

Multi-Domain Joint Training for Person Re-Identification

Lu Yang, Lingqiao Liu, Yunlong Wang, Peng Wang, and Yanning Zhang

Abstract—Deep learning-based person Re-Identification (ReID) often requires a large amount of training data to achieve good performance. Thus it appears that collecting more training data from diverse environments tends to improve the ReID performance. This paper re-examines this common belief and makes a somehow surprising observation: using more samples, i.e., training with samples from multiple datasets, does not necessarily lead to better performance by using the popular ReID models. In some cases, training with more samples may even hurt the performance of the evaluation is carried out in one of those datasets. We postulate that this phenomenon is due to the incapability of the standard network in adapting to diverse environments. To overcome this issue, we propose an approach called **Domain-Camera-Sample Dynamic network (DCSD)** whose parameters can be adaptive to various factors. Specifically, we consider the **internal domain-related factor** that can be identified from the input features, and **external domain-related factors**, such as domain information or camera information. Our discovery is that training with such an adaptive model can better benefit from more training samples. Experimental results show that our DCSD can greatly boost the performance (up to 12.3%) while joint training in multiple datasets.

Index Terms—Person Re-identification, Dynamic Convolution, Multi-Domain Joint Training, Domain Conflict.

I. INTRODUCTION

Due to the urgent demand for public security and the increasing number of surveillance cameras on campus, parks, streets, and shopping malls, person Re-Identification (ReID) embraces many applications in intelligent video surveillance and presents a significant challenge in computer vision. ReID, as an important application of deep metric learning, aims to retrieve the images with the same identity of queries. The common solution is to construct an embedding space such that samples with identical identities (IDs) are gathered while samples with different IDs are well separated. Presently, deep learning methods dominate this community, with convincing superiority against hand-crafted competitors. The development of deep convolution network [1], [2] introduces a more powerful representation method for pedestrian images, which boosts the performance of ReID to a high level.

In real scenarios, there may be few person images in some special domains. In order to improve the performance of the models deployed in the domain, researchers may use more person images collected from other domains for joint training. Please note that the model is jointly trained on multiple domains during training, but only deployed on one of the source domains during the test phase. Generally speaking, the performance of deep convolution network on vision tasks increases based on the volume of training data size [3]. That means the more data, the better performance should be.

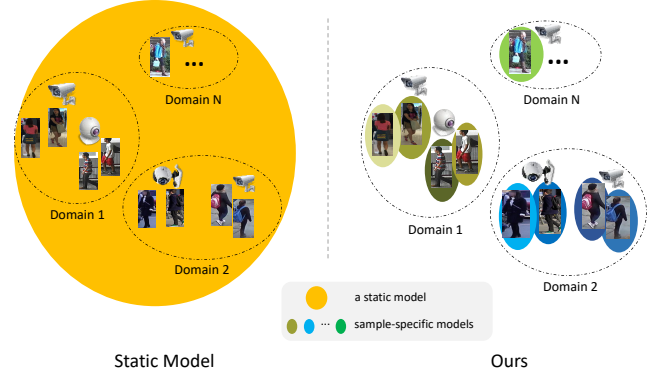


Fig. 1. Data in different domains usually have large data distribution shifts, it will be difficult for a traditional static model to deal with the data of all domains at the same time. The static network needs to learn a very powerful feature extraction ability to handle these datasets, it brings a heavy learning burden to the network. Unlike the traditional fixed network, DCSD uses internal domain-related factor, such as features from the sample, and external domain-related factors, such as dataset IDs and camera IDs, to dynamically generate the sample-specific network. It can reduce the learning burden of the model and makes the model easier to learn. Each colored ellipse represents a model, best viewed in color.

However, as Table II shows, when multiple domains are jointly trained, the performance of person ReID is even worse than that of each domain independently trained. We call it **domain conflict problem**. The domain conflicts may come from three different aspects: **datasets (domain)**, **cameras**, and **samples**. Different person ReID datasets are collected from different regions and times, resulting in great differences in appearance. As shown in Figure 2, people in CUHK-SYSU wear cool clothes and often go to shopping malls, but people in MSMT17 wear warm clothes. [4] highlight the presence of **camera-level sub-domains** as a unique characteristic in person ReID. And some works [5] treat each sample as an independent micro domain. Extracting robust feature representation is a key challenge in ReID. However, it is difficult for a traditional static network to extract robust feature representations in all domains. Because once the traditional convolution network is trained, the network parameters are fixed. Due to the domain conflict problem, it will be difficult for a traditional approach to deal with the data of all domains at the same time, especially when the data distribution of these domains is very different.

In this work, we propose a **Domain-Camera-Sample Dynamic network (DCSD)** to solve the domain conflict problem. When there is little training data in the current domain, DCSD can combine the data of other domains for joint training without domain conflict problems, so as to

TABLE I

THE STATISTICS OF PERSON RE-IDENTIFICATION DATASETS IN OUR EXPERIMENTS. “*” DENOTES THAT WE MODIFIED THE DATASET BY USING THE GROUND-TRUTH BOUNDING BOX ANNOTATION FOR OUR EXPERIMENTS RATHER THAN USING THE ORIGINAL IMAGES WHICH WERE ORIGINALLY USED FOR PERSON SEARCH EVALUATION.

	Training Set			Test Set (Query)			Test Set (Gallery)		
	ID#	Image#	Camera#	ID#	Image#	Camera#	ID#	Image#	Camera#
Market1501	751	12,936	6	750	3,368	6	751	15,913	6
CUHK-SYSU*	942	4,374	1	2,900	2,900	1	2,900	5,447	1
Duke	702	16,522	8	702	2,228	8	1,110	17,661	8
CUHK03	767	7,368	2	700	1,400	2	700	5,328	2
MSMT17	1,041	30,248	15	3,060	11,657	15	3,060	82,161	15

improve the performance of the model when deployed in the current domain. Unlike the fixed network parameters, DCSD can simultaneously consider the domain, camera, and sample information to generate the sample-specific network. We use three different levels of domain-related information to generate network parameters, the key insight is that datasets, cameras, and samples are different levels of factors that lead to shifts in data distribution. For example, there are location and season bias among different datasets, illumination and angle bias among different cameras, and micro-specific bias among samples.

