Latent Distribution Alignment for Domain Generalizable Person Re-identification

Ren Nie
School of Automation Engineering
University of Electronic Science and Technology of China
Chengdu, China
nieren@std.uestc.edu.cn

Jin Ding
School of Automation Engineering
University of Electronic Science and Technology of China
Chengdu, China
1121716558@qq.com

Lingxiao He
Meituan
Beijing, China
xiaomingzhidao1@gmail.com

Xue Zhou*
Shenzhen Institute for Advanced Study, UESTC
Shenzhen, China
School of Automation Engineering
University of Electronic Science and Technology of China
Chengdu, China
zhouxue@uestc.edu.cn

***Abstract*—Domain Generalizable Person Re-identification (DG ReID) is a practical task that generalizes the model trained on multiple source domains to unseen target domains without fine-tuning. Existing methods usually combine diverse normalization techniques to remove style information while retaining discriminative features. However, they overlook the absence of the target domain while training. In this work, we propose a Latent Distribution Alignment (LDA) method to indirectly align source and target domains through the dynamically constructed latent distribution without any added learnable parameters. Specifically, we design an expert network to dynamically construct a latent distribution and store the domain-specific representation information by fully utilizing batch-wise statistical parameters. Subsequently, one source domain, acting as the mimical target domain of the remaining source domains, explicitly aligns with the latent distribution through an instance-wise domain alignment network through Anti-Normalization (AN). Extensive experiments show that our method is simple yet effective in enhancing the generalizable capability.**

***Index Terms*—Domain Generalizable Person Re-identification, Distribution Alignment, Normalization Layer**

I. Introduction

Person re-identification (ReID) constitutes a necessary component within the field of computer vision, it aims at retrieving identical persons in a multi-camera network having non-overlapping field-of-views. Most existing approaches [1], [2] can achieve remarkable performance when both training and testing of models on the same domain. However, when these well-trained models are applied to unfamiliar testing scenarios, an issue known as ”domain shift” arises [3], significantly impacting the performance of the models.

To alleviate this issue, recent efforts are devoted to two approaches: domain adaptive (DA) ReID [4], [5] and domain generalizable (DG) ReID [6]–[13]. Diverging from the DA ReID, which leverages labeled source domains training and subsequently employs unlabeled target domain datasets for unsupervised fine-tuning. DG ReID trains on the source domains and then directly transfers this learned knowledge to the unexplored target domain. This paper primarily focuses on DG ReID, exploring how to achieve person re-identification in unfamiliar domains.

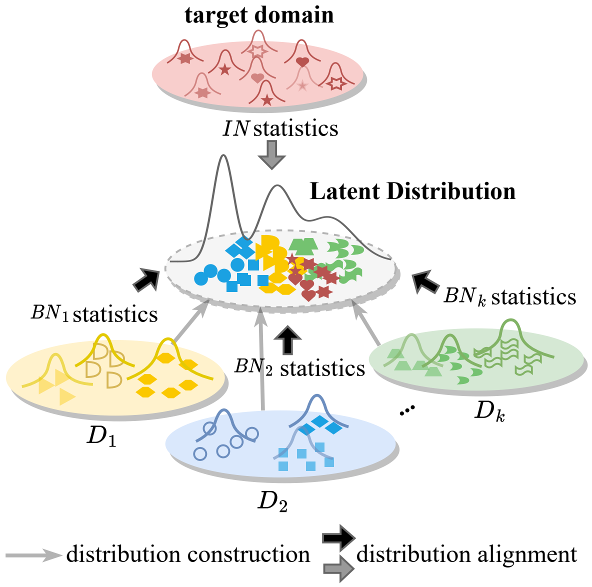

Fig. 1. Illustration of LDA motivation, involves constructing a latent distribution using statistical parameters such as mean and variance from multiple source domains $(D_1, D_2, ..., D_k)$. The model learns to adaptively align the statistical parameters of both source and target domains with the latent distribution.

To overcome the domain shift problem and the absence

This work was supported by the Natural Science Foundation of China (No. 62372082, 61972071, U20A20184) and Natural Science Foundation of Sichuan Province (No.2023NSFSC0485).

