# DEX: Domain Embedding Expansion for Generalized Person Re-identification

Eugene P.W. Ang[1,2]
phuaywee001@e.ntu.edu.sg

Lin Shan[1]
shan.lin@ntu.edu.sg

Alex C. Kot[1]
eackot@ntu.edu.sg

[1] Rapid-Rich Object Search (ROSE) Lab
Nanyang Technological University
Singapore

[2] Defence Science and Technology
Agency
Singapore

## Abstract

In recent years, supervised Person Re-identification (Person ReID) approaches have demonstrated excellent performance. However, when these methods are applied to inputs from a different camera network, they typically suffer from significant performance degradation. Different from most domain adaptation (DA) approaches addressing this issue, we focus on developing a domain generalization (DG) Person ReID model that can be deployed without additional fine-tuning or adaptation. In this paper, we propose the Domain Embedding Expansion (DEX) module. DEX dynamically manipulates and augments deep features based on person and domain labels during training, significantly improving the generalization capability and robustness of Person ReID models to unseen domains. We also developed a light version of DEX (DEXLite), applying negative sampling techniques to scale to larger datasets and reduce memory usage for multi-branch networks. Our proposed DEX and DEXLite can be combined with many existing methods, Bag-of-Tricks (BagTricks), the Multi-Granularity Network (MGN), and Part-Based Convolutional Baseline (PCB), in a plug-and-play manner. With DEX and DEXLite, existing methods can gain significant improvements when tested on other unseen datasets, thereby demonstrating the general applicability of our method. Our solution outperforms the state-of-the-art DG Person ReID methods in all large-scale benchmarks as well as in most the small-scale benchmarks.

# 1 Introduction

Person Re-identification (Person ReID) has achieved impressive performance on academic benchmarks in recent years. However, generalization issues prevent its transition into the applied world. For example, a model trained on one Person ReID domain using standard techniques can achieve high accuracy when independently tested within the same domain, but its performance degrades drastically when tested on a unseen domain. This reveals a lack of generalization ability in single-dataset supervised models and suggests that the features learned by these models over-fit the training domain instead of capturing the general features relevant for person discrimination. In recent years, much of Person ReID research has focused on unsupervised domain adaptation (DA) [4, 16, 20, 29], which uses unlabeled

data collected from the target domain to alleviate the domain over-fitting problem. However, in many real-world applications, access to the data from the target domain beforehand may not be a valid assumption to make. Therefore, domain generalization (DG) is a more practical strategy for that problem. DG methods leverage the different distributions of multiple datasets to reduce domain bias.

Current DG Person ReID methods involve complex frameworks such as meta-learning, hyper-networks and memory banks [15, 27, 34] supported by custom loss functions and normalizations. In this paper, we tackle the DG problem of Person ReID from a novel perspective using deep feature augmentation. We propose **D**omain **E**mbedding E**x**pansion (DEX), a deep feature augmentation module that leverages person and domain labels to fill the domain gap between deep features during training. Many Generative Adversarial Network (GAN) [5] based methods also transfer style information from one domain to another while preserving person identity features. However, such methods are computationally expensive and require a nontrivial GAN-training stage. Our proposed DEX implicitly projects extracted deep features across domain manifolds during the training process, as shown in Figure 1. Applying DEX on our baseline (DualNorm [11] + BagTricks [18] over a ResNet-50 [7] backbone) allows us to outperform state-of-the-art methods on all large-scale benchmarks by a wide margin. Integrating DEX on other popular Person ReID architectures such as Multi-Granularity Network (MGN) [28] and Part-Based Convolutional Baseline (PCB) [23], demonstrates consistent improvement in model generalization. When utilizing multiple Person ReID datasets for training, the increase in unique person-identities (PIDs) could lead to a huge demand for GPU memory. We also developed a memory-light version of DEX (DEXLite) that applies negative sampling to reduce computation, memory consumption, and training time. DEXLite supports datasets with large numbers of PIDs, larger batch sizes, and multi-branch model architectures.

To summarize, our contributions are as follows:

- We propose DEX, a deep feature data augmentation method tailored for the multi-domain generalization Person ReID problem that leverages domain labels to implicitly project deep features over domain manifolds.
- With DEX and Instance Normalization (IN), a simple ResNet50 backbone can outperform state-of-the-art performance on all large scale Person ReID DG benchmarks (Market-1501, DukeMTMC-reID, CUHK03 and MSMT17) by a wide margin.
- For memory-limited machines, we developed DEXLite, a memory-light version of DEX, that uses negative sampling to reduce memory use during training, enabling the use of this technique on a broader range of problems. We apply DEXLite on two multi-branch architectures, MGN and PCB, to demonstrate significant improvements.
- DEXLite also outperforms state-of-the-art methods in most of the small-dataset DG benchmark metrics and closely matches the top performers for the remaining metrics.
- Our proposed method is straightforward and does not require complex frameworks or specialized neural network architectures.

## 2 Related Work

**Person Re-identification.** Deep learning based Person ReID approaches developed in recent years, such as PCB [23], BagTricks [18], and MGN [28] have achieved impressive accuracy in Person ReID. These methods are usually trained and evaluated within the same dataset. However, due to the difficulty of data collection and limited numbers of cameras and

pedestrians, existing datasets suffer from limited variability in location, weather, pedestrian clothing, illumination, and camera color settings. Such limitations induce a strong domain bias during training and significantly degrade model performance when testing on other domains. Hence, many domain adaptation (DA) methods such as UMDL [20], SPGAN [4], TJ-AIDL [29], MMFA [16] were proposed to bridge the gap between domains for the Person ReID problem. DA methods utilize unlabeled data from the target domain to generate pseudo-labels or transfer the target domain image style to the source domain. Although many DA approaches yield good performance improvements when dealing with the cross-domain problem, these methods still require a large amount of data from the target domain, hence limiting their application to real-world Person ReID problems. Domain generalization (DG) methods, on the other hand, operate under a more challenging scenario, assuming no access to any target domain data. The common objective among DG methods is to learn a general universal feature representation that is robust to domain shift. DualNorm [11] first introduced instance normalization (IN) in the early stages of the network to normalize the style and content variations of the datasets. On the other hand, MMFA-AAE [17] used a domain adversarial learning approach to remove domain-specific features. Latest DG methods such as DIMN [22], QAConv [15] and M³L [34] used hyper-networks or meta-learning coupled with a memory bank strategy. These meta-learning approaches require complicated training procedures, which makes model optimization difficult. Different from all existing methods, DEX addresses the domain generalization issue from a novel deep feature domain augmentation perspective. Our method uses widely available tools, a ResNet-50 backbone, and does not require complex frameworks or training procedures.

