

Omni-Scale Feature Learning for Person Re-Identification

Kaiyang Zhou¹ Yongxin Yang¹ Andrea Cavallaro² Tao Xiang^{1,3}

¹University of Surrey ²Queen Mary University of London

³Samsung AI Center, Cambridge

{k.zhou, yongxin.yang, t.xiang}@surrey.ac.uk a.cavallaro@qmul.ac.uk

Abstract

As an instance-level recognition problem, person re-identification (re-ID) relies on discriminative features, which not only capture different spatial scales but also encapsulate an arbitrary combination of multiple scales. We call features of both homogeneous and heterogeneous scales *omni-scale features*. In this paper, a novel deep re-ID CNN is designed, termed *omni-scale network (OSNet)*, for *omni-scale feature learning*. This is achieved by designing a residual block composed of multiple convolutional streams, each detecting features at a certain scale. Importantly, a novel unified aggregation gate is introduced to dynamically fuse multi-scale features with input-dependent channel-wise weights. To efficiently learn spatial-channel correlations and avoid overfitting, the building block uses pointwise and depthwise convolutions. By stacking such block layer-by-layer, *our OSNet is extremely lightweight and can be trained from scratch on existing re-ID benchmarks*. Despite its small model size, OSNet achieves state-of-the-art performance on six person re-ID datasets, outperforming most large-sized models, often by a clear margin. Code and models are available at: <https://github.com/KaiyangZhou/deep-person-reid>.

1. Introduction

Person re-identification (re-ID), a fundamental task in distributed multi-camera surveillance, *aims to match people appearing in different non-overlapping camera views*. As an instance-level recognition problem, person re-ID faces two major challenges as illustrated in Fig. 1. First, the *intra-class (instance/identity) variations are typically big due to the changes of camera viewing conditions*. For instance, both people in Figs. 1(a) and (b) carry a backpack; the view change across cameras (frontal to back) brings large appearance changes in the backpack area, making matching the same person difficult. Second, there are also small *inter-class variations* – people in public space often wear similar



Figure 1: Person re-ID is a hard problem, as exemplified by the four triplets of images above. Each sub-figure shows, from left to right, the query image, a true match and an impostor/false match.

clothes; from a distance as typically in surveillance videos, they can look incredibly similar (see the impostors for all four people in Fig. 1).

To overcome these two challenges, *key to re-ID is to learn discriminative features*. We argue that such features need to be of *omni-scale*, defined as the combination of variable homogeneous scales and heterogeneous scales, each of which is composed of a mixture of multiple scales. The need for omni-scale features is evident from Fig. 1. To match people and distinguish them from impostors, features corresponding small local regions (e.g. shoes, glasses) and global whole body regions are equally important. For example, given the query image in Fig. 1(a) (left), looking at the global-scale features (e.g. young man, a white T-shirt + grey shorts combo) would narrow down the search to the true match (middle) and an impostor (right). Now the local-scale features come into play. The shoe region gives away the fact that the person on the right is an impostor (trainers vs. sandals). However, for more challenging cases, even features of variable homogeneous scales would not be enough and more complicated and richer features that span multiple scales are required. For instance, to eliminate the impostor in Fig. 1(d) (right), one needs features that repre-

sent a white T-shirt with a specific logo in the front. Note that the logo is not distinctive on its own – without the white T-shirt as context, it can be confused with many other patterns. Similarly, the white T-shirt is likely everywhere in summer (e.g. Fig. 1(a)). It is however the unique combination, captured by heterogeneous-scale features spanning both small (logo size) and medium (upper body size) scales, that makes the features most effective.

Nevertheless, none of the existing re-ID models addresses omni-scale feature learning. In recent years, deep convolutional neural networks (CNNs) have been widely used in person re-ID to learn discriminative features [2, 29, 33, 47, 53, 69, 71, 84]. However, most of the CNNs adopted, such as ResNet [14], were originally designed for object category-level recognition tasks that are fundamentally different from the instance-level recognition task in re-ID. For the latter, omni-scale features are more important, as explained earlier. A few attempts at learning multi-scale features also exist [38, 2]. Yet, none has the ability to learn features of both homogeneous and heterogeneous scales.

In this paper, we present *OSNet*, a novel CNN architecture designed for learning omni-scale feature representations. The underpinning building block consists of multiple convolutional streams with different receptive field sizes¹ (see Fig. 2). The feature scale that each stream focuses on is determined by *exponent*, a new dimension factor that is linearly increased across streams to ensure that various scales are captured in each block. Critically, the resulting multi-scale feature maps are dynamically fused by channel-wise weights that are generated by a unified aggregation gate (AG). The AG is a mini-network sharing parameters across all streams with a number of desirable properties for effective model training. With the trainable AG, the generated channel-wise weights become input-dependent, hence the dynamic scale fusion. This novel AG design allows the network to learn omni-scale feature representations: depending on the specific input image, the gate could focus on a single scale by assigning a dominant weight to a particular stream or scale; alternatively, it can pick and mix and thus produce heterogeneous scales.

Apart from omni-scale feature learning, another key design principle adopted in OSNet is to construct a *lightweight network*. This brings a couple of benefits: (1) re-ID datasets are often of moderate size due to the difficulties in collecting across-camera matched person images. A lightweight network with a small number of parameters is thus less prone to overfitting. (2) In a large-scale surveillance application (e.g. city-wide surveillance using thousands of cameras), the most practical way for re-ID is to perform feature extraction at the camera end. Instead of sending the raw videos to a central server, only the extracted features need to be sent. For on-device processing, small re-ID networks are clearly

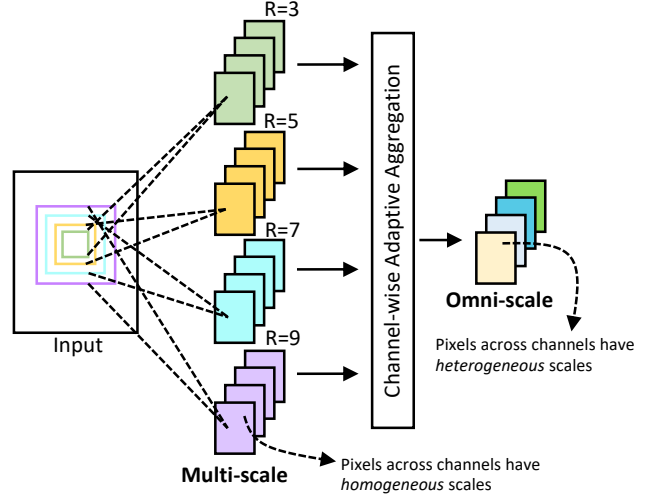


Figure 2: A schematic of the proposed building block for OSNet. R: Receptive field size.

preferred. To this end, in our building block, we factorise standard convolutions with pointwise and depthwise convolutions [18, 43]. The **contributions** of this work are thus both *the concept of omni-scale feature learning* and *an effective and efficient implementation of it in OSNet*. The end result is a lightweight re-ID model that is more than one order of magnitude smaller than the popular ResNet50-based models, but performs better: OSNet achieves state-of-the-art performance on six person re-ID datasets, beating much larger networks, often by a clear margin. We also demonstrate the effectiveness of OSNet on object category recognition tasks, namely CIFAR [23] and ImageNet [7], and a multi-label person attribute recognition task. The results suggest that omni-scale feature learning is useful beyond instance recognition and can be considered for a broad range of visual recognition tasks. Code and pre-trained models are available in Torchreid [91]².

2. Related Work

Deep re-ID architectures. Most existing deep re-ID CNNs [27, 1, 59, 46, 13, 50, 63] borrow architectures designed for generic object categorisation problems, such as ImageNet 1K object classification. Recently, some architectural modifications are introduced to reflect the fact that images in re-ID datasets contain instances of only one object category (i.e., person) that mostly stand upright. To exploit the upright body pose, [53, 77, 10, 61] add auxiliary supervision signals to features pooled horizontally from the last convolutional feature maps. [47, 48, 29] devise attention mechanisms to focus feature learning on the foreground person regions. In [81, 49, 69, 52, 57, 80], body part-specific CNNs are learned by means of off-the-shelf pose detectors.

¹We use scale and receptive field interchangeably.

²<https://github.com/KaiyangZhou/deep-person-reid>

In [28, 25, 82], CNNs are branched to learn representations of global and local image regions. In [73, 2, 33, 64], multi-level features extracted at different layers are combined. However, *none* of these re-ID networks learns multi-scale features explicitly at each layer of the networks as in our OSNet – they typically rely on an external pose model and/or hand-pick specific layers for multi-scale learning. Moreover, heterogeneous-scale features computed from a mixture of different scales are not considered.

Multi-scale feature learning. As far as we know, the concept of omni-scale deep feature learning has never been introduced before. Nonetheless, the importance of multi-scale feature learning has been recognised recently and the multi-stream building block design has also been adopted. Compared to a number of re-ID networks with multi-stream building blocks [2, 38], OSNet is significantly different. Specifically the layer design in [2] is based on ResNeXt [68], where each stream learns features at the same scale, while our streams in each block have different scales. Different to [2], the network in [38] is built on Inception [54, 55], where multiple streams were originally designed for low computational cost with handcrafted mixture of convolution and pooling layers. In contrast, our building block uses a scale-controlling factor to diversify the spatial scales to be captured. Moreover, [38] fuses multi-stream features with learnable but fixed-once-learned streamwise weights only at the final block. Whereas we fuse multi-scale features within each building block using dynamic (input-dependent) channel-wise weights to learn combinations of multi-scale patterns. Therefore, only our OSNet is capable of learning omni-scale features with each feature channel potentially capturing discriminative features of either a single scale or a weighted mixture of multiple scales. Our experiments (see Sec. 4.1) show that OSNet significantly outperforms the models in [2, 38].

