*, 2007.

[16] K. He, X. Zhang, S. Ren, and S. Jian. Deep residual learning for image recognition. In *CVPR*, 2016.

[17] M. Hirzer, C. Beleznai, P. M. Roth, and H. Bischof. Person re-identification by descriptive and discriminative classification. In *Scandinavian Conference on Image Analysis*, 2011.

[18] H. Huang, W. Yang, X. Chen, X. Zhao, K. Huang, J. Lin, G. Huang, and D. Du. Eanet: Enhancing alignment for cross-domain person re-identification. *arXiv preprint arXiv:1812.11369*, 2018.

[19] S. Ioffe and C. Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *ICML*, 2015.

[20] A. Jabri, A. Joulin, and L. Van der Maaten. Revisiting visual question answering baselines. In *ECCV*, 2016.

[21] A. Khosla, T. Zhou, T. Malisiewicz, A. Efros, and A. Torralba. Undoing the damage of dataset bias. In *ECCV*, 2012.

[22] D. Li, Y. Yang, Y. Song, and T. Hospedales. Learning to generalize: Meta-learning for domain generalization. In *AAAI*, 2018.

[23] K. Li, Z. Ding, S. Li, and Y. Fu. Discriminative semi-coupled projective dictionary learning for low-resolution person re-identification. In *AAAI*, 2018.

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

[25] W. Li, R. Zhao, T. Xiao, and X. Wang. Deepreid: Deep filter pairing neural network for person re-identification. In *CVPR*, 2014.

[26] W. Li, X. Zhu, and S. Gong. Person re-identification by deep joint learning of multi-loss classification. In *IJCAI*, 2017.

[27] Y. Li, Y. Yang, W. Zhou, and T. Hospedales. Feature-critic networks for heterogeneous domain generalization. In *ICML*, 2019.

[28] S. Lin, H. Li, C.-T. Li, and A. C. Kot. Multi-task mid-level feature alignment network for unsupervised cross-dataset person re-identification. In *BMVC*, 2018.

[29] C. C. Loy, T. Xiang, and S. Gong. Multi-camera activity correlation analysis. In *CVPR*, 2009.

[30] H. Luo, Y. Gu, X. Liao, S. Lai, and W. Jiang. Bag of tricks and a strong baseline for deep person re-identification. In *CVPR Workshops*, 2019.

[31] T. Matsukawa, T. Okabe, E. Suzuki, and Y. Sato. Hierarchical gaussian descriptor for person re-identification. In *CVPR*, 2016.

[32] H. Nam and H.-E. Kim. Batch-instance normalization for adaptively style-invariant neural networks. In *NIPS*, 2018.

[33] S. Paisitkriangkrai, C. Shen, and A. Van Den Hengel. Learning to rank in person re-identification with metric ensembles. In *CVPR*, 2015.

[34] X. Pan, P. Luo, J. Shi, and X. Tang. Two at once: Enhancing learning and generalization capacities via ibn-net. In *ECCV*, 2018.

[35] A. Paszke, S. Gross, S. Chintala, G. Chanan, E. Yang, Z. DeVito, Z. Lin, A. Desmaison, L. Antiga, and A. Lerer. Automatic differentiation in pytorch. In *NIPS Workshops*, 2017.

[36] P. Peng, T. Xiang, Y. Wang, M. Pontil, S. Gong, T. Huang, and Y. Tian. Unsupervised cross-dataset transfer learning for person re-identification. In *CVPR*, 2016.

[37] B. Romera-Paredes and P. H. S. Torr. An embarrassingly simple approach to zero-shot learning. In *ICML*, 2015.

[38] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen. Mobilenetv2: Inverted residuals and linear bottlenecks. In *CVPR*, 2018.

[39] S. Santurkar, D. Tsipras, A. Ilyas, and A. Madry. How does batch normalization help optimization? In *NIPS*, 2018.

[40] J. Song, Y. Yang, Y. Song, T. Xiang, and T. Hospedales. Generalizable person re-identification by domain-invariant mapping network. In *CVPR*, 2019.

[41] C. Su, S. Zhang, J. Xing, W. Gao, and Q. Tian. Deep attributes driven multi-camera person re-identification. In *ECCV*, 2016.

[42] X. Tong, L. Shuang, B. Wang, L. Liang, and X. Wang. End-to-end deep learning for person search. 2016.

[43] E. Tzeng, J. Hoffman, K. Saenko, and T. Darrell. Adversarial discriminative domain adaptation. In *CVPR*, 2017.

[44] D. Ulyanov, A. Vedaldi, and V. Lempitsky. Instance normalization: The missing ingredient for fast stylization. *arXiv preprint arXiv:1607.08022*, 2016.

[45] J. Wang, X. Zhu, S. Gong, and W. Li. Transferable joint attribute-identity deep learning for unsupervised person re-identification. In *CVPR*, 2018.

[46] T. Xiao, H. Li, W. Ouyang, and X. Wang. Learning deep feature representations with domain guided dropout for person re-identification. In *CVPR*, 2016.

[47] R. Xu, G. Li, J. Yang, and L. Lin. Unsupervised domain adaptation: An adaptive feature norm approach. *arXiv preprint arXiv:1811.07456*, 2018.

[48] H. Yan, Y. Ding, P. Li, Q. Wang, Y. Xu, and W. Zuo. Mind the class weight bias: Weighted maximum mean discrepancy for unsupervised domain adaptation. In *CVPR*, 2017.

[49] H. Yu, A. Wu, and W. Zheng. Cross-view asymmetric metric learning for unsupervised person re-identification. In *ICCV*, 2017.

[50] L. Zhang, T. Xiang, and S. Gong. Learning a discriminative null space for person re-identification. In *CVPR*, 2016.

[51] H. Zhao, M. Tian, S. Sun, J. Shao, J. Yan, S. Yi, X. Wang, and X. Tang. Spindle net: Person re-identification with human body region guided feature decomposition and fusion. In *CVPR*, 2017.

[52] L. Zheng, L. Shen, L. Tian, S. Wang, J. Wang, and Q. Tian. Scalable person re-identification: A benchmark. In *ICCV*, 2015.

[53] W. Zheng, S. Gong, and T. Xiang. Associating groups of people. *Active Range Imaging Dataset for Indoor Surveillance*, 2009.

[54] Y. Zheng, D. K. Pal, and M. Savvides. Ring loss: Convex feature normalization for face recognition. In *CVPR*, 2018.

[55] Z. Zheng, L. Zheng, and Y. Yang. Unlabeled samples generated by gan improve the person re-identification baseline in vitro. In *ICCV*, 2017.

[56] Z. Zhong, L. Zheng, Z. Luo, S. Li, and Y. Yang. Invariance matters: Exemplar memory for domain adaptive person re-identification. In *CVPR*, 2019.