**Augmentation.** Most of the proposed augmentation solutions in Person ReID use GANs [5] to transfer target domain image styles to source domain images. LSRO [37] and DG-Net [38] first used GANs as an augmentation for better representation learning in Person ReID. SPGAN [4] and PTGAN [31] then expanded this to cross dataset scenarios. However, these GAN-based approaches mainly transfer image-style differences across different domains without considering other semantic differences such as clothing styles, weather, etc. Furthermore, training such generative models is difficult and computationally expensive, with the final outputs exhibiting noticeable artifacts. Our method focuses on a new augmentation direction for Person ReID, which directly performs semantic transformations in the deep feature space. Many works [8, 26, 30] discovered that it was possible to meaningfully alter the semantics of image samples (*e.g.* changing the color of an object) by perturbing their corresponding deep features in specific directions. DeepAugment [8] proposed to pass images through a pretrained image-to-image model while altering the features via stochastic operations in the model, producing multiple diverse yet semantically consistent images for training. However, training good image generation models is challenging and time-consuming, and the increase in data leads to increased memory use and training time. ISDA [30] proposed to project deep features by sampling semantic directions from class-conditional covariance matrices, implicitly performing the projections and reducing memory footprint by optimizing a surrogate robust upper-bound loss. Our proposed DEX is tailored for the multi-domain generalization Person ReID problem. It expands feature samples in meaningful domain-semantic directions while preserving person identity information.
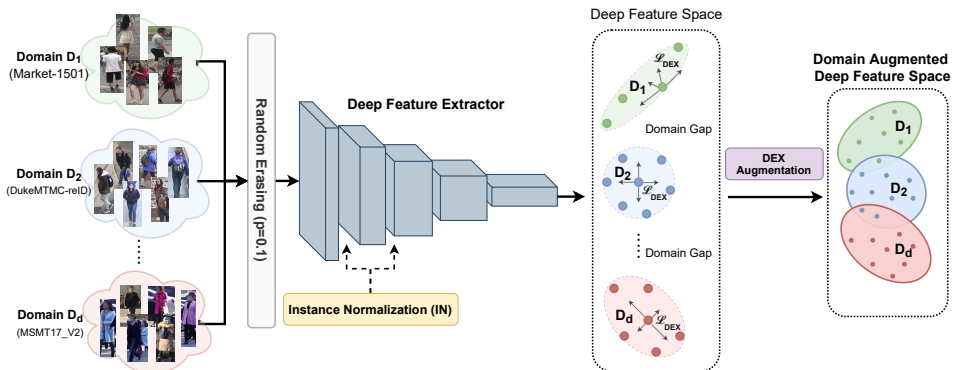
Figure 1: Our proposed solution illustrated. Our baseline is enhanced by reducing the probability of RE and applying IN at early layers of a deep feature extractor. In feature space, our deep augmentation method DEX improves domain manifold coverage by implicitly projecting the feature points in directions of the domain distribution. Best viewed in color.

# 3 Method

## 3.1 Domain Embedding Expansion (DEX)

To improve feature representation learning and close domain gaps, we propose DEX, a deep feature augmentation technique specially adapted for multi-domain generalization problems. Building on previous deep feature augmentation methods [8, 26, 30], we perturb deep features generated by the model as a form of augmentation. Each deep feature is projected along directions sampled from a zero-mean normal distribution with covariance matrix estimated from the feature's domain; this is so that features from different domains are projected in directions that correspond to semantic transformations meaningful to their domain. The covariance matrices are estimated online during training. Similar to ISDA [30], the data samples are not explicitly perturbed. Instead, we optimize a proxy loss function that upperbounds the expected cross-entropy loss of perturbed data samples. Figure 2 presents an overview of our method.

**Formulation** DEX is a modification to the classification branch of the model. The regular softmax loss $\mathcal{L}_{soft}$ is computed from the feature output of the model and the weights of its fully connected layer as such:

$$\mathcal{L}_{soft} = \frac{1}{N}\sum_{i=1}^{N} -\log\left(\frac{\exp(\mathbf{w}_{y_i}^{\mathsf{T}}\mathbf{a_i})}{\sum_{j=1}^{C}\exp(\mathbf{w}_j^{\mathsf{T}}\mathbf{a_i})}\right), \qquad (1)$$

where $N$ is the batch size indexed by $i$, $C$ is the number of unique PIDs indexed by $j$, $\mathbf{a_i}$ are deep features output by the model and $\mathbf{w}$ are weight vectors of the model's fully connected layer; biases are omitted in our classifier layer. The design rationale behind DEX is derived from [30]. First we consider the *expected* softmax loss if the deep features were projected

along domain-conditional covariance directions $\Sigma_{d_i}$:

$$
\begin{aligned}
\mathcal{L}_\infty &= \frac{1}{N} \sum_{i=1}^{N} \mathbb{E}_{\tilde{\mathbf{a}}_{\mathbf{i}}} \left[ -\log \left( \frac{\exp(\mathbf{w}_{y_i}^\mathsf{T} \tilde{\mathbf{a}}_{\mathbf{i}})}{\sum_{j=1}^{C} \exp(\mathbf{w}_j^\mathsf{T} \tilde{\mathbf{a}}_{\mathbf{i}})} \right) \right] \\
&= \frac{1}{N} \sum_{i=1}^{N} \mathbb{E}_{\tilde{\mathbf{a}}_{\mathbf{i}}} \left[ \log \left( \sum_{j=1}^{C} \exp((\mathbf{w}_j^\mathsf{T} - \mathbf{w}_{y_i}^\mathsf{T}) \tilde{\mathbf{a}}_{\mathbf{i}}) \right) \right],
\end{aligned}
\tag{2}
$$