Lightweight network designs. With embedded AI becoming topical, lightweight CNN design has attracted increasing attention. SqueezeNet [22] compresses feature dimensions using 1×1 convolutions. IGCNet [76], ResNeXt [68] and CondenseNet [20] leverage group convolutions. Xception [5] and MobileNet series [18, 43] are based on depthwise separable convolutions. Dense 1×1 convolutions are grouped with channel shuffling in ShuffleNet [78]. In terms of lightweight design, our OSNet is similar to MobileNet by employing factorised convolutions, with some modifications that empirically work better for omni-scale feature learning.

3. Omni-Scale Feature Learning

In this section, we present OSNet, which specialises in learning omni-scale feature representations for the person re-ID task. We start with the factorised convolutional layer and then introduce the omni-scale residual block and the

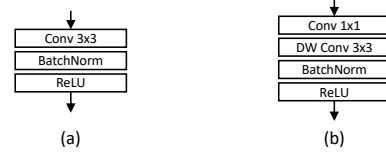


Figure 3: (a) Standard 3×3 convolution. (b) *Lite* 3×3 convolution. DW: Depth-Wise.

unified aggregation gate.

3.1. Depthwise Separable Convolutions

To reduce the number of parameters, we adopt the depthwise separable convolutions [18, 5]. The basic idea is to divide a convolution layer $\text{ReLU}(w * x)$ with kernel $w \in \mathbb{R}^{k \times k \times c \times c'}$ into two separate layers $\text{ReLU}((v \circ u) * x)$ with depthwise kernel $u \in \mathbb{R}^{k \times k \times 1 \times c'}$ and pointwise kernel $v \in \mathbb{R}^{1 \times 1 \times c \times c'}$, where $*$ denotes convolution, k the kernel size, c the input channel width and c' the output channel width. Given an input tensor $x \in \mathbb{R}^{h \times w \times c}$ of height h and width w , the computational cost is reduced from $h \cdot w \cdot k^2 \cdot c \cdot c'$ to $h \cdot w \cdot (k^2 + c) \cdot c'$, and the number of parameters from $k^2 \cdot c \cdot c'$ to $(k^2 + c) \cdot c'$. In our implementation, we use $\text{ReLU}((u \circ v) * x)$ (pointwise \rightarrow depthwise instead of depthwise \rightarrow pointwise), which turns out to be more effective for omni-scale feature learning³. We call such layer *Lite* 3×3 hereafter. The implementation is shown in Fig. 3.

3.2. Omni-Scale Residual Block

The building block in our architecture is the residual bottleneck [14], equipped with the *Lite* 3×3 layer (see Fig. 4(a)). Given an input x , this bottleneck aims to learn a residual \tilde{x} with a mapping function F , i.e.

$$y = x + \tilde{x}, \quad \text{s.t.} \quad \tilde{x} = F(x), \quad (1)$$

where F represents a *Lite* 3×3 layer that learns single-scale features (scale = 3). Note that here the 1×1 layers are ignored in notation as they are used to manipulate feature dimension and do not contribute to the aggregation of spatial information [14, 68].

Multi-scale feature learning. To achieve multi-scale feature learning, we extend the residual function F by introducing a new dimension, *exponent* t , which represents the scale of the feature. For F^t , with $t > 1$, we stack t *Lite* 3×3 layers, and this results in a receptive field of size $(2t + 1) \times (2t + 1)$. Then, the residual to be learned, \tilde{x} , is the sum of incremental scales of representations up to T :

$$\tilde{x} = \sum_{t=1}^T F^t(x), \quad \text{s.t.} \quad T \geq 1. \quad (2)$$

³The subtle difference between the two orders is when the channel width is increased: pointwise \rightarrow depthwise increases the channel width before spatial aggregation.

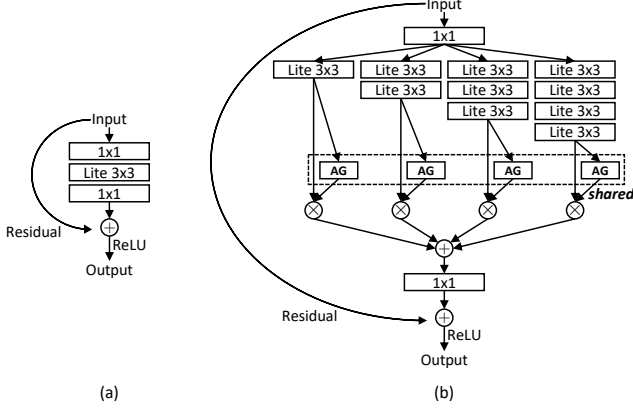


Figure 4: (a) Baseline bottleneck. (b) Proposed bottleneck. AG: Aggregation Gate. The first/last 1×1 layers are used to reduce/restore feature dimension.

When $T = 1$, Eq. 2 reduces to Eq. 1 (see Fig. 4(a)). In this paper, our bottleneck is set with $T = 4$ (i.e. the largest receptive field is 9×9) as shown in Fig. 4(b). The shortcut connection allows features at smaller scales learned in the current layer to be preserved effectively in the next layers, thus enabling the final features to capture a whole range of spatial scales.

Unified aggregation gate. So far, each stream can give us features of a specific scale, i.e., they are scale homogeneous. To learn omni-scale features, we propose to combine the outputs of different streams in a dynamic way, i.e., different weights are assigned to different scales according to the input image, rather than being fixed after training. More specifically, the dynamic scale-fusion is achieved by a novel aggregation gate (AG), which is a *learnable neural network*.

Let \mathbf{x}^t denote $F^t(\mathbf{x})$, the omni-scale residual $\tilde{\mathbf{x}}$ is obtained by

$$\tilde{\mathbf{x}} = \sum_{t=1}^T G(\mathbf{x}^t) \odot \mathbf{x}^t, \quad \text{s.t.} \quad \mathbf{x}^t \triangleq F^t(\mathbf{x}), \quad (3)$$

where $G(\mathbf{x}^t)$ is a vector with length spanning the entire channel dimension of \mathbf{x}^t and \odot denotes the Hadamard product. G is implemented as a mini-network composed of a non-parametric global average pooling layer [30] and a multi-layer perceptron (MLP) with one ReLU-activated hidden layer, followed by the sigmoid activation. To reduce parameter overhead, we follow [67, 19] to reduce the hidden dimension of the MLP with a reduction ratio, which is set to 16.

It is worth pointing out that, in contrast to using a single scalar-output function that provides a coarse scale-fusion, we choose to use channel-wise weights, i.e., the output of the AG network $G(\mathbf{x}^t)$ is a vector rather a scalar for the t -th stream. This design results in a more fine-grained fusion that tunes each feature channel. In addition, the weights are

dynamically computed by being conditioned on the input data. This is crucial for re-ID as the test images contain people of different identities from those in training; thus an adaptive/input-dependent feature-scale fusion strategy is more desirable.

Note that in our architecture, the AG is *shared* for all feature streams in the same omni-scale residual block (dashed box in Fig. 4(b)). This is similar in spirit to the convolution filter parameter sharing in CNNs, resulting in a number of advantages. First, the number of parameters is independent of T (number of streams), thus the model becomes more scalable. Second, unifying AG (sharing the same AG module across streams) has a nice property while performing backpropagation. Concretely, suppose the network is supervised by a loss function \mathcal{L} which is differentiable and the gradient $\frac{\partial \mathcal{L}}{\partial \tilde{\mathbf{x}}}$ can be computed; the gradient w.r.t G , based on Eq. 3, is

$$\frac{\partial \mathcal{L}}{\partial G} = \frac{\partial \mathcal{L}}{\partial \tilde{\mathbf{x}}} \frac{\partial \tilde{\mathbf{x}}}{\partial G} = \frac{\partial \mathcal{L}}{\partial \tilde{\mathbf{x}}} \left(\sum_{t=1}^T \mathbf{x}^t \right). \quad (4)$$

The second term in Eq. 4 indicates that the supervision signals from all streams are gathered together to guide the learning of G . This desirable property disappears when each stream has its own gate.

3.3. Network Architecture

OSNet is constructed by simply stacking the proposed lightweight bottleneck layer-by-layer without any effort to customise the blocks at different depths (stages) of the network. The detailed network architecture is shown in Table 1. For comparison, the same network architecture with standard convolutions has 6.9 million parameters and 3,384.9 million mult-add operations, which are $3\times$ larger than our OSNet with the Lite 3×3 convolution layer design. The standard OSNet in Table 1 can be easily scaled up or down in practice, to balance model size, computational cost and performance. To this end, we use a width multiplier⁴ and an image resolution multiplier, following [18, 43, 78].