In summary, the contribution of this study is three-fold:

- In this work, we deliver some surprising findings that the performance of joint training in multiple-person ReID datasets is not as effective as that of individual training in each domain.
- We propose a Domain-Camera-Sample Dynamic network (DCSD) to solve the domain conflict problem. It simultaneously considering the domain, camera, and sample information to generate the sample-specific network.
- By conducting extensive experiments on various person ReID datasets, we demonstrate the superior performance of the proposed DCSD and show that it can lead to consistent performance improvement when the model is jointly trained on multiple domains. And we also achieve new state-of-the-art performance on multiple ReID benchmarks.

II. RELATED WORK

A. Domain Adaptation

Domain gap often exists between multiple datasets, and many researchers study on the domain adaptation problem. [6] propose unsupervised domain adaptation of deep feed-forward architectures, which allows large-scale training based on a large amount of annotated data in the source domain and a large amount of unannotated data in the target domain. Similar to many previous shallow and deep domain adaptive techniques, the adaptation is achieved through aligning the distributions of features across the two domains. Such ideas have appeared in other works [7], [8]. Recent GAN methods [9], [10], [11] use an adversarial approach to learn a transformation in the pixel space from one domain to another. The methods seek to find a domain-invariant feature space using the maximum mean discrepancy for this purpose is also achieved superior results to a certain extent. Dynamic transfer

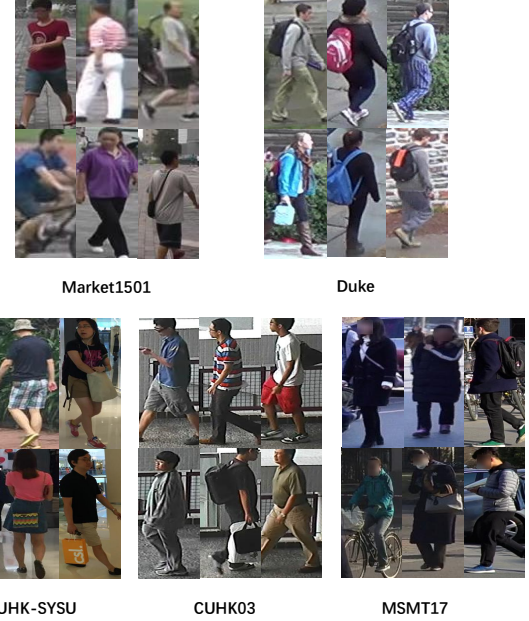


Fig. 2. Illustration of the domain gap between the person Re-Identification datasets. We can find that the people in Market1501 wear cool clothes, people in Duke often carry schoolbags, people in CUHK-SYSU wear cool clothes and often go to shopping malls, people in CUHK03 seem to often go in and out of buildings, and people in MSMT17 wear warm clothes.

for multi-source domain adaptation [5] across multiple domains and unifies multiple source domains into a single source domain, which simplifies the alignment between source and target domains. [12] proposed a multi-source framework from two perspectives, i.e. domain-specific view and domain-fusion view, for the unsupervised domain adaptive person ReID. The above domain adaptation methods focus on how to improve the performance of the target domain, and the target domain is different from the source domain. But the target domain studied in this paper is one of source domains.

B. Dynamic Convolution

The convolution operator is the core of convolutional neural networks (CNNs) and occupies the most computation cost. In order to address the issue that light-weight convolutional neural networks (CNNs) suffer performance degradation as their low computational budgets constrain both the depth (number of convolution layers) and the width (number of channels) of CNNs, dynamic convolution, a new design method that







	Dataset 0			Dataset 1	Dataset 2	
						
Domain ID	0			1	2	
Camera ID	0	1	2	0	0	1
Global Camera ID	0	1	2	3	4	5

Fig. 3. Example of domainID, camera ID and global camera ID. Each dataset is treated as a domain, so each dataset has a unique domain ID. Since camera ID already existed in the datasets, there may be the same camera ID among datasets, so we use the global camera ID in our experiments by default. Best viewed in color.

increases model complexity without increasing the network depth or width.

Instead of using a single convolution kernel per layer, [13] aggregates multiple parallel convolution kernels dynamically based upon their attention, which are input dependent. Assembling multiple kernels is not only computationally efficient due to the small kernel size, but also has more representation power since these kernels are aggregated in a non-linear way via attention. [14] propose a novel dynamic convolution method to adaptively generate convolution kernels based on image content, which reduces the redundant computation cost existed in conventional convolution kernels. [15] propose an effective and efficient operator for visual representation learning, reversing the design principles of convolution and generalizing the formulation of self-attention, which are able to disclose the underlying relationship between self-attention and convolution. [16] propose a lightweight content-adaptive filtering technique called DDF, which the main subject is to predict decoupled spatial and channel dynamic filters. This method can seamlessly replace standard convolution layers, consistently improving the performance of ResNets while also reducing model parameters and computational costs.

Most of these methods use dynamic convolution to get a larger model capacity. These methods generate different parameters according to different input features. And domain, camera, and sample are the three factors that cause large domain shifts in person ReID. The proposed DCSD method uses three information to generate model parameters. It reduces the domain conflict between multiple datasets and improves the performance of multi-domain joint training.

III. APPROACH

In person Re-ID, there is a common problem of domain gap between different datasets, and even sub-domain gap exist between different cameras in the same dataset. Therefore, there are a lot of works on domain transfer for person Re-ID, that is, training in one domain and testing in another domain. However, few researchers have studied the domain conflict problem when training in multiple domains and testing in one of them.

From our experiments, we found that the performance of multiple Re-ID datasets in joint training is even worse than

that of trained on single dataset. Further more, datasets with many cameras may have sub-domain conflicts problems. A static model is difficult to handle multiple datasets with large domain variances. Therefore, we propose a Domain-Camera-Sample Dynamic convolution module, which can generate a sample specific network according to the domain and camera information of each sample, so as to reduce the learning burden of the network and improve performance.