where $\tilde{\mathbf{a}}_{\mathbf{i}} \sim \mathcal{N}(\mathbf{a}_{\mathbf{i}}, \lambda \Sigma_{d_i})$ are the augmented features assumed to be normally distributed around $\mathbf{a}_{\mathbf{i}}$ with domain-conditional covariance $\Sigma_{d_i}$, and $\lambda \geq 0$ controls the strength of the augmentation. Applying Jensen's Inequality, $\mathbb{E}[\log(X)] \leq \log(\mathbb{E}[X])$, we can move the log out of the expectation to get:

$$
\mathcal{L}_\infty \leq \frac{1}{N} \sum_{i=1}^{N} \log \left( \sum_{j=1}^{C} \mathbb{E}_{\tilde{\mathbf{a}}_{\mathbf{i}}} [\exp((\mathbf{w}_j^\mathsf{T} - \mathbf{w}_{y_i}^\mathsf{T}) \tilde{\mathbf{a}}_{\mathbf{i}})] \right)
\tag{3}
$$

We apply the moment generating function $\mathbb{E}[\exp(tX)] = \exp(t\mu + \frac{1}{2}\sigma^2 t^2)$, $X \sim \mathcal{N}(\mu, \sigma^2)$, substituting $t$ with $(\mathbf{w}_j^\mathsf{T} - \mathbf{w}_{y_i}^\mathsf{T})$ and $X \sim \mathcal{N}(\mu, \sigma^2)$ with $\tilde{\mathbf{a}}_{\mathbf{i}} \sim \mathcal{N}(\mathbf{a}_{\mathbf{i}}, \lambda \Sigma_{d_i})$, to derive our loss:

$$
\mathcal{L}_{DEX}(\lambda) = \frac{1}{N} \sum_{i=1}^{N} -\log \left( \frac{\exp(\mathbf{w}_{y_i}^\mathsf{T} \mathbf{a}_{\mathbf{i}})}{\sum\limits_{j=1}^{C} \exp(\mathbf{w}_j^\mathsf{T} \mathbf{a}_{\mathbf{i}} + \frac{\lambda}{2}(\mathbf{w}_j^\mathsf{T} - \mathbf{w}_{y_i}^\mathsf{T}) \Sigma_{d_i}(\mathbf{w}_j - \mathbf{w}_{y_i}))} \right) \geq \mathcal{L}_\infty
\tag{4}
$$

Thus, $\mathcal{L}_{DEX}$ upper bounds the *expected* softmax loss of projecting deep features $\mathbf{a}_{\mathbf{i}}$ over directions encoded in $\Sigma_{d_i}$, and we can directly optimize $\mathcal{L}_{DEX}$ to reap the benefits of augmentation while avoiding the extra computation of explicitly projecting features.

Different from [50] that performs implicit augmentation in class-semantic directions, DEX explores meaningful *domain-semantic directions* in the deep manifold space for different Person ReID datasets. ISDA was proposed in the context of image classification problems and is impractical for multi-domain Person ReID problems where thousands of per-class covariance matrices collectively impose a prohibitive memory overhead. Approximations have been proposed by [50] to overcome this memory issue, but even so, our experiments show that applying ISDA naively fails to improve the baseline for all benchmarks. Our analysis reveals that Person ReID datasets have very few samples-per-class (10-25 on average) which makes covariance estimation unreliable, compared to the datasets used in ISDA which have 500-5,000 on average. DEX is a superior design for two reasons. Firstly, there are much fewer domains than classes so we need not rely on approximations to reduce memory use. Secondly, since the number of samples-per-domain is very large (numbering in the 10,000s), the domain-conditional covariance estimates are stable. The wide domain gaps observed in DG Person ReID show that *deep features invariably encode domain semantics*; consequently, their estimated domain-conditional covariance matrices would contain meaningful domain-semantic directions. Experimentally, DEX significantly improves performance for every benchmark. Ablation studies comparing ISDA with our method are presented in Section 4, and a detailed study of the differences between Person ReID datasets and those studied in ISDA are presented in Supplemental Material.
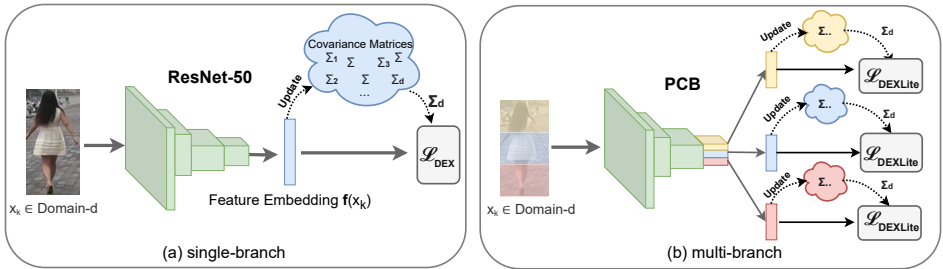
Figure 2: DEX: (a) This illustrates how DEX is applied to a single classifier branch; (b) Given a model with multiple classifier branches, such as PCB shown above, we store and update a separate set of covariance matrices for each branch and also compute a per-branch DEXLite loss. For illustration only, we reduced the PCB network to 3 stripes.