Relation to prior architectures. In terms of multi-stream design, OSNet is related to Inception [54] and ResNeXt [68], but has crucial differences in several aspects. First, the multi-stream design in OSNet strictly follows the scale-incremental principle dictated by the exponent (Eq. 2). Specifically, different streams have different receptive fields but are built with the same Lite 3×3 layers (Fig. 4(b)). Such a design is more effective at capturing a wide range of scales. In contrast, Inception was originally designed to have low computational costs by sharing computations with multiple streams. Therefore its structure, which includes mixed operations of convolution and pooling, was handcrafted. ResNeXt has multi-

⁴Width multiplier with magnitude smaller than 1 works on all layers in OSNet except the last FC layer whose feature dimension is fixed to 512.

stage	output	OSNet
conv1	128×64, 64	7×7 conv, stride 2
	64×32, 64	3×3 max pool, stride 2
conv2	64×32, 256	bottleneck × 2
transition	64×32, 256	1×1 conv
	32×16, 256	2×2 average pool, stride 2
conv3	32×16, 384	bottleneck × 2
transition	32×16, 384	1×1 conv
	16×8, 384	2×2 average pool, stride 2
conv4	16×8, 512	bottleneck × 2
conv5	16×8, 512	1×1 conv
gap	1×1, 512	global average pool
fc	1×1, 512	fc
# params		2.2M
Mult-Adds		978.9M

Table 1: Architecture of OSNet with input image size 256×128 .

ple equal-scale streams thus learning representations at the same scale. Second, Inception/ResNeXt aggregates features by concatenation/addition while OSNet uses a unified AG (Eq. 3), which facilitates the learning of combinations of multi-scale features. Critically, it means that the fusion is dynamic and adaptive to each individual input image. Therefore, OSNet’s architecture is fundamentally different from that of Inception/ResNeXt in nature. Third, OSNet uses factorised convolutions and thus the building block and subsequently the whole network is lightweight. Compared with SENet [19], OSNet is conceptually different. Concretely, SENet aims to re-calibrate the feature channels by re-scaling the activation values for a single stream, whereas OSNet is designed to selectively fuse multiple feature streams of different receptive field sizes in order to learn omni-scale features (see Fig. 2).

4. Experiments

4.1. Evaluation on Person Re-Identification

Datasets and settings. We conduct experiments on six widely used person re-ID datasets: Market1501 [83], CUHK03 [27], DukeMTMC-reID (Duke) [42, 85], MSMT17 [65], VIPeR [12] and GRID [35]. Detailed dataset statistics are provided in Table 2. The first four are considered as ‘big’ datasets even though their sizes (around 30K training images for the largest MSMT17) are fairly moderate; while VIPeR and GRID are generally too small to train without using those big datasets for pre-training. For CUHK03, we use the 767/700 split [86] with the detected images. For VIPeR and GRID, we first train a single OSNet from scratch using training images from Market1501, CUHK03, Duke and MSMT17 (Mix4), and then perform fine-tuning. Following [28], the results on VIPeR and GRID are averaged over 10 random splits. Such a fine-tuning strat-

Dataset	# IDs (T-Q-G)	# images (T-Q-G)
Market1501	751-750-751	12936-3368-15913
CUHK03	767-700-700	7365-1400-5332
Duke	702-702-1110	16522-2228-17661
MSMT17	1041-3060-3060	30248-11659-82161
VIPeR	316-316-316	632-632-632
GRID	125-125-900	250-125-900

Table 2: Dataset statistics. T: Train. Q: Query. G: Gallery.

egy has been commonly adopted by other deep learning approaches [33, 66, 81, 28, 82]. Cumulative matching characteristics (CMC) Rank-1 accuracy and mAP are used as evaluation metrics.

Implementation details. A classification layer (linear FC + softmax) is mounted on the top of OSNet. Training follows the standard classification paradigm where each person identity is regarded as a unique class. Similar to [29, 2], cross entropy loss with label smoothing [55] is used for supervision. For fair comparison against existing models, we implement two versions of OSNet. One is trained from scratch and the other is fine-tuned from ImageNet pre-trained weights. Person matching is based on the ℓ_2 distance of 512-D feature vectors extracted from the last FC layer (see Table 1). Batch size and weight decay are set to 64 and $5e-4$ respectively. For training from scratch, SGD is used to train the network for 350 epochs. The learning rate starts from 0.065 and is decayed by 0.1 at 150, 225 and 300 epochs. Data augmentation includes random flip, random crop and random patch⁵. For fine-tuning, we train the network with AMSGrad [41] and initial learning rate of 0.0015 for 150 epochs. The learning rate is decayed by 0.1 every 60 epochs. During the first 10 epochs, the ImageNet pre-trained base network is frozen and only the randomly initialised classifier is open for training. Images are resized to 256×128 . Data augmentation includes random flip and random erasing [87]. The code is based on Torchreid [91].

Results on big re-ID datasets. From Table 3, we have the following observations. (1) OSNet achieves state-of-the-art performance on all datasets, outperforming most published methods by a clear margin. It is evident from Table 3 that the performance on re-ID benchmarks, especially Market1501 and Duke, has been saturated lately. Therefore, the improvements obtained by OSNet are significant. Crucially, the improvements are achieved with *much smaller model size* – most existing state-of-the-art re-ID models employ a ResNet50 backbone, which has more than 24 million parameters (considering their extra customised modules), while our OSNet has only 2.2 million parameters. This verifies the effectiveness of omni-scale feature learning for re-ID achieved by an extremely com-

⁵RandomPatch works by (1) constructing a patch pool that stores randomly extracted image patches and (2) pasting a random patch selected from the patch pool onto an input image at random position.

Method	Publication	Backbone	Market1501		CUHK03		Duke		MSMT17	
			R1	mAP	R1	mAP	R1	mAP	R1	mAP
ShuffleNet ^{†‡} [78]	CVPR'18	ShuffleNet	84.8	65.0	38.4	37.2	71.6	49.9	41.5	19.9
MobileNetV2 ^{†‡} [43]	CVPR'18	MobileNetV2	87.0	69.5	46.5	46.0	75.2	55.8	50.9	27.0
BraidNet [†] [63]	CVPR'18	BraidNet	83.7	69.5	-	-	76.4	59.5	-	-
HAN [†] [29]	CVPR'18	Inception	91.2	75.7	41.7	38.6	80.5	63.8	-	-
OSNet [†] (ours)	ICCV'19	OSNet	93.6	81.0	57.1	54.2	84.7	68.6	71.0	43.3
DaRe [64]	CVPR'18	DenseNet	89.0	76.0	63.3	59.0	80.2	64.5	-	-
PNGAN [39]	ECCV'18	ResNet	89.4	72.6	-	-	73.6	53.2	-	-
KPM [46]	CVPR'18	ResNet	90.1	75.3	-	-	80.3	63.2	-	-
MLFN [2]	CVPR'18	ResNeXt	90.0	74.3	52.8	47.8	81.0	62.8	-	-
FDGAN [11]	NeurIPS'18	ResNet	90.5	77.7	-	-	80.0	64.5	-	-
DuATM [47]	CVPR'18	DenseNet	91.4	76.6	-	-	81.8	64.6	-	-
Bilinear [52]	ECCV'18	Inception	91.7	79.6	-	-	84.4	69.3	-	-
G2G [44]	CVPR'18	ResNet	92.7	82.5	-	-	80.7	66.4	-	-
DeepCRF [3]	CVPR'18	ResNet	93.5	81.6	-	-	84.9	69.5	-	-
PCB [53]	ECCV'18	ResNet	93.8	81.6	63.7	57.5	83.3	69.2	68.2	40.4
SGGNN [45]	ECCV'18	ResNet	92.3	82.8	-	-	81.1	68.2	-	-
Mancs [60]	ECCV'18	ResNet	93.1	82.3	65.5	60.5	84.9	71.8	-	-
AANet [56]	CVPR'19	ResNet	93.9	83.4	-	-	87.7	74.3	-	-
CAMA [71]	CVPR'19	ResNet	94.7	84.5	66.6	64.2	85.8	72.9	-	-
IANet [17]	CVPR'19	ResNet	94.4	83.1	-	-	87.1	73.4	75.5	46.8
DGNet [84]	CVPR'19	ResNet	94.8	86.0	-	-	86.6	74.8	77.2	52.3
OSNet (ours)	ICCV'19	OSNet	94.8	84.9	72.3	67.8	88.6	73.5	78.7	52.9

Table 3: Results (%) on big re-ID datasets. It is clear that OSNet achieves state-of-the-art performance on all datasets, surpassing most published methods by a clear margin. It is noteworthy that *OSNet has only 2.2 million parameters*, which are far less than the current best-performing ResNet-based methods. -: not available. †: model trained from scratch. ‡: reproduced by us. (Best and second best results in red and blue respectively)

pact network. As OSNet is orthogonal to some methods, such as the image generation based DGNet [84], they can be potentially combined to further boost the re-ID performance. (2) OSNet yields strong performance with or without ImageNet pre-training. Among the very few existing lightweight re-ID models that can be trained from scratch (HAN and BraidNet), OSNet exhibits huge advantages. At R1, OSNet beats HAN/BraidNet by 2.4%/9.9% on Market1501 and 4.2%/8.3% on Duke. The margins at mAP are even larger. In addition, general-purpose lightweight CNNs are also compared without ImageNet pre-training. Table 3 shows that OSNet surpasses the popular MobileNetV2 and ShuffleNet by large margins on all datasets. Note that all three networks have similar model sizes. These results thus demonstrate the versatility of our OSNet: It enables effective feature tuning from generic object categorisation tasks and offers robustness against model overfitting when trained from scratch on datasets of moderate sizes. (3) Compared with re-ID models that deploy a multi-scale/multi-stream architecture, namely those with a Inception or ResNeXt backbone [29, 49, 4, 66, 2, 47], OSNet is clearly superior. As analysed in Sec. 3, this is attributed to the unique ability of OSNet to learn heterogeneous-scale features by combining multiple homogeneous-scale features with the dynamic AG.