* Corresponding author.

979-8-3503-9015-5/24/$31.00 ©2024 IEEE

of target domains, existing methods have made significant progress in DG ReID. Among these works, a concise and effective approach is to modify the normalization layers by combining Batch Normalization (BN) and Instance Normalization (IN) in different ways [9], [14] to alleviate style discrepancies. Nevertheless, these methods mainly focus on preserving mutually discriminative features and eliminating domain-specific information from the provided source domains. They indirectly anticipate the effectiveness of this domain-invariant information in an unknown target domain; however, the network has not yet acquired the capability to directly generalize existing learned information to unknown target domains.

To address these limitations, we introduce a specialized normalization layer that treats one source domain as a mimical target domain of other source domains. We also design a latent distribution to indirectly align source and target domains by explicitly using diverse normalization statistical parameters. Thus, we begin our approach by constructing a latent distribution using the statistical parameters (i.e., mean and variance) of each source domain from the output of the backbone. Subsequently, we adaptively map each source domain and each target domain sample into this latent distribution space, as illustrated in Figure 1.

Specifically, we propose a novel Latent Distribution Alignment (LDA) framework for DG ReID without any added learnable parameters. As shown in Figure 2, our framework consists of two sub-networks: an expert network and a domain-aligned network. In the expert network, we design a domain-specific Buffer Bank (BB) to dynamically store the BN information for each source domain across different normalization layers. After passing through the backbone of the expert network, we use the output BN statistics of each source domain to dynamically construct and constrain a latent distribution. Subsequently, with the help of our Anti-Normalization (AN) embedding, the source domain, acting as the mimical target domain of the remaining source domains, will proceed into the domain-aligned network which is responsible for maintaining the latent distribution and explicitly aligning the output IN statistics with the latent distribution. Additionally, we introduce a consistency loss and an alignment loss to regulate the distribution. Overall, our proposed LDA framework can effectively alleviate domain shift and generalize well to unseen target domains for DG ReID. Our work makes the following contributions:

- We propose a novel Latent Distribution Alignment (LDA) framework for DG ReID, which is the first to dynamically construct a latent distribution, through which BN statistics of source domains and IN statistics of target domains are aligned directly.
- We introduce two operations based on normalization layers, namely Buffer Bank (BB) and Anti-Normalization (AN), which treat each source domain as the mimical target domain of remaining source domains, effectively improving the generalization of the model on unseen target domains without any added learnable parameters.
- Extensive experiments conducted on various DG ReID protocols demonstrate the simplicity yet effectiveness and comparable or even better state-of-the-art performance of our LDA framework.

II. Methodology

A. Preliminary

In DG ReID, we have access to K source domains (datasets) for training. We denote the k-th source domain as $D_k, k \in \{1, 2, ..., K\}$. In the training phase, each mini-batch is collected from one randomly selected source domain. The mini-batch will first be input to the expert network and then put it back into the domain-aligned network at each iteration.

Assume that $X_k \in \mathbb{R}^{N \times C \times H \times W}$ is an input feature map from a certain domain D_k, where $N, C, H,$ and W denote the batch size, the number of channels, the height, and the width, respectively. $\mu^{bn}, \sigma^{bn^2} \in \mathbb{R}^C$ are statistical parameters of Batch Normalization (BN), $\mu^{in}, \sigma^{in^2} \in \mathbb{R}^{N \times C}$ are the statistical parameters of instance normalization (IN) of the input feature map.

B. Expert Network

Previous studies [15], [16] have shown that Batch Normalization (BN) can extract domain-specific discriminative information. As shown in Figure 2, from the perspective of distribution alignment, we design an expert network that utilizes the BN statistics of each source domain from the backbone(*i.e.*, $\tilde{\mu}_k^{bn}, \tilde{\sigma}_k^{bn^2}$) to dynamically construct and constrain a latent distribution. Additionally, a Buffer Bank (BB) is employed to store domain-specific BN information (*i.e.*, μ_k^{pop} , $\sigma_k^{pop^2}$) for each source domain across different normalization layers.