## 3.2 Domain Embedding Expansion Lite (DEXLite)

DEX requires additional memory during training. During each back-propagation step, intermediate tensors require *extra* space on the order of $O(B_{CE}(Df^2 + CN))$. $B_{CE}$ is the number of cross-entropy loss branches that apply DEX, $D$ is the number of domains, $f$ is the feature dimension, $C$ is the number of PIDs in the training set with batch size $N$. By using approximations [30], we can reduce the complexity to $O(B_{CE}(Df + CN))$. However, there are many reasonable scenarios where $B_{CE}$ and $C$ can dominate the complexity. If there are multiple classification loss branches as in the case of PCB [23] and MGN [28], or if the number of PIDs explodes as a result of merging more source domains, memory overhead grows infeasible. For these use cases, we developed a lightweight version of DEX (DEXLite) that applies negative sampling to alleviate this issue. Negative sampling is most popularly used in classification problems with a large number of classes, most notably in training language models to speed up softmax loss computation over a large vocabulary [19]. Computing the full denominator of Equation 4 is memory intensive and wasteful since most classes are negatives; in our sampling strategy, a batch size of $B$ only has $\frac{B}{4}$ unique identities, meaning that if we train on a dataset with 10,000 PIDs, a batch size of 32 contains 8 unique positive IDs that account for only 0.08% of all identities. We propose to sample only a subset of negatives to reduce the total number of classes considered during DEX augmentation. The loss function of Equation 4 then changes to:

$$\mathcal{L}_{DEXLite}(\lambda) = \frac{1}{N}\sum_{i=1}^{N} -\log\left(\frac{\exp(\mathbf{w}_{y_i}^{\mathsf{T}}\mathbf{a_i})}{\sum_{j\in P_s}\exp(\mathbf{w}_j^{\mathsf{T}}\mathbf{a_i} + \frac{\lambda}{2}(\mathbf{w}_j^{\mathsf{T}} - \mathbf{w}_{y_i}^{\mathsf{T}})\Sigma_{d_i}(\mathbf{w}_j - \mathbf{w}_{y_i}))}\right), \quad (5)$$

where $P_s$ is the set of sampled PIDs, including positives; the space complexity is reduced to $O(B_{CE}(Df + |P_s|N))$, with $|P_s| \ll C$. The development of DEXLite enables us to apply this augmentation onto a wider range of models and datasets with many PIDs. As an added bonus, it also speeds up training. To demonstrate its capability, we implement DEXLite on MGN and PCB and report significant improvements in Section 4.

## 3.3 Overall Loss Function

Our proposed method is based on simple ResNet-50 backbone and trained with three losses: softmax loss over PIDs $\mathcal{L}_{soft}$, triplet loss $\mathcal{L}_{tri}$ [9] and center loss $\mathcal{L}_{cen}$ [32]. $\mathcal{L}_{soft}$ is defined

in Equation 1 and definitions of $\mathcal{L}_{tri}$ and $\mathcal{L}_{cen}$ are provided in Supplemental Material. We train the baseline with combined loss $\mathcal{L}_{base}$ with $\beta_{soft} = 1.0$, $\beta_{tri} = 1.0$, and $\beta_{cen} = 5 \times 10^{-4}$:

$$\mathcal{L}_{base} = \beta_{soft}\mathcal{L}_{soft} + \beta_{tri}\mathcal{L}_{tri} + \beta_{cen}\mathcal{L}_{cen} \tag{6}$$

The overall loss $\mathcal{L}_{overall}(t)$ is parameterized by current epoch $t$ which controls the strength of the implicit augmentation. $\mathcal{L}_{DEX}$ and $\mathcal{L}_{DEXLite}$ are defined in Equation 4 and Equation 5. At epoch $t$, $\lambda_t = \frac{t-1}{T-1}\lambda$ with $\lambda = 7.5$; we gradually increase $\lambda_t$ to pay more attention to later model features as they become more informative over time. For baseline and DEX we train for 60 epochs ($T = 60$) with a batch size of 32 using the Adam [13] optimizer with a learning rate schedule similar to [18]. Full details are available in Supplemental Material.

$$\mathcal{L}_{overall}(t) = \mathcal{L}_{DEX/DEXLite}(\lambda_t) + \beta_{tri}\mathcal{L}_{tri} + \beta_{cen}\mathcal{L}_{cen} \tag{7}$$

# 4 Experiments

## 4.1 Datasets and Settings

We model our experiments after the most recent state-of-the-art method M³L [34] in DG Person ReID in four large-scale benchmarks: Market-1501 [35], DukeMTMC-reID [37], CUHK03 [27] (or its new partition CUHK03-NP [39]) and MSMT17_V2 [31]. Table 1 breaks down the number of IDs and images in each of the training, query and gallery splits for each dataset. For simplicity and clarity, we denote Market-1501, MSMT17, DukeMTMC-reID, CUHK03, and CUHK03-NP as M, MS, D, C, and C-NP.

| Dataset | Abbreviation | Train-IDs | Train-Images | Query-IDs | Query-Images | Gallery-IDs | Gallery-Images |
|---------|--------------|-----------|--------------|-----------|--------------|-------------|----------------|
| Market-1501 [35] | M | 751 | 12,936 | 750 | 3,368 | 750 | 15,913 |
| MSMT17_V2 [31] | MS | 1,041 | 32,621 | 3,060 | 11,659 | 3,060 | 82,161 |
| DukeMTMC-reID [37] | D | 702 | 16,522 | 702 | 2,228 | 1,110 | 17,661 |
| CUHK03-NP [39] | C-NP | 767 | 7,365 | 700 | 1,400 | 700 | 5,332 |
| CUHK03 [27] | C | 1,367 | 26,263 | - | - | - | - |

Table 1: Dataset details. For testing we always use the query/gallery split of C-NP. We follow [34] to select either C or C-NP as source domain. C is never used for testing.

## 4.2 Comparison with State-of-the-art Methods under New Evaluation

Following the new evaluation methodology proposed in [34], we use the detected test subset of the CUHK03 new protocol, CUHK-NP (detected), for testing and CUHK03 as one of the source domains for training. The training splits of any three datasets are combined into a training set and the query/gallery split of the remaining dataset is used for testing.