Results on small re-ID datasets. VIPeR and GRID are very challenging datasets for deep re-ID approaches because they have only hundreds of training images - training on the large re-ID datasets and fine-tuning on them is thus necessary. Table 4 compares OSNet with six state-of-the-art deep re-ID methods. On VIPeR, it can be observed that OSNet outperforms the alternatives by a significant margin – more than 11.4% at R1. GRID is much more challenging than VIPeR because it has only 125 training identities (250 images) and extra distractors. Further, it was captured by real (operational) analogue CCTV cameras installed in busy public spaces. JLML [28] is currently the best published method on GRID. It is noted that OSNet is marginally better than JLML on GRID. Overall, the strong performance of OSNet on these two small datasets is indicative of its practical usefulness in real-world applications where collecting large-scale training data is unscalable.

Ablation experiments. Table 5 evaluates our architectural design choices where our primary model is model 1. T is the stream cardinality in Eq. 2. (1) vs. *standard convolutions*: Factorising convolutions reduces the R1 marginally by 0.4% (model 2 vs. 1). This means our architecture design maintains the representational power even though the model size is reduced by more than $3\times$. (2) vs. *ResNeXt-like design*: OSNet is transformed into a ResNeXt-like archi-

Method	Backbone	VIPeR	GRID
MuDeep [38]	Inception	43.0	-
DeepAlign [82]	Inception	48.7	-
JLML [28]	ResNet	50.2	37.5
Spindle [81]	Inception	53.8	-
GLAD [66]	Inception	54.8	-
HydraPlus-Net [33]	Inception	56.6	-
OSNet (ours)	OSNet	68.0	38.2

Table 4: Comparison with deep learning approaches on VIPeR and GRID. Only Rank-1 accuracy (%) is reported. -: not available.

Model	Architecture	Market1501	
		R1	mAP
1	$T = 4$ + unified AG (primary model)	93.6	81.0
2	$T = 4$ w/ full conv + unified AG	94.0	82.7
3	$T = 4$ (same depth) + unified AG	91.7	77.9
4	$T = 4$ + concatenation	91.4	77.4
5	$T = 4$ + addition	92.0	78.2
6	$T = 4$ + separate AGs	92.9	80.2
7	$T = 4$ + unified AG (stream-wise)	92.6	80.0
8	$T = 4$ + learned-and-fixed gates	91.6	77.5
9	$T = 1$	86.5	67.7
10	$T = 2$ + unified AG	91.7	77.0
11	$T = 3$ + unified AG	92.8	79.9

Table 5: Ablation study on architectural design choices.

ture by making all streams homogeneous in depth while preserving the unified AG, which refers to model 3. We observe that this variant is clearly outperformed by the primary model, with 1.9%/3.1% difference in R1/mAP. This further validates the necessity of our omni-scale design. (3) *Multi-scale fusion strategy*: To justify our design of the unified AG, we conduct experiments by changing the way how features of different scales are aggregated. The baselines are concatenation (model 4) and addition (model 5). The primary model is better than the two baselines by more than 1.6%/2.8% at R1/mAP. Nevertheless, models 4 and 5 are still much better than the single-scale architecture (model 9). (4) *Unified AG vs. separate AGs*: When separate AGs are learned for each feature stream, the model size is increased and the nice property in gradient computation (Eq. 4) is lost. Empirically, unifying AG improves by 0.7%/0.8% at R1/mAP (model 1 vs. 6), despite having less parameters. (5) *Channel-wise gates vs. stream-wise gates*: By turning the channel-wise gates into stream-wise gates (model 7), both the R1 and the mAP decline by 1%. As feature channels encapsulate sophisticated correlations and can represent numerous visual concepts [9], it is advantageous to use channel-specific weights. (6) *Dynamic gates vs. static gates*: In model 8, feature streams are fused by static (learned-and-then-fixed) channel-wise gates to mimic the design in [38]. As a result, the R1/mAP drops off by 2.0%/3.5% compared with that of dynamic gates (primary

β	# params	γ	Mult-Adds	Market1501	
				R1	mAP
1.0	2.2M	1.0	978.9M	94.8	84.9
0.75	1.3M	1.0	571.8M	94.5	84.1
0.5	0.6M	1.0	272.9M	93.4	82.6
0.25	0.2M	1.0	82.3M	92.2	77.8
1.0	2.2M	0.75	550.7M	94.4	83.7
1.0	2.2M	0.5	244.9M	92.0	80.3
1.0	2.2M	0.25	61.5M	86.9	67.3
0.75	1.3M	0.75	321.7M	94.3	82.4
0.75	1.3M	0.5	143.1M	92.9	79.5
0.75	1.3M	0.25	35.9M	85.4	65.5
0.5	0.6M	0.75	153.6M	92.9	80.8
0.5	0.6M	0.5	68.3M	91.7	78.5
0.5	0.6M	0.25	17.2M	85.4	66.0
0.25	0.2M	0.75	46.3M	91.6	76.1
0.25	0.2M	0.5	20.6M	88.7	71.8
0.25	0.2M	0.25	5.2M	79.1	56.0

Table 6: Results (%) of varying width multiplier β and resolution multiplier γ for OSNet. For input size, $\gamma = 0.75$: 192×96 ; $\gamma = 0.5$: 128×64 ; $\gamma = 0.25$: 64×32 .

model). Therefore, adapting the scale fusion for individual input images is essential. (7) *Evaluation on stream cardinality*: The results are substantially improved from $T = 1$ (model 9) to $T = 2$ (model 10) and gradually progress to $T = 4$ (model 1).

Model shrinking hyper-parameters. We can trade-off between model size, computations and performance by adjusting the width multiplier β and the image resolution multiplier γ . Table 6 shows that by keeping one multiplier fixed and shrinking the other, the R1 drops off *smoothly*. It is worth noting that 92.2% R1 accuracy is obtained by a much shrunken version of OSNet with *merely 0.2M parameters and 82M mult-adds* ($\beta = 0.25$). Compared with the results in Table 3, we can see that the shrunken OSNet is still very competitive against the latest proposed models, most of which are $100\times$ bigger in size. This indicates that OSNet has a great potential for efficient deployment in resource-constrained devices such as a surveillance camera with an AI processor.

Visualisation of unified aggregation gate. As the gating vectors produced by the AG inherently encode the way how the omni-scale feature streams are aggregated, we can understand what the AG sub-network has learned by visualising images of similar gating vectors. To this end, we concatenate the gating vectors of four streams in the last bottleneck, perform k-means clustering on test images of Mix4, and select top-15 images closest to the cluster centres. Fig. 5 shows four example clusters where images within the same cluster exhibit similar patterns, i.e., combinations of global-scale and local-scale appearance.

Visualisation of attention. To understand how our designs

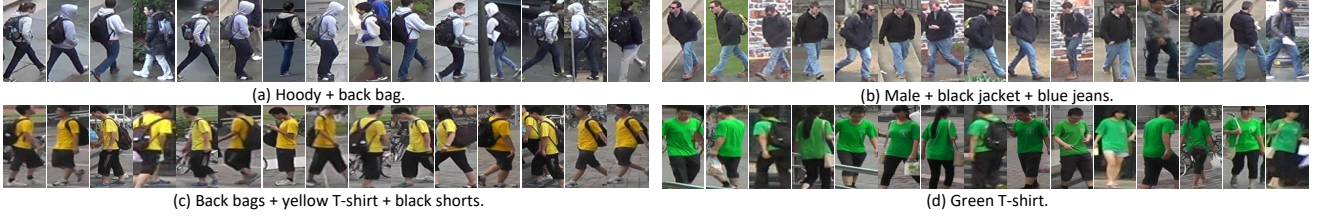


Figure 5: Image clusters of similar gating vectors. The visualisation shows that our unified aggregation gate is capable of learning the combination of homogeneous and heterogeneous scales in a dynamic manner.

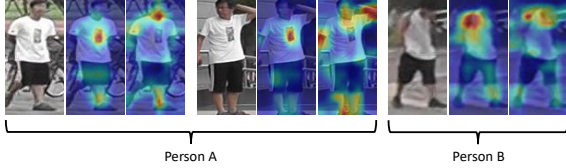


Figure 6: Each triplet contains, from left to right, original image, activation map of OSNet and activation map of single-scale baseline. These images indicate that OSNet can detect subtle differences between visually similar persons.

help OSNet learn discriminative features, we visualise the activations of the last convolutional feature maps to investigate where the network focuses on to extract features, i.e. attention. Following [75], the activation maps are computed as the sum of absolute-valued feature maps along the channel dimension followed by a spatial ℓ_2 normalisation. Fig. 6 compares the activation maps of OSNet and the single-scale baseline (model 9 in Table 5). It is clear that OSNet can capture the local discriminative patterns of Person A (e.g., the clothing logo) which distinguish Person A from Person B. In contrast, the single-scale model over-concentrates on the face region, which is unreliable for re-ID due to the low resolution of surveillance images. Therefore, this qualitative result shows that our multi-scale design and unified aggregation gate enable OSNet to identify subtle differences between visually similar persons – a vital requirement for accurate re-ID.