As shown in Figure 4, the DCSD is a $K \times K$ depthwise dynamic convolution, and its parameters are generated by its domain, camera and sample information. The domain information and camera information are predicted by the input feature with two MLPs, and then the input feature, domain information and camera information are added together to generating the convolution parameters. The input feature (sample information) is from the sample, which is an internal factor, while the domain information and camera information are external factors. In the process of generating convolution parameters, we use two branches, spatial branch and channel branch, to reduce the memory consumption. The spatial branch generates the features ($K \times K \times H \times W$) through a 1×1 convolution, and the channel branch generates the features ($K \times K \times C$) through two layers of fully connection. During the convolution operation, the corresponding ($K \times K$) kernel is obtained from the outputs from the two branches according to the spatial position and channel position, and they are multiplied to obtain the final ($K \times K$) kernel for the current position. We can use DCSD module to replace the standard convolution in the classical model to realize the DCSD model. In this paper, we use DCSD ($K = 3$) to replace all (3×3) convolution layers in ResNet50, which is called ResNet50-DCSD.

A. DCSD Module

To better understand our approach, we give a brief review of the standard convolution. The input feature can be written as $X \in \mathbb{R}^{H \times W \times C}$, where H , W and C are height, width and number of channels of the input X respectively. The convolution kernel of the standard convolution is $W \in \mathbb{R}^{K \times K \times C \times C'}$, where K is the kernel size. If $pad = K - 1$ and $stride = 1$, then the output feature is $output \in \mathbb{R}^{H \times W \times C'}$, C' is the output channels. And the convolution operation (ignore the bias) can be written as

$$\text{Output}_{k,l,c'} = \sum_{i,j,c} W_{i,j,c,c'} \cdot X_{k+i-1,l+j-1,c} \quad (1)$$

In dynamic convolution, the model needs to generate convolution parameters before the convolution operation. If the convolution kernel parameters are generated directly for standard convolution, the model size and computational cost will be extremely vast. In order to decrease the model size and computation cost, some methods [15], [16] use depthwise dynamic convolution instead of standard convolution. We also use depthwise dynamic convolution in DCSD, but there are some subtle details in DCSD are distinct from the normal methodologies. The parameters of our depthwise version of

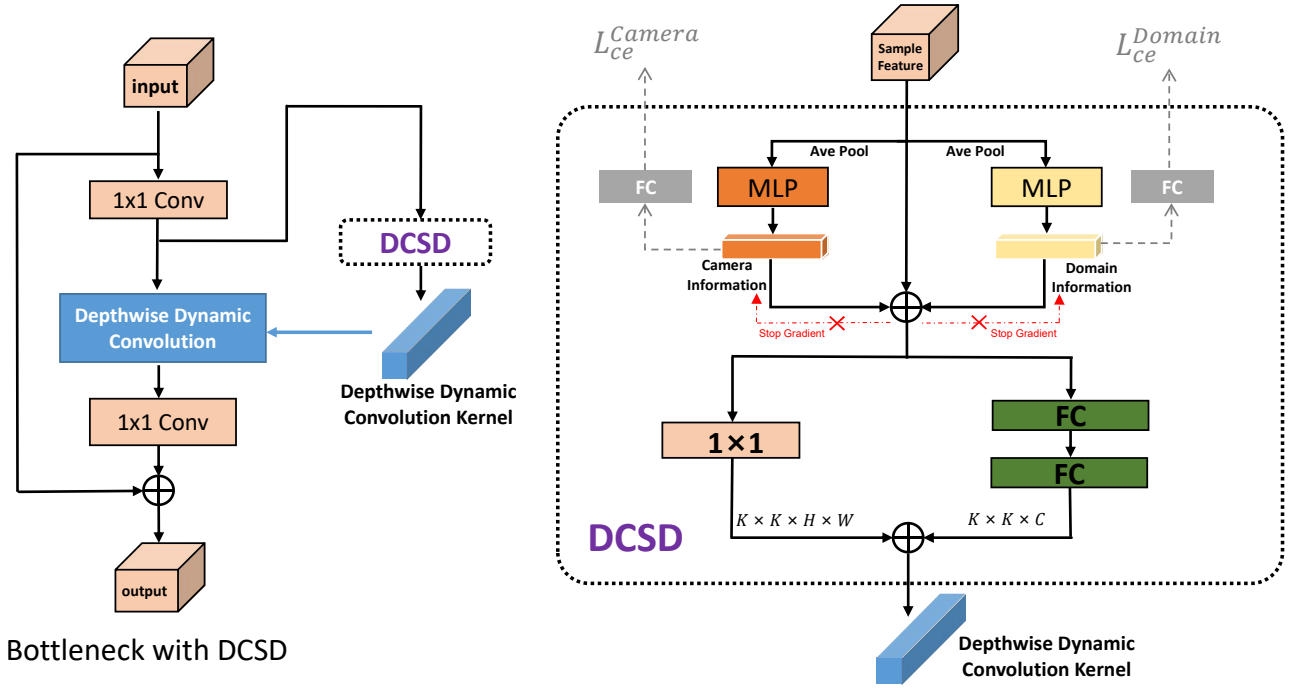


Fig. 4. The architecture of DCSD module. The DCSD is a $K \times K$ depthwise dynamic convolution, and its parameters are generated by its domain, camera and sample information. The domain information and camera information are predicted by the input feature with two MLPs, and then the input feature, domain information and camera information are added together to generating the convolution parameters. The MLP consists of two fully connection layers, and hidden dimension is 1/4 of the input dimension. In the process of generating convolution parameters, we use two branches to reduce memory consumption. Best viewed in color.

dynamic convolution are jointly generated by domain, camera and sample information. The DCSD module can be written as

$$\begin{aligned} \hat{\mathbf{W}} &= \text{DCSD}(\mathbf{X}, \text{Domain.detach}(), \text{Camera.detach}()), \\ \text{Output}_{k,l,c} &= \sum_{i,j} \hat{\mathbf{W}}_{i,j,c} \cdot \mathbf{X}_{k+i-1,l+j-1,c}, \end{aligned} \quad (2)$$

where X is the feature extracted from the sample, the information of *Domain* and *Camera* is predicted by X through two MLPs, and the domain ID and camera ID are used for supervision training by cross entropy losses. We cut off the gradient propagation for *Domain* and *Camera* for making *Domain* and *Camera* only supervised by domain ID and camera ID. *detach()* means that it detaches the output from the computational graph, so no gradient will be back propagated along the variable. As Figure 4 shows, DCSD generates convolution parameters through domain, camera and sample information, and then the convolution parameters are applied to the input feature.