Latent Distribution. We assume that after inputting a mini-batch from different source domains into the backbone, the BN statistical parameters of its output, including mean and variance(*i.e.*, $\tilde{\mu}_k^{bn}, \tilde{\sigma}_k^{bn^2}$), center around a single distribution. Thus, we first design a memory-based module to store these statistical parameters from the expert network of each source domain.

Similar to the moving average operation, we update the memory with the mean and variance of BN (*i.e.*, $\tilde{\mu}_k^{bn}, \tilde{\sigma}_k^{bn^2}$) in the current mini-batch during the training phase. The memory is updated by:

$$\begin{aligned} \mathcal{M}[0][k] &\leftarrow \omega \mathcal{M}[0][k] + (1 - \omega) \tilde{\mu}_k^{bn} \\ \mathcal{M}[1][k] &\leftarrow \omega \mathcal{M}[1][k] + (1 - \omega) \tilde{\sigma}_k^{bn^2} \end{aligned} \tag{1}$$

where $\omega \in [0, 1]$ is the ratio used for updating. Through constantly updating the memory, we obtain the domain-specific distribution of each source domain from the expert network.

Then we utilize the concept of weighting to obtain the latent distribution(*i.e.*, μ_p, σ_p^2) as follows:

$$S_k = \frac{1}{||\mu_k - \frac{\Sigma_k \mu_k}{K}||_2 + ||\sigma_k^2 - \frac{\Sigma_k \sigma_k^2}{K}||_2} \tag{2}$$

$$\mu_p = \Sigma_k \frac{S_k}{\Sigma_k S_k} \mathcal{M}[0][k], \;\; \sigma_p^2 = \Sigma_k \frac{S_k}{\Sigma_k S_k} \mathcal{M}[1][k] \tag{3}$$