Table 2 compares our proposed solution **DEX** against several recent state-of-the-art (SOTA) methods such as DualNorm [11], QAConv [15] and M³L [34]. All experiments are evaluated on the new large scale DG Person ReID benchmarks. For clarity of presentation and alignment with subsequent ablation studies, extra experiments using CUHK-NP as one of the source domains (which were added later in [34]) are presented in Supplemental Material. Nevertheless, our proposed solution surpasses the most recent state-of-the-art method [34] in *all* experimental settings by a significant margin.

| Sources | Method | Market-1501 | | Sources | Method | DukeMTMC-reID | |
|---|---|---|---|---|---|---|---|
| | | Rank-1 | mAP | | | Rank-1 | mAP |
| C+D+MS | DualNorm$_{50}$ | 78.9 | 52.3 | C+M+MS | DualNorm$_{50}$ | 68.5 | 51.7 |
| | QAConv$_{50}$ | 65.7 | 35.6 | | QAConv$_{50}$ | 66.1 | 47.1 |
| | M$^3$L (ResNet-50) | 74.5 | 48.1 | | M$^3$L (ResNet-50) | 69.4 | 50.5 |
| | M$^3$L (IBN-Net50) | 75.9 | 50.2 | | M$^3$L (IBN-Net50) | 69.2 | 51.1 |
| | DEX (Ours) | **81.5** | **55.2** | | DEX (Ours) | **73.7** | **55.0** |

| Sources | Method | MSMT17_V2 | | Sources | Method | CUHK-NP | |
|---|---|---|---|---|---|---|---|
| | | Rank-1 | mAP | | | Rank-1 | mAP |
| C+D+M | DualNorm$_{50}$ | 37.9 | 15.4 | D+M+MS | DualNorm$_{50}$ | 28.0 | 27.6 |
| | QAConv$_{50}$ | 24.3 | 7.5 | | QAConv$_{50}$ | 23.5 | 21.0 |
| | M$^3$L (ResNet-50) | 33.0 | 12.9 | | M$^3$L (ResNet-50) | 30.7 | 29.9 |
| | M$^3$L (IBN-Net50) | 36.9 | 14.7 | | M$^3$L (IBN-Net50) | 33.1 | 32.1 |
| | DEX (Ours) | **43.5** | **18.7** | | DEX (Ours) | **36.7** | **33.8** |

Table 2: Comparison with state-of-the-art for DG Person ReID. Bold numbers denote highest scores, while underlined numbers denote second-highest. With DEX augmentation, our model surpasses the state-of-the-art in all benchmarks.

## 4.3 Comparison with State-of-the-art Methods under Old Evaluation

While the preceding sections focused on new evaluation benchmarks set by M$^3$L [34], we appreciate that much of previous work on DG Person ReID based their evaluations on the old small-scale datasets VIPeR [6], PRID [10], QMUL-GRID [2], and i-LIDS [36]. For completeness, we present our evaluation of DEX on these small-scale benchmarks. Table 3 compares our DEX under this small-dataset evaluation scheme against other current state-of-the-art DG Person ReID methods. Following standard evaluation methodology, we trained on multiple large-scale benchmark datasets CUHK02 [14], CUHK03 [27], CUHK-SYSU [21], Market-1501 [55] and DukeMTMC-reID [57] and compared against other state-of-the-art methods trained under the same setting. Combining all source datasets resulted in a total of 121,765 images with 18,530 unique PIDs. Because the large number of PIDs takes up a significant GPU memory overhead for DEX, we trained our model using DEXLite instead, sampling 2,000 PIDs to reduce memory use and speed up training.

| Method | VIPeR (V) | | | | PRID (P) | | | | GRID (G) | | | | i-LIDS (L) | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | R-1 | R-5 | R-10 | mAP | R-1 | R-5 | R-10 | mAP | R-1 | R-5 | R-10 | mAP | R-1 | R-5 | R-10 | mAP |
| AugMining [30] | 49.8 | 70.8 | 77.0 | - | 34.3 | 56.2 | 65.7 | - | 46.6 | 67.5 | 76.1 | - | 76.3 | 93.0 | 95.3 | - |
| DDAN [1] | 56.5 | 65.6 | 76.3 | 60.8 | 62.9 | 74.2 | 85.3 | 67.5 | 46.2 | 55.4 | 68.0 | 50.9 | 78.0 | 85.7 | 93.2 | 81.2 |
| DIMN [31] | 51.2 | 70.2 | 76.0 | 60.1 | 39.2 | 67.0 | 76.7 | 52.0 | 29.3 | 53.3 | 65.8 | 41.1 | 70.2 | 89.7 | 94.5 | 78.4 |
| DIR-ReID [53] | 58.3 | 66.9 | 77.3 | 62.9 | **71.1** | 82.4 | 88.6 | 75.6 | 47.8 | 51.1 | 70.5 | 52.1 | 74.4 | 83.1 | 90.2 | 78.6 |
| DualNorm$_{resnet}$ [13] | 59.4 | - | - | - | 69.6 | - | - | - | 43.7 | - | - | - | 78.2 | - | - | - |
| MMFA-AAE [20] | 58.4 | - | - | - | 57.2 | - | - | - | 47.4 | - | - | - | 84.8 | - | - | - |
| RaMoE [8] | 56.6 | - | - | 64.6 | 57.7 | - | - | 67.3 | 46.8 | - | - | 54.2 | 85.0 | - | - | 90.2 |
| SNR [17] | 52.9 | - | - | 61.3 | 52.1 | - | - | 66.5 | 40.2 | - | - | 47.7 | 84.1 | - | - | 89.9 |
| BCaR [4] | **65.8** | - | - | - | 70.2 | - | - | - | 52.8 | - | - | - | 81.3 | - | - | - |
| **DEXLite (Ours)** | 65.5 | 79.2 | 83.6 | 72.0 | 71.0 | 87.8 | 92.5 | 78.5 | 53.3 | 69.4 | 79.0 | 61.7 | 86.3 | 95.2 | 97.3 | 90.7 |

Table 3: Evaluation on small-scale benchmarks VIPeR, GRID, PRID and i-LIDS.

DEXLite demonstrates good all-round performance, surpassing the state-of-the-art methods in GRID and i-LIDS for all measures and coming in first for Rank-5, Rank-10 and mAP for all benchmarks. For VIPeR and PRID, our method is second place for Rank-1, closely matching the top performers. It is interesting to note that our Rank-5, Rank-10 and mAP scores surpass other methods by a significant margin even without re-ranking. This indicates that application of DEX improves feature generalization such that **all positives** in the gallery obtain a better ranking, and attests to the effectiveness of DEX as a plug-and-play method.