4.2. Evaluation on Person Attribute Recognition

Although person attribute recognition is a category-recognition problem, it is closely related to the person re-ID problem in that omni-scale feature learning is also critical: some attributes such as ‘view angle’ are global; others such as ‘wearing glasses’ are local; heterogeneous-scale features are also needed for recognising attributes such as ‘age’.

Datasets and settings. We use PA-100K [33], the largest person attribute recognition dataset. PA-100K contains 80K training images and 10K test images. Each image is annotated with 26 attributes, e.g., male/female, wearing glasses, carrying hand bag. Following [33], we adopt five evaluation metrics, including mean Accuracy (mA), and four instance-based metrics, namely Accuracy (Acc), Precision (Prec),

Method	PA-100K				
	mA	Acc	Prec	Rec	F1
DeepMar [24]	72.7	70.4	82.2	80.4	81.3
HydraPlusNet [33]	74.2	72.2	83.0	82.1	82.5
OSNet	74.6	76.0	88.3	82.5	85.3

Table 7: Results (%) on pedestrian attribute recognition.

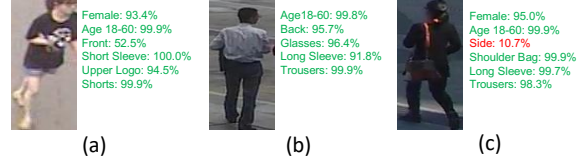


Figure 7: Likelihoods on ground-truth attributes predicted by OSNet. Correct/incorrect classifications based on threshold 50% are shown in green/red.

Recall (Rec) and F1-score (F1). Please refer to [26] for the detailed definitions.

Implementation details. A sigmoid-activated attribute prediction layer is added on the top of OSNet. Following [24, 33], we use the weighted multi-label classification loss for supervision. For data augmentation, we adopt random translation and mirroring. OSNet is trained from scratch with SGD, momentum of 0.9 and initial learning rate of 0.065 for 50 epochs. The learning rate is decayed by 0.1 at 30 and 40 epochs.

Results. Table 7 compares OSNet with two state-of-the-art methods [24, 33] on PA-100K. It can be seen that OSNet outperforms both alternatives on all five evaluation metrics. Fig. 7 provides some qualitative results. It shows that OSNet is particularly strong at predicting attributes that can only be inferred by examining features of heterogeneous scales such as age and gender.

4.3. Evaluation on CIFAR

Datasets and settings. CIFAR10/100 [23] has 50K training images and 10K test images, each with the size of 32×32 . OSNet is trained following the setting in [15, 74]. Apart from the default OSNet in Table 1, a deeper version is constructed by increasing the number of staged bottlenecks from 2-2-2 to 3-8-6. Error rate is reported as the metric.

Results. Table 8 compares OSNet with a number of state-

Method	Depth	# params	CIFAR10	CIFAR100
pre-act ResNet [15]	164	1.7M	5.46	24.33
pre-act ResNet [15]	1001	10.2M	4.92	22.71
Wide ResNet [74]	40	8.9M	4.97	22.89
Wide ResNet [74]	16	11.0M	4.81	22.07
DenseNet [21]	40	1.0M	5.24	24.42
DenseNet [21]	100	7.0M	4.10	20.20
OSNet	78	2.2M	4.41	19.21
OSNet	210	4.6M	4.18	18.88

Table 8: Error rates (%) on CIFAR datasets. All methods here use translation and mirroring for data augmentation. Pointwise and depthwise convolutions are counted as separate layers.

Architecture	CIFAR10	CIFAR100
$T = 1$	5.49	21.78
$T = 4 + \text{addition}$	4.72	20.24
$T = 4 + \text{unified AG}$	4.41	19.21

Table 9: Ablation study on OSNet on CIFAR10/100.

Method	β	# params	Mult-Adds	Top1
SqueezeNet [22]	1.0	1.2M	-	57.5
MobileNetV1 [18]	0.5	1.3M	149M	63.7
MobileNetV1 [18]	0.75	2.6M	325M	68.4
MobileNetV1 [18]	1.0	4.2M	569M	70.6
ShuffleNet [78]	1.0	2.4M	140M	67.6
ShuffleNet [78]	1.5	3.4M	292M	71.5
ShuffleNet [78]	2.0	5.4M	524M	73.7
MobileNetV2 [43]	1.0	3.4M	300M	72.0
MobileNetV2 [43]	1.4	6.9M	585M	74.7
OSNet (ours)	0.5	1.1M	424M	69.5
OSNet (ours)	0.75	1.8M	885M	73.5
OSNet (ours)	1.0	2.7M	1511M	75.5

Table 10: Single-crop top1 accuracy (%) on ImageNet-2012 validation set. β : width multiplier. M: Million.

of-the-art object recognition models. The results suggest that, although OSNet is originally designed for fine-grained object instance recognition task in re-ID, it is also highly competitive on object category recognition tasks. Note that CIFAR100 is more difficult than CIFAR10 because it contains ten times fewer training images per class (500 vs. 5,000). However, OSNet’s performance on CIFAR100 is stronger, indicating that it is better at capturing useful patterns with limited data, hence its excellent performance on the data-scarce re-ID benchmarks.

Ablation study. We compare our primary model with model 9 (single-scale baseline in Table 5) and model 5 (four streams + addition) on CIFAR10/100. Table 9 shows that both omni-scale feature learning and unified AG contribute positively to the overall performance of OSNet.

4.4. Evaluation on ImageNet

In this section, the results on the larger-scale ImageNet 1K category dataset (LSVRC-2012 [7]) are presented.

Implementation. OSNet is trained with SGD, initial learning rate of 0.4, batch size of 1024 and weight decay of $4e-5$ for 120 epochs. For data augmentation, we use random 224×224 crops on 256×256 images and random mirroring. To benchmark, we report single-crop⁶ top1 accuracy on the LSVRC-2012 validation set [7].

Results. Table 10 shows that OSNet outperforms the alternative lightweight models by a clear margin. In particular OSNet $\times 1.0$ surpasses MobiltNetV2 $\times 1.0$ by 3.5% and MobiltNetV2 $\times 1.4$ by 0.8%. It is noteworthy that MobiltNetV2 $\times 1.4$ is around $2.5\times$ larger than our OSNet $\times 1.0$. OSNet $\times 0.75$ performs on par with ShuffleNet $\times 2.0$ and outperforms ShuffleNet $\times 1.5/\times 1.0$ by 2.0%/5.9%. These results give a strong indication that OSNet has a great potential for a broad range of visual recognition tasks. Note that although the model size is smaller, our OSNet does have a higher number of multi-adds operations than its main competitors. This is mainly due to the multi-stream design. However, if both model size and number of Multi-Adds need to be small for a certain application, we can reduce the latter by introducing pointwise convolutions with group convolutions and channel shuffling [78]. The overall results on CIFAR and ImageNet show that omni-scale feature learning is beneficial beyond re-ID and should be considered for a broad range of visual recognition tasks.

5. Conclusion

We presented OSNet, a lightweight CNN architecture that is capable of learning omni-scale feature representations. Extensive experiments on six person re-ID datasets demonstrated that OSNet achieved state-of-the-art performance, despite its lightweight design. The superior performance on object categorisation tasks and a multi-label attribute recognition task further suggested that OSNet is of wide interest to visual recognition beyond re-ID.

Supplementary

The results in the main paper have been presented at ICCV’19. In this supplementary, we show additional results to further demonstrate the stength of OSNet.

A. A Strong Backbone for Cross-Domain Re-ID

In this section, we construct a strong backbone model for cross-domain re-ID based on OSNet. Following [36], we add instance normalisation (IN) [58] to the lower layers (conv1, conv2) in OSNet. Specifically, IN is inserted

⁶ 224×224 centre crop from 256×256 .