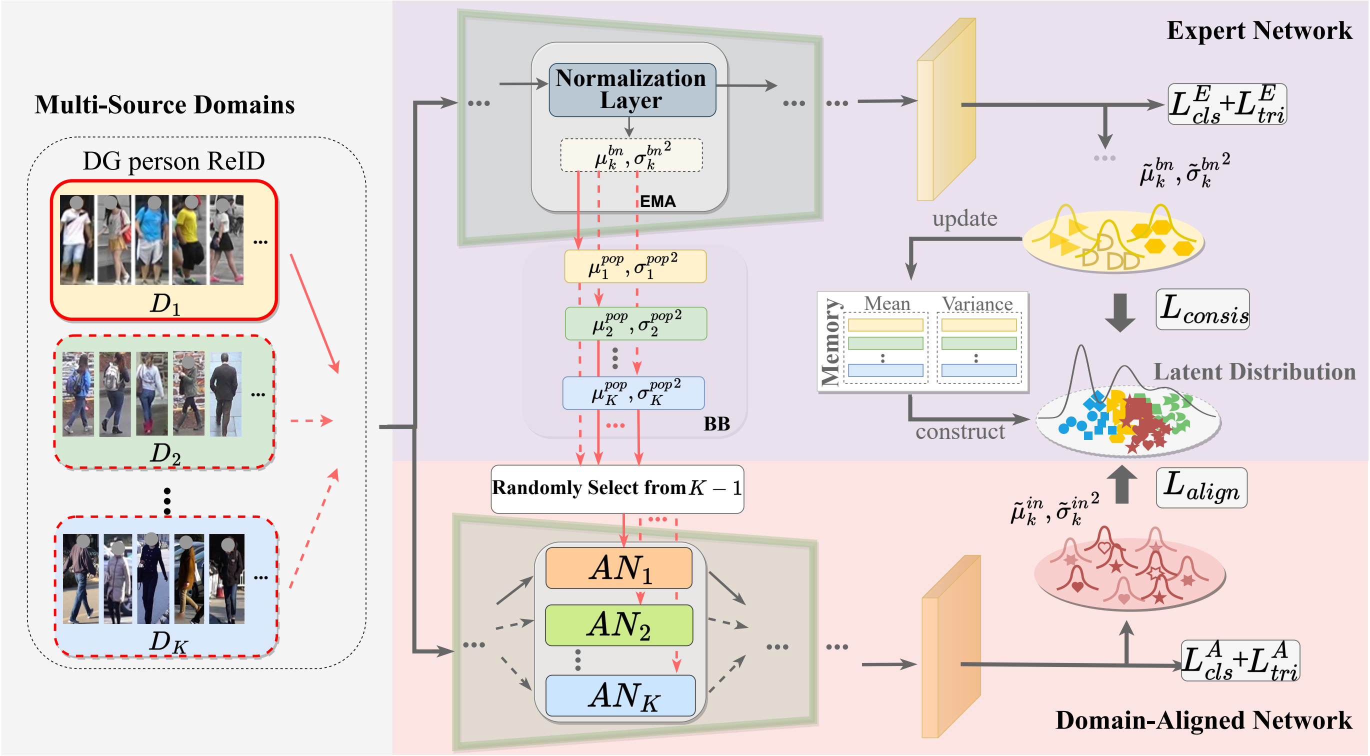

Fig. 2. Overview of our proposed LDA framework. The expert network, equipped with Buffer Bank (BB) to store domain-specific representation information. The domain-aligned network uses a novel Anti-Normalization (AN) layer to transfer each target-domain distribution to the randomly selected source domain distribution. The network dynamically constructs a latent distribution and aligns the BN statistics of source domains and the IN statistics of the mimical target domain with the latent distribution, respectively.

the memory $\mathcal{M}$ will first be averaged channel-wise to obtain the average mean $\mu_k \in \mathbb{R}^1$ and variance $\sigma_k^2 \in \mathbb{R}^1$ for each domain, where $k \in \{1, 2, ..., K\}$. Thus, the distribution of each source domain can be regarded as a two-dimensional coordinate point, forming a K-sided shape. Then, we obtain the weights S_k for each source domain by taking the inverse proportion of Euclidean distance from each point (μ_k, σ_k^2) to the center $(\frac{\Sigma_k \mu_k}{K}, \frac{\Sigma_k \sigma_k^2}{K})$.

After constructing the latent distribution, our goal is to align the BN statistics from the expert network of each source domains with this dynamic latent distribution, as illustrated in Figure 2. Thus, we use a mean squared error loss function to ensure the consistency between the distribution of each source domain and the latent space. Through continuous iterations, the distributions of the source domains will gradually become more concentrated, resulting in more domain-invariant features. This is expressed as:

$$L_{consis} = ||\tilde{\mu}_k^{bn} - \mu_p||_2^2 + ||\tilde{\sigma}_k^{bn^2} - \sigma_p^2||_2^2 \tag{4}$$

Buffer Bank. In DG ReID, for data from different source domains, previous works mostly utilized only one buffer (*i.e.*, μ^{pop} , σ^{pop^2}) for each BN layers to store the BN information from all source domains. However, they overlooked how to effectively utilize the BN information, which contains rich domain-specific information. Therefore, as illustrated in Figure 2, for each normalization layer, we design a Buffer Bank (BB) module to dynamically store the domain-specific representation information for each source domain. The formula is as follows:

$$\begin{aligned} \mu_k^{\text{pop}} &\leftarrow \lambda \mu_k^{\text{pop}} + (1-\lambda)\mu_k^{bn} \\ \sigma_k^{pop^2} &\leftarrow \lambda \sigma_k^{pop^2} + (1-\lambda)\sigma_k^{bn^2} \end{aligned} \tag{5}$$

here, $\mu_k^{bn}, \sigma_k^{bn^2} \in \mathbb{R}^C$ denote the BN statistics parameters of the current layer input feature map $X_k \in \mathbb{R}^{N \times C \times H \times W}$. μ_k^{pop} and $\sigma_k^{pop^2}$ denote the stored mean and variance. $\lambda \in [0, 1]$ is the ratio used for BB updating. It is worth noting that our expert network does not involve any additional learnable parameters.