## 4.4 Ablation Study on the Effects of each Technique

Table 4 studies the effects of instance normalization (IN), reducing probability of random erasing (RE), ISDA [30], and DEX to the BagTricks [18] baseline. Adding IN yields the most significant benefit. Reducing the probability of RE or applying DEX also yield significant generalization benefits. Combined with IN, we tested two configurations of RE, in the first removing the augmentation entirely (NoRE), and in the other reducing the probability of applying RE on a sample from 0.5 to 0.1 (RE(0.1)). A small probability of RE seems to improve model generalization slightly. Using just IN and RE(0.1) can yield very competitive performance in all benchmarks except for CUHK-NP. Adding ISDA does not always improve the baseline. With DEX, performance gains are more consistent and are especially significant in the case of CUHK-NP with a **7.7%** improvement in Rank-1 and **5%** increase in mAP that outperforms the current state-of-the-art by a large margin.

| Sources | Method | Market-1501 | | Sources | Method | DukeMTMC-reID | |
|---|---|---|---|---|---|---|---|
| | | Rank-1 | mAP | | | Rank-1 | mAP |
| C+D+MS | BagTricks | 71.6 | 43.3 | C+M+MS | BagTricks | 58.2 | 40.8 |
| | +IN (=DualNorm) | 78.9 | 52.3 | | +IN (=DualNorm) | 68.5 | 51.7 |
| | +RE(0.1) | 74.0 | 45.6 | | +RE(0.1) | 63.1 | 44.6 |
| | +DEX | 71.9 | 45.4 | | +DEX | 66.1 | 48.1 |
| | +IN+NoRE | 80.8 | 54.0 | | +IN+NoRE | 70.4 | 52.8 |
| | +IN+RE(0.1) | 81.0 | 54.3 | | +IN+RE(0.1) | 71.0 | 53.4 |
| | +IN+RE(0.1)+ISDA | 79.6 | 53.9 | | +IN+RE(0.1)+ISDA | 72.3 | 54.5 |
| | +IN+RE(0.1)+DEX (**Ours**) | **81.5** | **55.2** | | +IN+RE(0.1)+DEX (**Ours**) | **73.7** | **55.0** |

| Sources | Method | MSMT17_V2 | | Sources | Method | CUHK-NP | |
|---|---|---|---|---|---|---|---|
| | | Rank-1 | mAP | | | Rank-1 | mAP |
| C+D+M | BagTricks | 19.4 | 6.9 | D+M+MS | BagTricks | 20.1 | 19.6 |
| | +IN (=DualNorm) | 37.9 | 15.4 | | +IN (=DualNorm) | 28.0 | 27.6 |
| | +RE(0.1) | 23.5 | 8.5 | | +RE(0.1) | 23.9 | 23.5 |
| | +DEX | 22.9 | 8.7 | | +DEX | 25.9 | 25.1 |
| | +IN+NoRE | 42.0 | 17.1 | | +IN+NoRE | 28.0 | 28.2 |
| | +IN+RE(0.1) | 42.4 | 17.5 | | +IN+RE(0.1) | 29.0 | 28.8 |
| | +IN+RE(0.1)+ISDA | 41.8 | 17.7 | | +IN+RE(0.1)+ISDA | 32.6 | 32.4 |
| | +IN+RE(0.1)+DEX (**Ours**) | **43.5** | **18.7** | | +IN+RE(0.1)+DEX (**Ours**) | **36.7** | **33.8** |

Table 4: Ablation study comparing the effects of applying IN, reducing RE probability, applying ISDA; and applying DEX on a strong baseline ResNet-50 model from BagTricks [11]

## 4.5 Applying DEXLite to Multi-branch Architectures

We demonstrate general applicability of DEX by applying it on multi-branch architectures that incur a large increase in memory use. For these architectures, we can limit the sample size to apply DEXLite effectively. We selected two well-known multi-branch models in supervised Person ReID: PCB [23], and MGN [28]. PCB divides the final feature tensor into six horizontal stripes, each of which becomes a local PID prediction branch. We apply DEXLite each of these local branches in PCB. MGN consists of eight PID prediction branches, with three global-level features and five part-level features. We apply DEXLite to the three global-level prediction branches to contrast PCB's part-level design.

Figure 2(b) illustrates the multi-branch application of our technique using PCB as an example. MGN's loss is averaged among all branches, so we kept $\lambda$ at 7.5, same as our original baseline experiments. However, PCB sums the losses from all branches instead of averaging so we reduced $\lambda$ by a factor of six (the number of stripes in PCB) to 1.25. We set the sample size to 2,000 for DEXLite and train for 40 epochs. Table 5 shows the result of our comparison study. Overall, DEX demonstrates its effectiveness as an augmentation strategy and ability to improve the generalization results of a range of models.

| Sources | Model | Method | Market-1501 Rank-1 | mAP | Sources | Model | Method | DukeMTMC-reID Rank-1 | mAP |
|---|---|---|---|---|---|---|---|---|---|
| C+D+MS | MGN [28] | +IN+RE(0.1) | 62.9 | 29.7 | C+M+MS | MGN [28] | +IN+RE(0.1) | 62.2 | 40.8 |
| | | +IN+RE(0.1)+DEXLite | **63.7** | **32.8** | | | +IN+RE(0.1)+DEXLite | **64.3** | **44.4** |
| | PCB [23] | +IN+RE(0.1) | 71.2 | 42.4 | | PCB [23] | +IN+RE(0.1) | 65.5 | 45.1 |
| | | +IN+RE(0.1)+DEXLite | **73.1** | **45.1** | | | +IN+RE(0.1)+DEXLite | **66.5** | **46.8** |

| Sources | Model | Method | MSMT17_V2 Rank-1 | mAP | Sources | Model | Method | CUHK-NP Rank-1 | mAP |
|---|---|---|---|---|---|---|---|---|---|
| C+D+M | MGN [28] | +IN+RE(0.1) | 23.1 | 8.3 | D+M+MS | MGN [28] | +IN+RE(0.1) | 18.1 | 15.5 |
| | | +IN+RE(0.1)+DEXLite | **25.9** | **9.6** | | | +IN+RE(0.1)+DEXLite | **22.9** | **21.0** |
| | PCB [23] | +IN+RE(0.1) | 36.4 | 14.6 | | PCB [23] | +IN+RE(0.1) | 25.2 | 25.2 |
| | | +IN+RE(0.1)+DEXLite | **37.3** | **15.2** | | | +IN+RE(0.1)+DEXLite | **28.4** | **27.2** |