Method	Source	Target: Duke				Source	Target: Market1501			
		R1	R5	R10	mAP		R1	R5	R10	mAP
MMFA [31]	Market1501 + Duke (<i>U</i>)	45.3	59.8	66.3	24.7	Duke + Market1501 (<i>U</i>)	56.7	75.0	81.8	27.4
SPGAN [8]	Market1501 + Duke (<i>U</i>)	46.4	62.3	68.0	26.2	Duke + Market1501 (<i>U</i>)	57.7	75.8	82.4	26.7
TJ-AIDL [62]	Market1501 + Duke (<i>U</i>)	44.3	59.6	65.0	23.0	Duke + Market1501 (<i>U</i>)	58.2	74.8	81.1	26.5
ATNet [32]	Market1501 + Duke (<i>U</i>)	45.1	59.5	64.2	24.9	Duke + Market1501 (<i>U</i>)	55.7	73.2	79.4	25.6
CamStyle [90]	Market1501 + Duke (<i>U</i>)	48.4	62.5	68.9	25.1	Duke + Market1501 (<i>U</i>)	58.8	78.2	84.3	27.4
HHL [88]	Market1501 + Duke (<i>U</i>)	46.9	61.0	66.7	27.2	Duke + Market1501 (<i>U</i>)	62.2	78.8	84.0	31.4
ECN [89]	Market1501 + Duke (<i>U</i>)	63.3	75.8	80.4	40.4	Duke + Market1501 (<i>U</i>)	75.1	87.6	91.6	43.0
OSNet-IBN (ours)	Market1501	48.5	62.3	67.4	26.7	Duke	57.7	73.7	80.0	26.1
MAR [72]	MSMT17+Duke (<i>U</i>)	67.1	79.8	-	48.0	MSMT17+Market1501 (<i>U</i>)	67.7	81.9	-	40.0
PAUL [70]	MSMT17+Duke (<i>U</i>)	72.0	82.7	86.0	53.2	MSMT17+Market1501 (<i>U</i>)	68.5	82.4	87.4	40.1
OSNet-IBN (ours)	MSMT17	67.4	80.0	83.3	45.6	MSMT17	66.5	81.5	86.8	37.2

Table 11: Cross-domain re-ID results. It is worth noting that OSNet-IBN (highlighted rows), without using any target data, can achieve competitive performance with state-of-the-art unsupervised domain adaptation re-ID methods. *U*: Unlabelled.

after the residual connection and before the ReLU function in a bottleneck. It has been shown in [36] that IN can improve the generalisation performance on cross-domain semantic segmentation tasks. Here we apply the same idea to OSNet and show that we can build a strong backbone model for cross-domain re-ID. We call this new network OSNet-IBN.

Settings. Following the recent works [89, 90, 32, 72, 70], we choose Market1501 and Duke as the target datasets. The source dataset is either Market1501, Duke or MSMT17⁷. Models are trained on labelled source data and directly tested on target data.

Implementation details. Similar to the conventional setting, we use cross-entropy loss as the objective function. We train OSNet-IBN with AMSGrad [41], batch size of 64, weight decay of 5e-4 and initial learning rate of 0.0015 for 150 epochs. The learning rate is decayed by 0.1 every 60 epochs. During the first 10 epochs, only the randomly initialised classification layer is open for training while the ImageNet pre-trained base network is frozen. All images are resized to 256×128 . Data augmentation includes random flip and color jittering. We observed that random erasing [87] dramatically decreased the cross-domain results so we did not use it.

Results. Table 11 compares OSNet-IBN with current state-of-the-art unsupervised domain adaptation (UDA) methods. It is clear that OSNet-IBN achieves highly competitive performance or even better results than some UDA methods on the target datasets, despite *only using source data for training*. In particular, on Market1501→Duke (at R1), OSNet-IBN beats all the UDA methods except ECN; on MSMT17→Duke, OSNet-IBN performs on par with MAR; on MSMT17→Market1501, OSNet-IBN obtains compara-

ble results with MAR and PAUL. These results demonstrate that our OSNet-IBN, with a minor modification, can be used as a strong backbone model for cross-domain re-ID⁸.

B. Training Recipes for Practitioners

We investigate some training methods in order to further improve OSNet’s performance. We not only show the methods that work, but also discuss what do not work in our experiments.

Implementation. We train the baseline OSNet following [92], where the main difference compared with the conference version is the use of cosine annealing strategy [34] to decay the learning rate. For image matching, we use cosine distance. To make sure the result is convincing, we run *every* experiment with 3 different random seeds and report the mean and standard deviation. We choose Market1501 and Duke for benchmarking.

Dataset-specific normalisation parameters. Most re-ID papers used the ImageNet mean and standard deviation for pixel normalisation, without justifying whether using dataset-specific statistics is a better choice. Typically, images from re-ID datasets exhibit drastic differences compared with the natural images from ImageNet, e.g., the person images for re-ID are usually of poor quality and blurred. Therefore, using the statistics from re-ID dataset for pixel normalisation seems to make more sense. However, Table 12a shows that the difference in performance is subtle, suggesting that collecting dataset-specific statistics might be unnecessary. In practice, we do, however, encourage practitioners to try both ways for their own datasets.

Will larger input size help? Table 12b shows that using larger input size improves the performance, but only marginally. This is because OSNet can learn omni-scale

⁷Following [72, 70], all 126,441 images of 4,101 identities in MSMT17 are used for training.

⁸See [92] for an improved OSNet-IBN (called OSNet-AIN [92]) which achieves better cross-domain performance via neural architecture search.

Mean & std from	Market1501		Duke	
	R1	mAP	R1	mAP
ImageNet	94.6±0.1	86.5±0.2	88.6±0.3	76.6±0.1
Re-ID dataset	94.4±0.0	86.3±0.1	88.5±0.1	76.5±0.2

(a) Pixel normalisation parameters

Input size	Market1501		Duke	
	R1	mAP	R1	mAP
256×128	94.6±0.1	86.5±0.2	88.6±0.3	76.6±0.1
320×160	94.9±0.1	86.9±0.1	88.5±0.5	76.8±0.2

(b) Input size

λ_e	Market1501		Duke	
	R1	mAP	R1	mAP
0	94.6±0.1	86.5±0.2	88.6±0.3	76.6±0.1
0.01	94.6±0.1	86.4±0.1	88.5±0.5	76.5±0.3
0.05	94.5±0.1	86.5±0.2	88.7±0.2	76.6±0.0
0.1	94.7±0.1	86.4±0.3	88.4±0.1	76.7±0.2
0.5	94.7±0.1	86.6±0.2	88.3±0.2	76.7±0.2

(c) Regularisation with entropy maximisation: $\mathcal{L}_{ID} - \lambda_e \mathcal{L}_{Entropy}$

	Market1501		Duke	
	R1	mAP	R1	mAP
w/o DML	94.6±0.1	86.5±0.2	88.6±0.3	76.6±0.1
w/ DML model-1	94.7±0.1	87.2±0.0	88.4±0.4	77.3±0.2
w/ DML model-2	94.8±0.1	87.3±0.0	88.5±0.8	77.3±0.2
w/ DML model-1+2	94.9±0.1	87.8±0.0	88.7±0.6	78.0±0.2

(d) Deep mutual learning and model ensemble

λ_t	Market1501		Duke	
	R1	mAP	R1	mAP
0	94.6±0.1	86.5±0.2	88.6±0.3	76.6±0.1
0.1	94.7±0.4	86.7±0.2	88.3±0.3	76.9±0.1
0.5	95.5±0.1	87.2±0.0	88.6±0.2	77.3±0.1
1.0	94.9±0.1	86.9±0.1	88.4±0.1	76.8±0.4

(e) Auxiliary loss with hard example-mining triplet loss: $\mathcal{L}_{ID} + \lambda_t \mathcal{L}_{Triplet}$

	Market1501		Duke	
	R1	mAP	R1	mAP
$\lambda_t = 0$ w/o DML	94.6±0.1	86.5±0.2	88.6±0.3	76.6±0.1
$\lambda_t = 0.5$ + DML model-1	95.7±0.1	88.1±0.2	89.1±0.2	78.4±0.0
$\lambda_t = 0.5$ + DML model-2	95.5±0.2	88.0±0.1	89.6±0.3	78.5±0.2
$\lambda_t = 0.5$ + DML model-1+2	95.7±0.1	88.7±0.1	89.0±0.0	79.5±0.1

(f) $\mathcal{L}_{Triplet}$ + deep mutual learning + model ensemble

Table 12: Investigation on various training methods for improving OSNet’s performance. All experiments are run for 3 times with different random seeds. Note that the implementation follows [92], which is slightly different from the conference version.

features, which are insensitive to the input size. Considering that using 320×160 increases the flops from 978.9M to 1,529.3M, we suggest using 256×128 .

Entropy maximisation. As re-ID datasets are small-scale, we add a entropy maximisation term [37] to further regularise the network (this term penalises confident predictions). The results are shown in Table 12c where we observe that this new regularisation term, with various balancing weights, has little effect on the performance.

Deep mutual learning (DML). Zhang et al. [79] has shown that DML can achieve notable improvement for re-ID (when using MobileNet [18]). We apply DML to training OSNet and report the results in Table 12d. It is clear that DML improves the mAP. This indicates that features learned with DML are more discriminative. As DML trains two networks simultaneously, it is natural to try model ensemble with these two networks. The results (last row in Table 12d) show clear improvements on both rank-1 and mAP. Note that when doing ensemble, we concatenate the features rather than performing mean-pooling. The latter makes more sense for classification tasks but not for retrieval tasks where features are used.

Auxiliary loss. Several recent re-ID approaches [40, 6, 51]

adopt a multi-loss training strategy, e.g., using both cross-entropy loss and triplet loss [16]. We investigate such training strategy for OSNet where a balancing weight λ_t is added to scale the triplet loss (see the caption of Table 12e). Table 12e shows that the triplet loss improves the performance when λ_t is carefully tuned. In practice, we encourage practitioners to use the cross-entropy loss as the main objective and the triplet loss as an auxiliary loss with a balancing weight (which needs to be tuned).

Combination. We combine the effective training techniques, i.e. DML and auxiliary loss learning (with the triplet loss), and show the results in Table 12f. It can be observed that the improvement is larger than that of using either technique alone. The best performance is obtained by fusing the two DML-trained models.

Therefore, we suggest training OSNet with cross-entropy loss + triplet loss ($\lambda_t = 0.5$ as the rule of thumb) + DML and testing with model ensemble.