C. Domain-Aligned Network

Building on the latent distribution and the domain-specific representation information for each source domain, a Domain-Aligned network is further proposed to solve the absence of the target domain and domain shifts. The goal of this network is to enable the model to transfer the existing source domain information to an unknown target domain. As the target domain is unseen, with the assistance of domain-specific representation information from the Buffer Bank, we can consider one source domain as a mimical target domain for the others and explicitly align the output Instance Normalization (IN) statistics with the latent distribution, leading to the alignment between the source and target domains. In this section, we will explain how to construct the mimical target domain and achieve alignment.

Anti-Normalization. In the field of style transfer, it has been demonstrated that by modifying the learnable scaling and shifting parameters of the IN layers, one can transform input images into different styles. Therefore, we consider the parameters in BB as domain-style information corresponding to the source domains. Specifically, we treat the input source domain of the domain-aligned network as an unseen target domain and map it to another chosen source domain at each normalization layer, effectively achieving domain transfer.

In each forward propagation, we replace all normalization layers in the expert network with AN. AN normalizes the activations using the statistical parameters computed over each sample (*i.e.*, $\mu_k^{in}, \sigma_k^{in^2}$) and transfer the distribution to a random one among the remaining source domains stored in BB. This iterative process enables the network to get the capability of generalizing existing learned information to unseen target domains. The formula of AN is as follows:

$$
\begin{aligned}
&AN_k(X_k[h,w]) \\
&= \frac{X_k[h,w]-\mu_k^{in}}{\sqrt{\sigma_k^{in^2}+\epsilon}}(\sqrt{\sigma_j^{pop^2}+\epsilon}) + \mu_j^{\text{pop}}, j \sim \{1,2,...,K\}\backslash\{k\}
\end{aligned}
\tag{6}
$$

for AN_k in the domain-aligned network, we use the k_{th} source domain as the mimical target domain, and transfer the distribution of each normalization layer to a source domain, which are randomly selected among the remaining, as shown by the solid line in Figure 2. It's worth mentioning that AN also has no learnable parameters.

Domain Alignment. During training, the distribution of each mimicked target sample is transferred to one of the remaining source domains. As each sample has its specific distribution across the channel-wise, we cannot directly align the distribution to the latent distribution as in the expert network. Thus, we constrain the statistical parameters of each sample in a mini-batch to the latent distribution using the following loss function:

$$
L_{align} = \frac{1}{N_b}\sum_{i=1}^{N_b} KL(\tilde{\mu}_k^{in}[i]||\mu_p) + KL(\tilde{\sigma}_k^{in^2}[i]||\sigma_p^2) \tag{7}
$$

where $KL(\cdot||\cdot)$ denotes the Kullback-Leibler divergence distance, and $\tilde{\mu}_k^{in}, \tilde{\sigma}_k^{in^2} \in \mathbb{R}^{N\times C}$ are the statistical parameters of instance normalization (IN) of the last convolutional feature map in the domain-aligned network.

Since the real target domain is unavailable to the remaining source domains, the domain-aligned network learns to align the distribution of the real target domain with the latent one.

D. Training and Inference

In the training phase, each mini-batch is collected from a randomly selected domain among K source domains. The mini-batch will first be input to the expert network for updating BB and the memory-based module so as to construct the latent distribution. Then we replace the normalization layers in expert network with AN to construct the domain-aligned network. After that, we randomly select one from the remaining $(K-1)$ source domains for all AN and input the mini-batch back into the domain-aligned network. Combing these two networks mentioned above, we have the following total loss:

$$
L_{total} = L_{cls}^{E} + L_{tri}^{E} + L_{cls}^{A} + L_{tri}^{A} + L_{consis} + L_{align} \tag{8}
$$

the cross-entropy loss (*i.e.*, L_{cls}) and triplet loss (*i.e.*, L_{tri}) are two commonly used losses in the preson re-identification community. E represents the expert network, and A represents the domain-aligned network.

During testing, we only feed the target domain to the expert network K times each. During the kth forward pass, the batch normalization operation utilizes the means and variances saved in the BB corresponding to the kth source domain(*i.e.*, μ_k^{pop}, $\sigma_k^{pop^2}$) for normalization. Subsequently, we compute the average of the K features obtained from the expert network to obtain the final inference result.