Table 5: Applying DEXLite to multi-branch architecture models such MGN [28] and PCB [23] similarly yield performance improvements

## 4.6 Ablation Study on Negative Sample Sizes for DEXLite

We investigate the effects of different sample sizes when applying negative sampling in DEXLite, shown in Table 6. The performance improves as the sample size increases but starts to plateau beyond a sample size of around 2,000. With 2,000 negative sample sizes for estimation, DEXLite can achieve comparable performance on par with the full DEX with much faster training speed and memory usage.

| Sources | Type | Samples | Market-1501 Rank-1 | mAP | Sources | Type | Samples | DukeMTMC-reID Rank-1 | mAP |
|---|---|---|---|---|---|---|---|---|---|
| C+D+MS | DEXLite | 10 | 54.6 | 30.0 | C+M+MS | DEXLite | 10 | 53.2 | 34.2 |
| | | 100 | 69.7 | 42.8 | | | 100 | 64.8 | 46.0 |
| | | 1000 | 77.0 | 51.4 | | | 1000 | 70.8 | 52.6 |
| | | 2000 | 80.3 | 54.3 | | | 2000 | 72.1 | 53.6 |
| | DEX | Full | 81.5 | 55.2 | | DEX | Full | 72.7 | 54.2 |

| Sources | Type | Sample | MSMT17_V2 Rank-1 | mAP | Sources | Type | Samples | CUHK-NP Rank-1 | mAP |
|---|---|---|---|---|---|---|---|---|---|
| D+M+MS | DEXLite | 10 | 16.8 | 16.8 | C+D+M | DEXLite | 10 | 17.5 | 6.5 |
| | | 100 | 27.6 | 27.0 | | | 100 | 31.3 | 12.7 |
| | | 1000 | 32.8 | 32.1 | | | 1000 | 38.6 | 16.4 |
| | | 2000 | 34.4 | 33.3 | | | 2000 | 41.5 | 17.6 |
| | DEX | Full | 34.3 | 32.9 | | DEX | Full | 43.5 | 18.7 |

Table 6: Effects of sample sizes in DEXLite in comparison with DEX

## 5 Conclusion

In this paper, we introduced a fresh perspective and a novel solution to the problem of domain generalization (DG) in Person ReID, leveraging on domain biases inherently encoded in deep features to augment them directly in domain-semantically meaningful directions. DEX, our proposed dual-label implicit augmentation method applied in simple ResNet50 network can surpasses the most of recent state-of-the-art Domain Generalization (DG) Person ReID methods in both new large-scale banchmarks and old small-scale benchmarks by a large margins. Reducing the memory use of our method with negative sampling techniques, we also developed DEXLite to make our method applicable to a wide range of model architectures and dataset situations. There is still room to grow before DG Person ReID solutions are robust enough for true real-world use, and we believe that our novel approach will seed more developments along this new paradigm and pave the way for faster innovation.

# References

[1] Peixian Chen, Pingyang Dai, Jianzhuang Liu, Feng Zheng, Qi Tian, and Rongrong Ji. Dual Distribution Alignment Network for Generalizable Person Re-Identification. In *Proc. AAAI Conference on Artificial Intelligence (AAAI)*, 2020.

[2] Chen Change Loy, Tao Xiang, and Shaogang Gong. Multi-camera activity correlation analysis. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1988–1995, 2009.

[3] Yongxing Dai, Xiaotong Li, Jun Liu, Zekun Tong, and Ling-Yu Duan. Generalizable Person Re-identification with Relevance-aware Mixture of Experts. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021.

[4] 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 *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 994–1003, 2018.

[5] Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative Adversarial Nets. In *Proc. Advances in Neural Information Processing Systems (NeurIPS)*, page 2672–2680, 2014.

[6] Doug Gray, Shane Brennan, and Hai Tao. Evaluating Appearance Models for Recognition, Reacquisition, and Tracking. In *International Workshop on Performance Evaluation for Tracking and Surveillance (PETS)*, 2007.

[7] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep Residual Learning for Image Recognition. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778, 2016.

[8] Dan Hendrycks, Steven Basart, Norman Mu, Saurav Kadavath, Frank Wang, Evan Dorundo, Rahul Desai, Tyler Zhu, Samyak Parajuli, Mike Guo, Dawn Song, Jacob Steinhardt, and Justin Gilmer. The Many Faces of Robustness: A Critical Analysis of Out-of-Distribution Generalization. *arXiv preprint*, 2020.

[9] Alexander Hermans, Lucas Beyer, and Bastian Leibe. In Defense of the Triplet Loss for Person Re-Identification. In *arXiv preprint*, 3 2017.

[10] Martin Hirzer, Csaba Beleznai, Peter M Roth, and Horst Bischof. Person Re-identification by Descriptive and Discriminative Classification. In *Scandinavian Conference on Image Analysis (SCIA)*, 2011.

[11] Jieru Jia, Qiuqi Ruan, and Timothy M. Hospedales. Frustratingly Easy Person Re-Identification: Generalizing Person Re-ID in Practice. In *Proc. British Machine Vision Conference (BMVC)*, 2019.

[12] Xin Jin, Cuiling Lan, Wenjun Zeng, Zhibo Chen, and Li Zhang. Style Normalization and Restitution for Generalizable Person Re-identification. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.

[13] Diederik P. Kingma and Jimmy Ba. Adam: A Method for Stochastic Optimization. In *Proc. International Conference on Learning Representations (ICLR)*, 2015.