The models in our experiments are trained by jointly minimizing the triplet loss and the cross-entropy losses. The cross-entropy losses include sample ID classification loss, domain ID classification losses and camera ID classification losses. Specifically, for N vehicle image samples selected from M IDs, the triplet loss can be formulated as

$$L_{tri} = \frac{1}{N} \sum_{i=1}^N [D(\mathbf{x}_i, \mathbf{x}_i^p) - D(\mathbf{x}_i, \mathbf{x}_i^n) + \alpha]_+, \quad (3)$$

where \mathbf{x}_i , \mathbf{x}_i^p , and \mathbf{x}_i^n denote the anchor, positive and negative samples respectively. In practice, we often choose the farthest

positive sample and the closest negative sample in the batch to form a hard triplet. $D(\cdot)$ is a distance metric and α is a pre-defined margin scalar (e.g., 0.3). The $[\cdot]_+$ denotes $\max([\cdot], 0)$.

The cross-entropy loss can be defined as

$$L_{ce} = -\frac{1}{N} \sum_{i=1}^N \sum_{j=1}^M [y_i = j] \cdot \log(\text{Prob}_{i,j}), \quad (4)$$

where $[\cdot]$ denotes the indicator function and y_i is the ground truth ID for the i -th sample. $\text{Prob}_{i,j}$ is the predicted probability for i -th sample belonging to the ID j . Note that, in our experiment, there are three different types of IDs: domain ID, camera ID and sample ID. Then the total loss is:

$$L_{total} = L_{tri} + L_{ce}^{Sample} + \frac{1}{B} \sum_{i=1}^B L_{ce}^{Domain} + \frac{1}{B} \sum_{i=1}^B L_{ce}^{Camera} \quad (5)$$

where B is the number of Bottleneck in the model.

B. Global Camera ID

Our DCSD requires domain ID and camera ID as input information. We number datasets and use the index as their domain IDs, so each dataset has a unique domain ID. Because camera ID already existed in the datasets, there may be the same camera ID among datasets, so we need to use the global camera ID. In other words, we uniformly number all cameras among multiple datasets. The numbering method is shown in figure 3.

TABLE II

PERFORMANCE (%) COMPARISONS WITH THE STATE-OF-THE-ARTS ON CUHK03, MARKET1501, DUKE AND MSMT17. “†” MEANS BY OUR IMPLEMENTATION. **BOLD** AND *Italic* FONTS REPRESENT THE BEST AND SECOND BEST RESPECTIVELY. THE “JOINT TRAINING GAIN” IS THE PERFORMANCE GAP BETWEEN “BoT/AGW (JOINT TRAINING)” AND “BoT/AGW + DCSD (JOINT TRAINING)”.

Method	CUHK03(L)		CUHK-SYSU		Market1501		Duke		MSMT17	
	Rank-1	mAP	Rank-1	mAP	Rank-1	mAP	Rank-1	mAP	Rank-1	mAP
IDE [17]	43.8	38.9	-	-	85.3	68.5	73.2	52.8	-	-
Gp-reid [18]	-	-	-	-	92.2	81.2	85.2	72.8	-	-
MGCAM [19]	50.1	50.2	-	-	83.8	74.3	-	-	-	-
MaskReID [20]	-	-	-	-	90.0	70.3	78.9	61.9	-	-
AACN [21]	-	-	-	-	85.9	66.9	76.8	59.3	-	-
SPReID [22]	-	-	-	-	92.5	81.3	84.4	71.0	-	-
HA-CNN [23]	44.4	41.0	-	-	91.2	75.7	80.5	63.8	-	-
DuATM [24]	-	-	-	-	91.4	76.6	81.8	64.6	-	-
Mancs [25]	69.0	63.9	-	-	93.1	82.3	84.9	71.8	-	-
MGN [26]	68.0	67.4	-	-	95.7	86.9	88.7	78.4	-	-
HPM [27]	63.9	57.5	-	-	94.2	82.7	86.6	74.3	-	-
DSA-reID [28]	78.9	75.2	-	-	95.7	87.6	86.2	74.3	-	-
OSNet [29]	72.3	67.8	-	-	94.8	84.9	88.6	73.5	78.7	52.9
IANet [30]	72.4	-	-	-	94.4	83.1	87.1	73.4	75.5	46.8
HOReid [31]	-	-	-	-	94.2	84.9	86.9	75.6	-	-
ISP [32]	76.5	74.1	-	-	95.3	88.6	89.6	80.0	-	-
TransReID (DeiT-S/16) [33]	-	-	-	-	-	-	-	-	76.3	55.2
BoT [†] [34]	69.1	67.3	87.2	86.0	94.2	86.1	86.4	76.8	74.1	50.2
BoT (Joint Training)	67.5	65.7	90.5	89.3	93.7	84.6	86.0	75.9	72.9	49.5
BoT-DCSD	66.8	66.5	87.2	85.6	95.3	87.8	88.6	79.0	79.8	58.2
BoT-DCSD (Joint Training)	79.4	78.0	92.4	91.3	95.9	89.5	89.6	80.6	80.3	61.5
Joint Training Gain	+11.9	+12.3	+1.9	+2.0	+2.2	+4.9	+3.6	+4.7	+7.4	+12.0
AGW [†] [35]	75.1	73.4	90.1	88.7	95.4	88.5	89.2	79.4	78.0	54.3
AGW (Joint Training)	70.3	69.1	91.8	90.8	94.0	86.2	88.0	78.5	75.5	52.6
AGW-DCSD	77.2	74.3	84.9	83.6	95.3	88.1	89.6	80.3	80.1	60.7
AGW-DCSD (Joint Training)	80.9	78.8	92.7	91.4	95.7	89.3	90.0	81.2	81.8	62.9
Joint Training Gain	+10.6	+9.7	+0.9	+0.6	+1.7	+3.1	+2.0	+2.7	+6.3	+10.3

IV. EXPERIMENTS

A. Datasets

We conduct extensive experiments on five public person ReID datasets, *i.e.*, Market1501, CUHK-SYSU, DukeMTMC, CUHK03 and MSMT17. The detailed statistics of person Re-Identification datasets are shown in Table I. For the joint training of multiple datasets, we mix the five datasets together and take random samples for training. The Cumulative Match Curve (CMC) at Rank-1 and the mean Average Precision (mAP) are used as the evaluation criteria.