III. Experiment

A. Datasets and Settings

DG person ReID. We conduct experiments on 5 large-scale datasets: Market1501 (M), MSMT17 (MS), CUHK02 (C2), CUHK03 (C3), and CUHK-SYSU (CS), and 4 small-scale datasets: PRID, GRID, VIPeR, and iLIDs. We set three evaluation protocols following [10]. For Protocol-1, all images including training and testing sets in the large-scale datasets are used for training, and the four small-scale datasets are used for testing. The final result of each dataset is evaluated on the average of 10 repeated random splits of gallery and query sets. For Protocol-2 and Protocol-3, we follow the leave-one-out setting in [10]. Under Protocol-2, only the training sets are used for training, while all images are exploited under Protocol-3.

Implementation Details. We adopt ResNet50 IBN-a [18] pretrained on ImageNet as our backbone. Referring to [1], we set the stride size of the last residual layer as 1. Each image is resized to 256 × 128. The size of the mini-batch is 64, including 16 identities and 4 images per identity. We use random horizontal flipping, random cropping, zero padding, color jittering, and auto-augmentation [2] as data augmentation. We adopt a warm-up strategy in the first 10 epochs. The learning rate is initialized as 3.5×10^{-4} and divided by 10 at the 30th, 60th and, 90th epochs, respectively, in all the 120 epochs. The memory updating ratio ω is set to 0.2 and λ is set to 0.9. The margin a in L_{tri} is set to 0.35. All experiments are conducted with Pytorch on a single RTX 3090 GPU.

B. Comparison with SOTA Methods

Comparison under the Protocol-1. As shown in Table I, we compare our methods with other SOTAs under Protocol-1. Since DukeMTMC-reID has been retracted, we only use other large-scale datasets for training. From the results, we

TABLE I

Comparison with the SOTAs in DG ReID methods under the setting of Protocol-1. "∗" indicates the re-implementation for this method based on the authors' code on Github. "–" indicates the value is not provided in the corresponding paper. The best and the second best results are highlighted in **bold** and underline, respectively.

Method	Reference	Source domain	Target domain								**Average**	
			PRID		GRID		VIPeR		iLIDs			
			mAP	Rank-1	mAP	Rank-1	mAP	Rank-1	mAP	Rank-1	mAP	Rank-1
SNR [9]	CVPR'20	M+D+C2 +C3+CS	66.5	52.1	47.7	40.2	61.3	52.9	89.9	84.1	66.4	57.3
RaMoE [8]	CVPR'21		67.3	57.7	54.2	46.8	64.6	56.6	**90.2**	**85.0**	62.0	61.5
MDA [12]	CVPR'22		–	–	62.9	**61.2**	71.7	63.5	84.4	80.4	–	–
$* QAConv_{50}$ [17]	ECCV'20	M+C2 +C3+CS	62.2	52.3	57.4	48.6	66.3	57.0	81.9	75.0	67.0	58.2
$* M^3L$ [6]	CVPR'21		65.3	55.0	50.5	40.0	68.2	60.8	74.3	65.0	64.6	55.2
∗ MetaBIN [7]	CVPR'21		70.8	61.2	57.9	50.2	64.3	55.9	82.7	74.7	68.9	60.5
META [10]	ECCV'22		71.7	61.9	60.1	52.4	68.4	61.5	83.5	79.2	70.9	63.8
ACL [13]	ECCV'22		73.4	63.0	**65.7**	55.2	**75.1**	**66.4**	86.5	81.8	**75.2**	66.6
LDA (Ours)	This paper		**76.9**	**69.5**	63.3	53.8	73.7	65.2	84.3	79.0	74.7	**67.1**

TABLE II

Comparison with the SOTAs in DG ReID methods under the setting of Protocol-2 (only training sets) and Protocol-3 (including training and testing sets). "∗" indicates the re-implementation for this method based on the authors' code on Github. The best and the second best results are highlighted in **bold** and underline, respectively.