[14] Wei Li and Xiaogang Wang. Locally Aligned Feature Transforms across Views. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3594–3601, 2013.

[15] Shengcai Liao and Ling Shao. Interpretable and Generalizable Person Re-identification with Query-Adaptive Convolution and Temporal Lifting. In *Proc. European Conference on Computer Vision (ECCV)*, 2020.

[16] Shan Lin, Haoliang Li, Chang-Tsun Li, and Alex C Kot. Multi-task Mid-level Feature Alignment Network for Unsupervised Cross-Dataset Person Re-Identification. In *Proc. British Machine Vision Conference (BMVC)*, 2018.

[17] Shan Lin, Chang-tsun Li, and Alex C Kot. Multi-Domain Adversarial Feature Generalization for Person Re-Identification. *IEEE Transactions on Image Processing*, 2021.

[18] Hao Luo, Youzhi Gu, Xingyu Liao, Shenqi Lai, and Wei Jiang. Bag of Tricks and a Strong Baseline for Deep Person Re-Identification. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 1487–1495, 2019.

[19] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. In *Proc. International Conference on Learning Representations Workshop (ICLRW)*, pages 1–12, 2013.

[20] Peixi Peng, Tao Xiang, Yaowei Wang, Massimiliano Pontil, Shaogang Gong, Tiejun Huang, and Yonghong Tian. Unsupervised Cross-Dataset Transfer Learning for Person Re-identification. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1306–1315, 2016.

[21] Siyuan Qiao, Chenxi Liu, Wei Shen, and Alan Yuille. Few-Shot Image Recognition by Predicting Parameters from Activations. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7229–7238, 2018. ISBN 978-1-5386-6420-9. doi: 10.1109/CVPR.2018.00755.

[22] Jifei Song, Yongxin Yang, Yi-Zhe Song, Tao Xiang, and Timothy M. Hospedales. Generalizable Person Re-Identification by Domain-Invariant Mapping Network. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

[23] Yifan Sun, Liang Zheng, Yi Yang, Qi Tian, and Shengjin Wang. Beyond Part Models: Person Retrieval with Refined Part Pooling. In *Proc. European Conference on Computer Vision (ECCV)*, 2018.

[24] Masato Tamura and Tomokazu Murakami. Augmented Hard Example Mining for Generalizable Person Re-Identification. In *arXiv preprint*, 2019.

[25] Masato Tamura and Tomoaki Yoshinaga. BCaR : Beginner Classifier as Regularization Towards Generalizable Re-ID. In *Proc. British Machine Vision Conference (BMVC)*, 2020.

[26] Paul Upchurch, Jacob Gardner, Geoff Pleiss, Robert Pless, Noah Snavely, Kavita Bala, and Kilian Weinberger. Deep feature interpolation for image content changes. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6090–6099, 2017.

[27] Faqiang Wang, Wangmeng Zuo, Liang Lin, David Zhang, and Lei Zhang. Joint Learning of Single-Image and Cross-Image Representations for Person Re-identification. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1288–1296, 2016.

[28] Guanshuo Wang, Yufeng Yuan, Xiong Chen, Jiwei Li, and Xi Zhou. Learning Discriminative Features with Multiple Granularities for Person Re-Identification. In *Proc. ACM International Conference on Multimedia (ACM MM)*, pages 274–282, 2018.

[29] Jingya Wang, Xiatian Zhu, Shaogang Gong, and Wei Li. Transferable Joint Attribute-Identity Deep Learning for Unsupervised Person Re-identification. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2275–2284, 2018.

[30] Yulin Wang, Xuran Pan, Shiji Song, Hong Zhang, Cheng Wu, and Gao Huang. Implicit Semantic Data Augmentation for Deep Networks. In *Proc. Advances in Neural Information Processing Systems (NeurIPS)*, 2019.

[31] Longhui Wei, Shiliang Zhang, Wen Gao, and Qi Tian. Person Transfer GAN to Bridge Domain Gap for Person Re-identification. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 79–88, 2018.

[32] Yandong Wen, Kaipeng Zhang, Zhifeng Li, and Yu Qiao. A Discriminative Feature Learning Approach for Deep Face Recognition. In *Proc. European Conference on Computer Vision (ECCV)*, pages 499–515, 2016.

[33] Yi-Fan Zhang, Hanlin Zhang, Zhang Zhang, Da Li, Zhen Jia, Liang Wang, and Tieniu Tan. Learning Domain Invariant Representations for Generalizable Person Re-Identification. *arXiv preprint*, 2021.

[34] Yuyang Zhao, Zhun Zhong, Fengxiang Yang, Zhiming Luo, Yaojin Lin, Shaozi Li, and Nicu Sebe. Learning to Generalize Unseen Domains via Memory-based Multi-Source Meta-Learning for Person Re-Identification. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021.

[35] Liang Zheng, Liyue Shen, Lu Tian, Shengjin Wang, Jingdong Wang, and Qi Tian. Scalable Person Re-identification: A Benchmark. In *Proc. IEEE International Conference on Computer Vision (ICCV)*, pages 1116–1124, 2015.

[36] Wei Shi Zheng, Shaogang Gong, and Tao Xiang. Associating Groups of People. In *Proc. British Machine Vision Conference (BMVC)*, pages 23–1, 2009. ISBN 1901725391. doi: 10.5244/C.23.23.

[37] Zhedong Zheng, Liang Zheng, and Yi Yang. Unlabeled Samples Generated by GAN Improve the Person Re-identification Baseline in Vitro. In *Proc. IEEE International Conference on Computer Vision (ICCV)*, pages 3774–3782, 2017.

[38] Zhedong Zheng, Xiaodong Yang, Zhiding Yu, Liang Zheng, Yi Yang, and Jan Kautz. Joint Discriminative and Generative Learning for Person Re-Identification. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2133–2142, 2019.

[39] Zhun Zhong, Liang Zheng, Donglin Cao, and Shaozi Li. Re-ranking person re-identification with k-reciprocal encoding. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3652–3661, 2017.