Market1501 [38] contains 32,668 images of 1,501 labeled persons of six camera views. It is collected from Tsinghua University and the pictures are from the outdoors. There are 751 identities in the training set and 750 identities in the testing set. View overlapping exists among different cameras, including 5 high-resolution cameras, and a low-resolution camera.

CUHK-SYSU [39] were originally used for person search which is collected from street snap and movies. And we modified the original dataset by using the ground-truth person bounding box annotation. For testing on this dataset, we fixed both query and gallery sets instead of using variable gallery sets. We used 2,900 query persons, with each query containing at least one image in the gallery. Note that CUHK-SYSU lacks

camera ID information, so the cross-camera setting is ignored during the evaluation phase.

Duke [40] is a subset of the DukeMTMC used for person re-identification by images. It is collected from Duke University and the pictures are from the outdoors. It consists of 36,411 images of 1,812 persons from 8 high-resolution cameras. 16,522 images of 702 persons are randomly selected from the dataset as the training set, and the remaining 702 persons are divided into the testing set where contains 2,228 query images and 17,661 gallery images.

CUHK03 [41] is collected from The Chinese University of Hong Kong and the pictures are from the indoors and outdoors. It has two ways of annotating bounding box including labelled by humans or detected from deformable part models (DPMs). The labeled dataset includes 7,368 training, 1,400 query and 5,328 gallery images while detected dataset consists of 7,365 training, 1,400 query and 5,332 gallery images. We use the labeled dataset as default.

MSMT17 [42] is the current largest publicly available person Re-ID dataset. It has 126,441 images of 4,101 identities from indoors and outdoors. The video is collected with different weather conditions at three-time slots (morning, noon, afternoon). All annotations, including camera IDs, weathers, and time slots, are available.

TABLE III

THE MULTI-DOMAIN JOINT TRAINING PERFORMANCE (%) COMPARISONS WITH THE STATE-OF-THE-ARTS DOMAIN GENERALIZATION METHODS ON CUHK03, MARKET1501, DUKE AND MSMT17. EACH ROW IS TRAINED ON THE SAME SOURCE DOMAIN, AND THEN TEST THE PERFORMANCE ON DIFFERENT TARGET DOMAINS. **BOLD** FONTS REPRESENT THE BEST, “*” MEANS BY OUR IMPLEMENTATION.

Method	Source	CUHK03(L)		CUHK-SYSU		Market1501		Duke		MSMT17	
		Rank-1	mAP	Rank-1	mAP	Rank-1	mAP	Rank-1	mAP	Rank-1	mAP
BoT (Baseline) [34]	CUHK03(L)	69.1	67.3	70.2	67.3	46.3	22.3	19.9	10.5	7.1	2.1
	CUHK-SYSU	3.9	4.4	87.2	86.0	46.2	23.5	24.5	14.5	12.8	4.8
	Market1501	4.0	4.4	74.4	70.8	94.2	86.1	27.6	14.8	7.7	2.4
	Duke	4.8	5.2	66.1	62.0	48.0	22.2	86.4	76.8	12.8	3.9
	MSMT17	9.7	10.3	79.5	76.7	56.4	28.7	49.6	33.5	74.1	50.2
	Joint Training	67.5	65.7	90.5	89.3	93.7	84.6	86.0	75.9	72.9	49.5
BoT-DN* [36]	CUHK03(L)	59.6	57.9	68.6	65.2	58.1	30.6	43.1	24.6	18.3	5.7
	CUHK-SYSU	5.1	6.5	79.9	77.5	50.4	27.6	34.9	19.9	14.8	4.8
	Market1501	14.3	14.6	73.9	71.2	92.6	81.8	48.9	29.5	19.1	6.5
	Duke	8.8	8.8	63.6	60.7	57.1	28.2	85.8	71.4	21.7	6.9
	MSMT17	14.7	15.1	76.0	72.8	59.7	31.6	60.8	41.4	74.5	48.4
	Joint Training	66.8	64.5	89.6	88.3	93.1	81.8	84.5	72.7	74.6	49.2
BoT-SNR* [37]	CUHK03(L)	71.9	70.6	77.7	74.5	65.4	38.9	41.2	25.6	22.1	7.5
	CUHK-SYSU	6.9	8.4	88.4	87.3	61.6	37.6	39.6	25.2	20.4	8.2
	Market1501	17.6	17.5	79.7	77.2	95.0	86.4	49.7	32.4	21.1	7.6
	Duke	11.0	10.9	71.3	68.1	62.1	33.0	89.1	77.5	26.7	9.1
	MSMT17	16.9	17.7	81.3	78.5	64.4	37.4	63.4	45.6	77.3	52.3
	Joint Training	66.6	65.8	90.7	89.5	94.2	84.9	87.5	76.4	76.2	51.3
BoT-DCSD (Ours)	CUHK03(L)	66.8	66.5	75.5	72.3	49.1	25.4	24.2	13.3	10.3	3.3
	CUHK-SYSU	4.0	4.9	87.2	85.6	46.1	24.1	27.1	15.1	13.2	5.0
	Market1501	7.6	8.0	76.7	87.8	95.3	87.8	32.0	18.9	10.2	3.4
	Duke	5.4	5.6	66.9	63.5	48.6	22.3	88.6	79.0	14.1	4.5
	MSMT17	12.5	12.3	78.3	75.0	52.4	27.1	49.9	33.1	79.8	59.5
	Joint Training	79.4	78.0	92.4	91.3	95.9	89.5	89.6	80.6	80.3	61.5

TABLE IV

THE PARAMS AND FLOPS COMPARISON WITH RESNET50.

Model	Params	FLOPs
ResNet50	25.6M	4.1B
ResNet50-DCSD	18.2M	2.3B

B. Implementation Details

The proposed model is implemented using PyTorch in Ubuntu16.04. All experiments are conducted on an NVIDIA GTX 1080Ti GPU with 11GB memory. The images were resized to 256×128 and the batch size is 64 (4 images/ID and 16 IDs). We didn’t utilize the test with flip and re-ranking. Unless specified otherwise, we use BoT [34] as the baseline by default. The backbone is pre-trained on ImageNet. The margin in the triplet loss is 0.3 in all our experiments. The model has trained 120 epochs and the learning rate is initialized to 3.5×10^{-4} and divided by 10 at the 40th epoch and 90th epoch. The detailed implementation of other methods follows the settings in their respective papers.