Method	Reference	Setting	M+MS+CS→ C3		M+CS+C3 → MS		MS+CS+C3 → M		**Average**	
			mAP	Rank-1	mAP	Rank-1	mAP	Rank-1	mAP	Rank-1
∗ SNR [9]	CVPR'20	Protocol-2	8.9	8.9	6.8	19.9	34.6	62.7	16.8	30.5
$* QAConv_{50}$ [17]	ECCV'20		25.4	24.8	16.4	45.3	63.1	83.7	35.0	51.3
$* M^3L$ [6]	CVPR'21		20.9	31.9	15.9	36.9	58.4	79.9	31.7	49.6
∗ MetaBIN [7]	CVPR'21		28.8	28.1	17.8	40.2	57.9	80.1	34.8	49.5
META [10]	ECCV'22		36.3	35.1	22.5	49.9	67.5	86.1	42.1	57.0
ACL [13]	ECCV'22		41.2	41.8	20.4	45.9	**74.3**	**89.3**	45.3	59.0
LDA (Ours)	This paper		**42.5**	**43.3**	**23.9**	**51.1**	70.0	86.9	**45.5**	**60.4**
∗ SNR [9]	CVPR'20	Protocol-3	17.5	17.1	7.7	22.0	52.4	77.8	25.9	39.0
$* QAConv_{50}$ [17]	ECCV'20		32.9	33.3	17.6	46.6	66.5	85.0	39.0	55.0
$* M^3L$ [6]	CVPR'21		32.3	33.8	16.2	36.9	61.2	81.2	36.6	50.6
∗ MetaBIN [7]	CVPR'21		43.0	43.1	18.8	41.2	67.2	84.5	43.0	56.3
META [10]	ECCV'22		47.1	46.2	24.4	52.1	76.5	90.5	49.3	62.9
ACL [13]	ECCV'22		**49.4**	**50.1**	21.7	47.3	**76.8**	**90.6**	49.3	62.7
LDA (Ours)	This paper		48.4	49.4	**25.9**	**53.6**	74.8	89.6	**49.7**	**64.2**

can find that our method achieves performance comparable to the SOTAs. Particularly on PRID, our method achieves the SOTAs, with 76.9% mAP and 69.5% Rank-1, creating a significant gap from the second-best approach.

Comparison under the Protocol-2 and Protocol-3. In order to evaluate our method on large-scale datasets, we compare our methods with SOTAs under Protocol-2 and Protocol-3, as shown in Table II. The results show that our method achieves SOTA average metrics under both Protocol-2 and Protocol-3. Specifically, when MS is used as the target domain, our method surpasses the second-best approach by 1.4% mAP, 1.2% Rank-1 and 1.5% mAP, 1.5% Rank-1 under Protocol-2 and Protocol-3, respectively. This demonstrates our LDA can alleviate the domain shifts between these large-scale datasets and achieve better generalization ability.

Remarkably, compared to the SOTA method in ACL, our approach designs a much simpler architecture and more relaxing training conditions. For instance, under Protocol-2, ACL requires a minimum of four 2080Ti GPUs, which translates to 44GB of VRAM, with an average training time of 44 hours for a training. In contrast, our approach only requires a single 3090 GPU and utilizes only 15GB of VRAM, with an average training time of 22 hours.

TABLE III

Ablation study on the effectiveness of main components of our method. The experiment is under Protocol-1. We calculated the average metrics on four datasets. The best results are highlighted in **bold**.

Method	L_{consis}	L_{align}	Protocol-1 (Average)	
			mAP	Rank-1
Expert Network Only			72.1	63.7
	✓		72.3	64.1
LDA(ours)			73.2	64.7
		✓	73.5	65.3
	✓		73.8	66.0
	✓	✓	**74.7**	**67.1**

C. Ablation Study

Effectiveness of main components of our method. In order to verify the effectiveness of the proposed two networks. We conduct ablation studies in Table III. From the first and third rows, after combining the expert network and the domain-aligned network without any loss functions, it improves the average performance with an increase of 1.1% in mAP and 1.0% in Rank-1. It is evident that introducing an additional loss function to constrain the latent distribution can enhance the network's generalization performance.