References

- [1] Ejaz Ahmed, Michael Jones, and Tim K Marks. An improved deep learning architecture for person re-identification. In *CVPR*, 2015. 2

- [2] Xiaobin Chang, Timothy M Hospedales, and Tao Xiang. Multi-level factorisation net for person re-identification. In *CVPR*, 2018. 2, 3, 5, 6
- [3] Dapeng Chen, Dan Xu, Hongsheng Li, Nicu Sebe, and Xiaogang Wang. Group consistent similarity learning via deep crf for person re-identification. In *CVPR*, 2018. 6
- [4] Yanbei Chen, Xiatian Zhu, and Shaogang Gong. Person re-identification by deep learning multi-scale representations. In *ICCVW*, 2017. 6
- [5] Francois Chollet. Xception: Deep learning with depthwise separable convolutions. In *CVPR*, 2017. 3
- [6] Zuozhuo Dai, Mingqiang Chen, Xiaodong Gu, Siyu Zhu, and Ping Tan. Batch dropblock network for person re-identification and beyond. In *ICCV*, 2019. 11
- [7] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *CVPR*, 2009. 2, 9
- [8] Weijian Deng, Liang Zheng, Qixiang Ye, Guoliang Kang, Yi Yang, and Jianbin Jiao. Image-image domain adaptation with preserved self-similarity and domain-dissimilarity for person re-identification. In *CVPR*, 2018. 10
- [9] Ruth Fong and Andrea Vedaldi. Net2vec: Quantifying and explaining how concepts are encoded by filters in deep neural networks. In *CVPR*, 2018. 7
- [10] Yang Fu, Yunchao Wei, Yuqian Zhou, Honghui Shi, Gao Huang, Xinchao Wang, Zhiqiang Yao, and Thomas Huang. Horizontal pyramid matching for person re-identification. In *AAAI*, 2019. 2
- [11] Yixiao Ge, Zhuowan Li, Haiyu Zhao, Guojun Yin, Shuai Yi, Xiaogang Wang, and Hongsheng Li. Fd-gan: Pose-guided feature distilling gan for robust person re-identification. In *NeurIPS*, 2018. 6
- [12] Douglas Gray, Shane Brennan, and Hai Tao. Evaluating appearance models for recognition, reacquisition, and tracking. In *PETS*, 2007. 5
- [13] Yiluan Guo and Ngai-Man Cheung. Efficient and deep person re-identification using multi-level similarity. In *CVPR*, 2018. 2
- [14] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016. 2, 3
- [15] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Identity mappings in deep residual networks. In *ECCV*, 2016. 8, 9
- [16] Alexander Hermans, Lucas Beyer, and Bastian Leibe. In defense of the triplet loss for person re-identification. *arXiv preprint arXiv:1703.07737*, 2017. 11
- [17] Ruibing Hou, Bingpeng Ma, Hong Chang, Xinqian Gu, Shiguang Shan, and Xilin Chen. Interaction-and-aggregation network for person re-identification. In *CVPR*, 2019. 6
- [18] Andrew G Howard, Menglong Zhu, Bo Chen, Dmitry Kalenichenko, Weijun Wang, Tobias Weyand, Marco Andreetto, and Hartwig Adam. Mobilenets: Efficient convolutional neural networks for mobile vision applications. *arXiv preprint arXiv:1704.04861*, 2017. 2, 3, 4, 9, 11
- [19] Jie Hu, Li Shen, and Gang Sun. Squeeze-and-excitation networks. In *CVPR*, 2018. 4, 5
- [20] Gao Huang, Shichen Liu, Laurens van der Maaten, and Kilian Q. Weinberger. Condensenet: An efficient densenet using learned group convolutions. In *CVPR*, 2018. 3
- [21] Gao Huang, Zhuang Liu, Kilian Q Weinberger, and Laurens van der Maaten. Densely connected convolutional networks. In *CVPR*, 2017. 9
- [22] Forrest N Iandola, Song Han, Matthew W Moskewicz, Khalid Ashraf, William J Dally, and Kurt Keutzer. Squeezenet: Alexnet-level accuracy with 50x fewer parameters and 0.5 mb model size. *arXiv preprint arXiv:1602.07360*, 2016. 3, 9
- [23] Alex Krizhevsky and Geoffrey Hinton. Learning multiple layers of features from tiny images. Technical report, Cite-seer, 2009. 2, 8
- [24] Dangwei Li, Xiaotang Chen, and Kaiqi Huang. Multi-attribute learning for pedestrian attribute recognition in surveillance scenarios. In *ACPR*, 2015. 8
- [25] Dangwei Li, Xiaotang Chen, Zhang Zhang, and Kaiqi Huang. Learning deep context-aware features over body and latent parts for person re-identification. In *CVPR*, 2017. 2
- [26] Dangwei Li, Zhang Zhang, Xiaotang Chen, Haibin Ling, and Kaiqi Huang. A richly annotated dataset for pedestrian attribute recognition. *TIP*, 2016. 8
- [27] Wei Li, Rui Zhao, Tong Xiao, and Xiaogang Wang. Deepreid: Deep filter pairing neural network for person re-identification. In *CVPR*, 2014. 2, 5
- [28] Wei Li, Xiatian Zhu, and Shaogang Gong. Person re-identification by deep joint learning of multi-loss classification. In *IJCAI*, 2017. 2, 5, 6, 7
- [29] Wei Li, Xiatian Zhu, and Shaogang Gong. Harmonious attention network for person re-identification. In *CVPR*, 2018. 2, 5, 6
- [30] Min Lin, Qiang Chen, and Shuicheng Yan. Network in network. *arXiv preprint arXiv:1312.4400*, 2013. 4
- [31] Shan Lin, Haoliang Li, Chang-Tsun Li, and Alex Chichung Kot. Multi-task mid-level feature alignment network for unsupervised cross-dataset person re-identification. In *BMVC*, 2018. 10
- [32] Jiawei Liu, Zheng-Jun Zha, Di Chen, Richang Hong, and Meng Wang. Adaptive transfer network for cross-domain person re-identification. In *CVPR*, 2019. 10
- [33] Xihui Liu, Haiyu Zhao, Maoqing Tian, Lu Sheng, Jing Shao, Shuai Yi, Junjie Yan, and Xiaogang Wang. Hydraplus-net: Attentive deep features for pedestrian analysis. In *ICCV*, 2017. 2, 3, 5, 7, 8
- [34] Ilya Loshchilov and Frank Hutter. Sgdr: Stochastic gradient descent with warm restarts. In *ICLR*, 2017. 10
- [35] Chen Change Loy, Tao Xiang, and Shaogang Gong. Multi-camera activity correlation analysis. In *CVPR*, 2009. 5
- [36] Xingang Pan, Ping Luo, Jianping Shi, and Xiaoou Tang. Two at once: Enhancing learning and generalization capacities via ibn-net. In *ECCV*, 2018. 9, 10
- [37] Gabriel Pereyra, George Tucker, Jan Chorowski, Łukasz Kaiser, and Geoffrey Hinton. Regularizing neural networks by penalizing confident output distributions. *arXiv preprint arXiv:1701.06548*, 2017. 11