C. Experimental Results

In this section, we evaluate our proposed DCSD in comparison with state-of-the-art approaches on several benchmarks. For a fair comparison, we only demonstrate the performance of the TransReID [33] model trained with DeiT-S/16 which is comparable with ResNet50.

It can be seen from Table II that after the joint training of multiple datasets (domains), the ReID performance of BoT and AGW decreased significantly on most of the datasets. Although the data involved in training increased, the performance decreased due to the existence of multi-domain conflicts. For example, the Rank-1 of AGW on CUHK03 drop from 75.1% to 70.3%.

When we replace the ResNet50 in BoT and AGW with its corresponding DCSD model, the performance has increased on some datasets, such as MSMT17 and Duke. And the performance dropped on CUHK-SYSU. The model size and FLOPs are listed in Table IV. The reason for this phenomenon may be that MSMT17 and Duke have many cameras (15 and 8), so there may be some sub-domain conflict inside MSMT17 and Duke, so “BoT-DCSD” and “AGW-DCSD” can get good performance on MSMT17 and Duke. However, CUHK-SYSU only has one camera, Therefore, the performance with DCSD on CUHK-SYSU is not so well. For DCSD joint training, consistent performance improvements were achieved for all datasets. Note that the mAP of “BoT-DCSD” on CUHK03 and CUHK-SYSU are lower than that of “BoT” (-0.8% and -0.4%), but the mAP of “BoT-DCSD (joint training)” are higher than that of “BoT (joint training)” ($+12.3\%$ and $+2.0\%$). Compared with the original method of joint training, “Joint Training Gain” can get up to 12.3% performance (mAP) improvement. Compared with other approaches, the performance of our method achieves state-of-

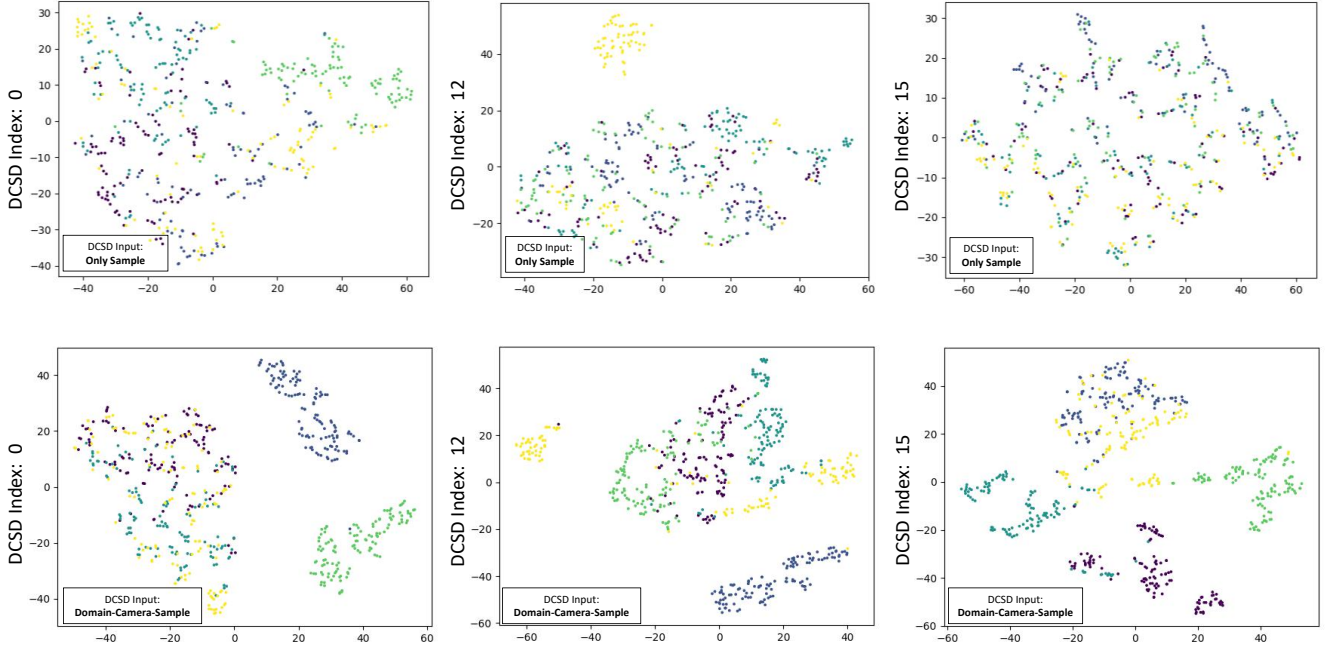


Fig. 5. Visualization of the generated parameter in DCSD. There are 16 DCSD modules in our ResNet50-DCSD, and we visualize the generated parameters of 0th, 12th, and 15th DCSD. The first column shows the visualization of convolution parameters generated only by sample features, and the second column shows the visualization of convolution parameters generated by comprehensively considering domain, camera, and sample information. The same color represents the same domain, best viewed in color.

the-art performance on all of the datasets.

D. Comparison with Domain Generalization Methods

Domain generalization [36], [37], [43] has attracted much attention in person ReID recently. It aims to make a model trained on multiple source domains generalize to an unseen target domain which can alleviate the domain discrepancy between datasets. In this subsection, we experiment whether the traditional domain generalization methods can solve the problem of domain conflict in multi-domain joint training for person ReID. We use BoT [34] as the baseline, and then add DN [36], SNR [37] or DCSD (ours) to the BoT. Table III shows that all three methods show better domain generalization ability than the baseline. However, when multiple domains are jointly trained, the performance of BoT-DN and BoT-SNR are even worse than that of each domain independently trained. Only BoT-DCSD (ours) can achieve consistent performance improvement on all of the benchmarks. One possible reason is that learning a domain agnostic model is difficult, since different domains can give rise to very different image distributions [5]. When forcing a model to be domain agnostic, it essentially averages the domain conflict rather than avoids it. Therefore, the domain generalization methods may not work well in multi-domain joint training.

V. ABLATION STUDY

In this section, we conduct investigations on how several key factors affect the model’s overall performance.

A. Domain-Related Factors

We divide the domain-related factors into internal domain-related factor and external domain-related factors. Specifically, we regard the domain information and camera information as the external domain-related factors while sample information as the internal domain-related factor. And in this subsection, we study the impact of these domain-related factors on performance in multi-domain joint training. In Table V, we can see that the performance of multi-domain joint training is low when DCSD is not used. With the gradual increase of using sample information, domain information and camera information to dynamically generate convolution parameters, its performance began to improve gradually. Finally, the mAP improved by 2.0% to 12.3% on different datasets. The experimental results show that both internal and external domain-related factors are helpful to generate dynamic convolution parameters, so as to reduce domain conflict.

TABLE V
THE PERFORMANCE (MAP) WITH THE DOMAIN-RELATED FACTORS TO GENERATE THE SAMPLE-SPECIFIC NETWORK IN DCSD. “D”, “C”, “S” AND “JT” ARE SHORT FOR “DOMAIN INFORMATION”, “CAMERA INFORMATION”, “SAMPLE INFORMATION” AND “JOINT TRAINING” RESPECTIVELY. .

Method	CUHK03(L)	CUHK-SYSU	Market1501	Duke	MSMT17
BoT (JT)	65.7	89.3	84.6	75.9	49.5
BoT-DCSD (JT), S	74.6	90.9	88.2	80.2	58.8
BoT-DCSD (JT), S + D	75.2	91.0	88.5	80.4	59.6
BoT-DCSD (JT), S + C	77.9	91.1	89.1	80.4	61.2
BoT-DCSD (JT), S + D + C	78.0	91.3	89.5	80.6	61.5

B. Stop Gradient

It can be seen from Figure 4 that the camera information and domain information predicted by MLPs will be used in two branches: the first branch is to predict the corresponding IDs and the second branch is to generate convolution parameters. In order to make the camera information and domain information more pure, we cut off the gradient propagation on the second branch that the camera information and domain information are only supervised by the corresponding IDs. We named it Stop Gradient (SG). In Table VI we can see that compared with “DCSD without GS”, “DCSD with GS” achieve consistent performance improvement on all datasets.

TABLE VI
DCSD WITH OR WITHOUT THE STOP GRADIENT (SG). “JT” IS SHORT FOR “JOINT TRAINING”.

Method	CUHK03(L)	CUHK-SYSU	Market1501	Duke	MSMT17
BoT-DCSD (JT) w/o GS	76.0	90.5	88.9	80.4	61.0
BoT-DCSD (JT) with GS	78.0	91.3	89.5	80.6	61.5
AGW-DCSD (JT) w/o GS	76.3	90.9	88.9	80.9	61.4
AGW-DCSD (JT) with GS	78.8	91.4	89.3	81.2	62.9

C. Visualization of Generated Parameter

We visualize the parameters generated by samples from different domains by t-SNE [44]. T-SNE visualizes high-dimensional data by giving each datapoint a location in a two or three-dimensional map. It is a variation of stochastic neighbor embedding that is much easier to optimize, and produces significantly better visualizations by reducing the tendency to crowd points together in the center of the map. In Figure 5, we can see that if only sample information is used to generate convolution parameters, the parameters generated by samples from different domains are not very different. But the parameters generated by the combination of domain, camera and sample information have good domain discrimination. It means that the network can dynamically generate a sample specific network which is suitable for the current domain.

VI. CONCLUSION

In this work, we deliver some surprising findings that the traditional static network is tough to deal with the conflict between multiple domains for person re-identification, which may lead to arousing the performance degradation when multi-domain joint training. And then we propose a Domain-Camera-Sample Dynamic network (DCSD) whose parameters can be adaptive to various factors. Without the need to dealing with the complex relationship between different domains, DCSD uses internal domain-related factor, such as features from the sample information, and external domain-related factors, such as domain information and camera information, to dynamically generate the sample-specific network. Experimental results show that DCSD can boost the performance (up to 12.3%) while joint training in multiple domains.