- [38] Xuelin Qian, Yanwei Fu, Yu-Gang Jiang, Tao Xiang, and Xiangyang Xue. Multi-scale deep learning architectures for person re-identification. In *ICCV*, 2017. 2, 3, 7
- [39] Xuelin Qian, Yanwei Fu, Tao Xiang, Wenxuan Wang, Jie Qiu, Yang Wu, Yu-Gang Jiang, and Xiangyang Xue. Pose-normalized image generation for person re-identification. In *ECCV*, 2018. 6
- [40] Ruijie Quan, Xuanyi Dong, Yu Wu, Linchao Zhu, and Yi Yang. Auto-reid: Searching for a part-aware convnet for person re-identification. In *ICCV*, 2019. 11
- [41] Sashank J. Reddi, Satyen Kale, and Sanjiv Kumar. On the convergence of adam and beyond. In *International Conference on Learning Representations*, 2018. 5, 10
- [42] Ergys Ristani, Francesco Solera, Roger Zou, Rita Cucchiara, and Carlo Tomasi. Performance measures and a data set for multi-target, multi-camera tracking. In *ECCVW*, 2016. 5
- [43] Mark Sandler, Andrew Howard, Menglong Zhu, Andrey Zhmoginov, and Liang-Chieh Chen. Mobilenetv2: Inverted residuals and linear bottlenecks. In *CVPR*, 2018. 2, 3, 4, 6, 9
- [44] Yantao Shen, Hongsheng Li, Tong Xiao, Shuai Yi, Dapeng Chen, and Xiaogang Wang. Deep group-shuffling random walk for person re-identification. In *CVPR*, 2018. 6
- [45] Yantao Shen, Hongsheng Li, Shuai Yi, Dapeng Chen, and Xiaogang Wang. Person re-identification with deep similarity-guided graph neural network. In *ECCV*, 2018. 6
- [46] Yantao Shen, Tong Xiao, Hongsheng Li, Shuai Yi, and Xiaogang Wang. End-to-end deep kronecker-product matching for person re-identification. In *CVPR*, 2018. 2, 6
- [47] Jianlou Si, Honggang Zhang, Chun-Guang Li, Jason Kuen, Xiangfei Kong, Alex C. Kot, and Gang Wang. Dual attention matching network for context-aware feature sequence based person re-identification. In *CVPR*, 2018. 2, 6
- [48] Chunfeng Song, Yan Huang, Wanli Ouyang, and Liang Wang. Mask-guided contrastive attention model for person re-identification. In *CVPR*, 2018. 2
- [49] Chi Su, Jianing Li, Shiliang Zhang, Junliang Xing, Wen Gao, and Qi Tian. Pose-driven deep convolutional model for person re-identification. In *ICCV*, 2017. 2, 6
- [50] Arulkumar Subramaniam, Moitrey Chatterjee, and Anurag Mittal. Deep neural networks with inexact matching for person re-identification. In *NIPS*, 2016. 2
- [51] Arulkumar Subramaniam, Athira Nambiar, and Anurag Mittal. Co-segmentation inspired attention networks for video-based person re-identification. In *ICCV*, 2019. 11
- [52] Yumin Suh, Jingdong Wang, Siyu Tang, Tao Mei, and Kyoung Mu Lee. Part-aligned bilinear representations for person re-identification. In *ECCV*, 2018. 2, 6
- [53] Yifan Sun, Liang Zheng, Yi Yang, Qi Tian, and Shengjin Wang. Beyond part models: Person retrieval with refined part pooling (and a strong convolutional baseline). In *ECCV*, 2018. 2, 6
- [54] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. In *CVPR*, 2015. 3, 4
- [55] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision. In *CVPR*, 2016. 3, 5
- [56] Chiat-Pin Tay, Sharmili Roy, and Kim-Hui Yap. Aanet: Attribute attention network for person re-identifications. In *CVPR*, 2019. 6
- [57] Maoqing Tian, Shuai Yi, Hongsheng Li, Shihua Li, Xuesen Zhang, Jianping Shi, Junjie Yan, and Xiaogang Wang. Eliminating background-bias for robust person re-identification. In *CVPR*, 2018. 2
- [58] Dmitry Ulyanov, Andrea Vedaldi, and Victor Lempitsky. Improved texture networks: Maximizing quality and diversity in feed-forward stylization and texture synthesis. In *CVPR*, 2017. 9
- [59] Rahul Rama Varior, Mrinal Haloi, and Gang Wang. Gated siamese convolutional neural network architecture for human re-identification. In *ECCV*, 2016. 2
- [60] Cheng Wang, Qian Zhang, Chang Huang, Wenyu Liu, and Xinggang Wang. Mancs: A multi-task attentional network with curriculum sampling for person re-identification. In *ECCV*, 2018. 6
- [61] Guanshuo Wang, Yufeng Yuan, Xiong Chen, Jiwei Li, and Xi Zhou. Learning discriminative features with multiple granularities for person re-identification. In *ACM MM*, 2018. 2
- [62] Jingya Wang, Xiatian Zhu, Shaogang Gong, and Wei Li. Transferable joint attribute-identity deep learning for unsupervised person re-identification. In *CVPR*, 2018. 10
- [63] Yicheng Wang, Zhenzhong Chen, Feng Wu, and Gang Wang. Person re-identification with cascaded pairwise convolutions. In *CVPR*, 2018. 2, 6
- [64] Yan Wang, Lequn Wang, Yurong You, Xu Zou, Vincent Chen, Serena Li, Gao Huang, Bharath Hariharan, and Kilian Q Weinberger. Resource aware person re-identification across multiple resolutions. In *CVPR*, 2018. 3, 6
- [65] Longhui Wei, Shiliang Zhang, Wen Gao, and Qi Tian. Person transfer gan to bridge domain gap for person re-identification. In *CVPR*, 2018. 5
- [66] Longhui Wei, Shiliang Zhang, Hantao Yao, Wen Gao, and Qi Tian. Glad: global-local-alignment descriptor for pedestrian retrieval. In *ACM MM*, 2017. 5, 6, 7
- [67] Sanghyun Woo, Jongchan Park, Joon-Young Lee, and In So Kweon. Cbam: Convolutional block attention module. In *ECCV*, 2018. 4
- [68] Saining Xie, Ross Girshick, Piotr Dollár, Zhuowen Tu, and Kaiming He. Aggregated residual transformations for deep neural networks. In *CVPR*, 2017. 3, 4
- [69] Jing Xu, Rui Zhao, Feng Zhu, Huaming Wang, and Wanli Ouyang. Attention-aware compositional network for person re-identification. In *CVPR*, 2018. 2
- [70] Qize Yang, Hong-Xing Yu, Ancong Wu, and Wei-Shi Zheng. Patch-based discriminative feature learning for unsupervised person re-identification. In *CVPR*, 2019. 10
- [71] Wenjie Yang, Houjing Huang, Zhang Zhang, Xiaotang Chen, Kaiqi Huang, and Shu Zhang. Towards rich feature discovery with class activation maps augmentation for person re-identification. In *CVPR*, 2019. 2, 6

- [72] Hong-Xing Yu, Wei-Shi Zheng, Ancong Wu, Xiaowei Guo, Shaogang Gong, and Jian-Huang Lai. Unsupervised person re-identification by soft multilabel learning. In *CVPR*, 2019. 10
- [73] Qian Yu, Xiaobin Chang, Yi-Zhe Song, Tao Xiang, and Timothy M Hospedales. The devil is in the middle: Exploiting mid-level representations for cross-domain instance matching. *arXiv preprint arXiv:1711.08106*, 2017. 3
- [74] Sergey Zagoruyko and Nikos Komodakis. Wide residual networks. In *BMVC*, 2016. 8, 9
- [75] Sergey Zagoruyko and Nikos Komodakis. Paying more attention to attention: Improving the performance of convolutional neural networks via attention transfer. In *ICLR*, 2017. 8
- [76] Ting Zhang, Guo-Jun Qi, Bin Xiao, and Jingdong Wang. Interleaved group convolutions. In *ICCV*, 2017. 3
- [77] Xuan Zhang, Hao Luo, Xing Fan, Weilai Xiang, Yixiao Sun, Qiqi Xiao, Wei Jiang, Chi Zhang, and Jian Sun. Alignedreid: Surpassing human-level performance in person re-identification. *arXiv preprint arXiv:1711.08184*, 2017. 2
- [78] Xiangyu Zhang, Xinyu Zhou, Mengxiao Lin, and Jian Sun. Shufflenet: An extremely efficient convolutional neural network for mobile devices. In *CVPR*, 2018. 3, 4, 6, 9
- [79] Ying Zhang, Tao Xiang, Timothy M. Hospedales, and Huchuan Lu. Deep mutual learning. In *CVPR*, 2018. 11
- [80] Zhizheng Zhang, Cuiling Lan, Wenjun Zeng, and Zhibo Chen. Densely semantically aligned person re-identification. In *CVPR*, 2019. 2
- [81] Haiyu Zhao, Maoqing Tian, Shuyang Sun, Jing Shao, Junjie Yan, Shuai Yi, Xiaogang Wang, and Xiaoou Tang. Spindle net: Person re-identification with human body region guided feature decomposition and fusion. In *CVPR*, 2017. 2, 5, 7
- [82] Liming Zhao, Xi Li, Yueting Zhuang, and Jingdong Wang. Deeply-learned part-aligned representations for person re-identification. In *ICCV*, 2017. 2, 5, 7
- [83] Liang Zheng, Liyue Shen, Lu Tian, Shengjin Wang, Jingdong Wang, and Qi Tian. Scalable person re-identification: A benchmark. In *ICCV*, 2015. 5
- [84] Zhedong Zheng, Xiaodong Yang, Zhiding Yu, Liang Zheng, Yi Yang, and Jan Kautz. Joint discriminative and generative learning for person re-identification. In *CVPR*, 2019. 2, 6
- [85] Zhedong Zheng, Liang Zheng, and Yi Yang. Unlabeled samples generated by gan improve the person re-identification baseline in vitro. In *ICCV*, 2017. 5
- [86] Zhun Zhong, Liang Zheng, Donglin Cao, and Shaozi Li. Re-ranking person re-identification with k-reciprocal encoding. In *CVPR*, 2017. 5
- [87] Zhun Zhong, Liang Zheng, Guoliang Kang, Shaozi Li, and Yi Yang. Random erasing data augmentation. *arXiv preprint arXiv:1708.04896*, 2017. 5, 10
- [88] Zhun Zhong, Liang Zheng, Shaozi Li, and Yi Yang. Generalizing a person retrieval model hetero- and homogeneously. In *ECCV*, 2018. 10
- [89] Zhun Zhong, Liang Zheng, Zhiming Luo, Shaozi Li, and Yi Yang. Invariance matters: Exemplar memory for domain adaptive person re-identification. In *CVPR*, 2019. 10
- [90] Zhun Zhong, Liang Zheng, Zhedong Zheng, Shaozi Li, and Yi Yang. Camstyle: A novel data augmentation method for person re-identification. *TIP*, 2019. 10
- [91] Kaiyang Zhou and Tao Xiang. Torchreid: A library for deep learning person re-identification in pytorch. *arXiv preprint arXiv:1910.10093*, 2019. 2, 5
- [92] Kaiyang Zhou, Yongxin Yang, Andrea Cavallaro, and Tao Xiang. Learning generalisable omni-scale representations for person re-identification. *arXiv preprint arXiv:1910.06827*, 2019. 10, 11