REFERENCES

- [1] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “Imagenet classification with deep convolutional neural networks,” *Advances in neural information processing systems*, vol. 25, pp. 1097–1105, 2012.
- [2] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.
- [3] C. Sun, A. Shrivastava, S. Singh, and A. Gupta, “Revisiting unreasonable effectiveness of data in deep learning era,” in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 843–852.
- [4] L. Qi, L. Wang, J. Huo, L. Zhou, Y. Shi, and Y. Gao, “A novel unsupervised camera-aware domain adaptation framework for person re-identification,” in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, October 2019.
- [5] Y. Li, L. Yuan, Y. Chen, P. Wang, and N. Vasconcelos, “Dynamic transfer for multi-source domain adaptation,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 10998–11007.
- [6] Y. Ganin and V. Lempitsky, “Unsupervised domain adaptation by back-propagation,” in *International conference on machine learning*. PMLR, 2015, pp. 1180–1189.
- [7] B. Sun, J. Feng, and K. Saenko, “Return of frustratingly easy domain adaptation,” 2016.
- [8] B. Sun and K. Saenko, “Deep coral: Correlation alignment for deep domain adaptation,” in *European conference on computer vision*. Springer, 2016, pp. 443–450.
- [9] K. Bousmalis, N. Silberman, D. Dohan, D. Erhan, and D. Krishnan, “Unsupervised pixel-level domain adaptation with generative adversarial networks,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 3722–3731.
- [10] M.-Y. Liu and O. Tuzel, “Coupled generative adversarial networks,” *Advances in neural information processing systems*, vol. 29, pp. 469–477, 2016.
- [11] J. Hoffman, E. Tzeng, T. Park, J.-Y. Zhu, P. Isola, K. Saenko, A. Efros, and T. Darrell, “Cycada: Cycle-consistent adversarial domain adaptation,” in *International conference on machine learning*. PMLR, 2018, pp. 1989–1998.
- [12] Z. Bai, Z. Wang, J. Wang, D. Hu, and E. Ding, “Unsupervised multi-source domain adaptation for person re-identification,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2021, pp. 12914–12923.
- [13] Y. Chen, X. Dai, M. Liu, D. Chen, and Z. Liu, “Dynamic convolution: Attention over convolution kernels,” in *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.
- [14] Y. Zhang, J. Zhang, Q. Wang, and Z. Zhong, “Dynet: Dynamic convolution for accelerating convolutional neural networks,” 2020.
- [15] D. Li, J. Hu, C. Wang, X. Li, Q. She, L. Zhu, T. Zhang, and Q. Chen, “Involution: Inverting the inheritance of convolution for visual recognition,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2021, pp. 12321–12330.
- [16] J. Zhou, V. Jampani, Z. Pi, Q. Liu, and M.-H. Yang, “Decoupled dynamic filter networks,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2021, pp. 6647–6656.
- [17] Y. Sun, L. Zheng, Y. Yang, Q. Tian, and S. Wang, “Beyond part models: Person retrieval with refined part pooling (and a strong convolutional baseline),” in *ECCV*, 2018, pp. 480–496.
- [18] J. Almazan, B. Gajic, N. Murray, and D. Larlus, “Re-id done right: towards good practices for person re-identification,” *arXiv:1801.05339*, 2018.
- [19] C. Song, Y. Huang, W. Ouyang, and L. Wang, “Mask-guided contrastive attention model for person re-identification,” in *CVPR*, 2018, pp. 1179–1188.
- [20] L. Qi, J. Huo, L. Wang, Y. Shi, and Y. Gao, “Maskreid: A mask based deep ranking neural network for person re-identification,” *arXiv:1804.03864*, 2018.
- [21] J. Xu, R. Zhao, F. Zhu, H. Wang, and W. Ouyang, “Attention-aware compositional network for person re-identification,” in *CVPR*, 2018, pp. 2119–2128.
- [22] M. M. Kalayeh, E. Basaran, M. Gökmen, M. E. Kamasak, and M. Shah, “Human semantic parsing for person re-identification,” in *CVPR*, 2018, pp. 1062–1071.
- [23] W. Li, X. Zhu, and S. Gong, “Harmonious attention network for person re-identification,” in *CVPR*, 2018, pp. 2285–2294.
- [24] J. Si, H. Zhang, C.-G. Li, J. Kuen, X. Kong, A. C. Kot, and G. Wang, “Dual attention matching network for context-aware feature sequence based person re-identification,” in *CVPR*, 2018, pp. 5363–5372.
- [25] C. Wang, Q. Zhang, C. Huang, W. Liu, and X. Wang, “Manacs: A multi-task attentional network with curriculum sampling for person re-identification,” in *ECCV*, 2018, pp. 365–381.

- [26] G. Wang, Y. Yuan, X. Chen, J. Li, and X. Zhou, "Learning discriminative features with multiple granularities for person re-identification," in *ACM Multimedia*, 2018, pp. 274–282.
- [27] Y. Fu, Y. Wei, Y. Zhou, H. Shi, G. Huang, X. Wang, Z. Yao, and T. Huang, "Horizontal pyramid matching for person re-identification," in *AAAI*, vol. 33, 2019, pp. 8295–8302.
- [28] Z. Zhang, C. Lan, W. Zeng, and Z. Chen, "Densely semantically aligned person re-identification," in *CVPR*, 2019, pp. 667–676.
- [29] K. Zhou, Y. Yang, A. Cavallaro, and T. Xiang, "Omni-scale feature learning for person re-identification," 2019.
- [30] R. Hou, B. Ma, H. Chang, X. Gu, S. Shan, and X. Chen, "Interaction-and-aggregation network for person re-identification," in *CVPR*, 2020.
- [31] G. Wang, S. Yang, H. Liu, Z. Wang, Y. Yang, S. Wang, G. Yu, E. Zhou, and J. Sun, "High-order information matters: Learning relation and topology for occluded person re-identification," in *CVPR*, 2020, pp. 6449–6458.
- [32] K. Zhu, H. Guo, Z. Liu, M. Tang, and J. Wang, "Identity-guided human semantic parsing for person re-identification," *arXiv:2007.13467*, 2020.
- [33] S. He, H. Luo, P. Wang, F. Wang, H. Li, and W. Jiang, "Transreid: Transformer-based object re-identification," 2021.
- [34] H. Luo, W. Jiang, Y. Gu, F. Liu, X. Liao, S. Lai, and J. Gu, "A strong baseline and batch normalization neck for deep person re-identification," *TMM*, 2019.
- [35] M. Ye, J. Shen, G. Lin, T. Xiang, and S. Hoi, "Deep learning for person re-identification: A survey and outlook," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. PP, no. 99, pp. 1–1, 2021.
- [36] J. Jia, Q. Ruan, and T. Hospedales, "Frustratingly easy person re-identification: Generalizing person re-id in practice," in *Proceedings of the British Machine Vision Conference (BMVC)*, K. Sidorov and Y. Hicks, Eds. BMVA Press, September 2019, pp. 141.1–141.14. [Online]. Available: <https://dx.doi.org/10.5244/C.33.141>
- [37] X. Jin, C. Lan, W. Zeng, Z. Chen, and L. Zhang, "Style normalization and restitution for generalizable person re-identification," in *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.
- [38] L. Zheng, L. Shen, L. Tian, S. Wang, J. Wang, and Q. Tian, "Scalable person re-identification: A benchmark," in *ICCV*, 2015, pp. 1116–1124.
- [39] X. Tong, L. Shuang, B. Wang, L. Liang, and X. Wang, "Joint detection and identification feature learning for person search," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.
- [40] Z. Zheng, L. Zheng, and Y. Yang, "Unlabeled samples generated by gan improve the person re-identification baseline in vitro," in *ICCV*, 2017, pp. 3754–3762.
- [41] W. Li, R. Zhao, T. Xiao, and X. Wang, "Deepreid: Deep filter pairing neural network for person re-identification," in *CVPR*, 2014, pp. 152–159.
- [42] L. Wei, S. Zhang, W. Gao, and Q. Tian, "Person transfer gan to bridge domain gap for person re-identification," in *CVPR*, 2018, pp. 79–88.
- [43] S. Choi, T. Kim, M. Jeong, H. Park, and C. Kim, "Meta batch-instance normalization for generalizable person re-identification," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2021, pp. 3425–3435.
- [44] L. v. d. Maaten and G. Hinton, "Visualizing data using t-sne," *Journal of machine learning research*, vol. 9, no. Nov, pp. 2579–2605, 2008.