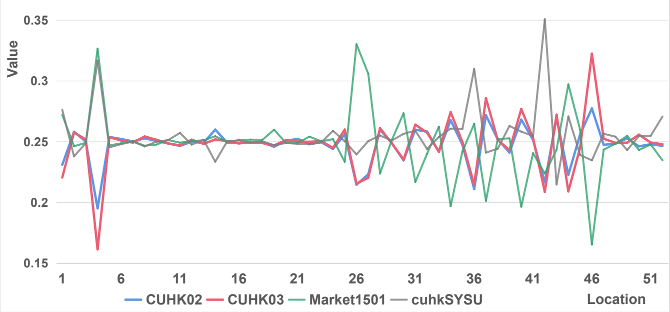

Fig. 3. Visualization of the Buffer Bank under Protocol-1.

Specifically, from the third and last rows, after the inclusion of two additional losses, the average performance reaches 74.7% mAP and 67.1% Rank-1.

Visualization of the Buffer Bank. We visualize the domain-specific information in BB under Protocol-1. As shown in Figure 3 , we sum the means of different source domains in each layer of BB and apply a softmax operation, approximating the domain-specific information within. The horizontal axis represents the position of BB at different normalization layers in the network, and the vertical axis represents the corresponding proportions. It is evident that the curves for CUHK02 and CUHK03, which are from the same scene, are highly similar. The differences among other datasets are quite apparent, especially in deeper layers of the network. This observation indicates that our BB retains rich domain-specific representation information.

IV. Conclusion

In this paper, we propose a novel Latent Distribution Alignment (LDA) framework that explicitly uses the statistical parameters of various normalizations for implicitly aligning the distributions of source domains with the target ones without any added learnable parameters. To this end, we design a domain-aligned network embedded with proposed Anti-Normalization (AN), which further helps the model generalize to unseen domains. Extensive experiments demonstrate the simplicity and effectiveness of our LDA framework.

References

[1] 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,” *IEEE Transactions on Multimedia*, vol. 22, no. 10, pp. 2597–2609, 2019.

[2] L. He, X. Liao, W. Liu, X. Liu, P. Cheng, and T. Mei, “Fastreid: A pytorch toolbox for general instance re-identification,” in *Proceedings of the 31st ACM International Conference on Multimedia*, 2023, pp. 9664–9667.

[3] L. Wei, S. Zhang, W. Gao, and Q. Tian, “Person transfer gan to bridge domain gap for person re-identification,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 79–88.

[4] J. Liu, Z.-J. Zha, D. Chen, R. Hong, and M. Wang, “Adaptive transfer network for cross-domain person re-identification,” in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 7202–7211.

[5] K. Jiang, T. Zhang, Y. Zhang, F. Wu, and Y. Rui, “Self-supervised agent learning for unsupervised cross-domain person re-identification,” *IEEE Transactions on Image Processing*, vol. 29, pp. 8549–8560, 2020.

[6] Y. Zhao, Z. Zhong, F. Yang, Z. Luo, Y. Lin, S. Li, and N. Sebe, “Learning to generalize unseen domains via memory-based multi-source meta-learning for person re-identification,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 6277–6286.

[7] 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*, 2021, pp. 3425–3435.

[8] Y. Dai, X. Li, J. Liu, Z. Tong, and L.-Y. Duan, “Generalizable person re-identification with relevance-aware mixture of experts,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 16 145–16 154.

[9] X. Jin, C. Lan, W. Zeng, Z. Chen, and L. Zhang, “Style normalization and restitution for generalizable person re-identification,” in *proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 3143–3152.

[10] B. Xu, J. Liang, L. He, and Z. Sun, “Mimic embedding via adaptive aggregation: Learning generalizable person re-identification,” in *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XIV*. Springer, 2022, pp. 372–388.

[11] Y. Bai, J. Jiao, W. Ce, J. Liu, Y. Lou, X. Feng, and L.-Y. Duan, “Person30k: A dual-meta generalization network for person re-identification,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 2123–2132.

[12] H. Ni, J. Song, X. Luo, F. Zheng, W. Li, and H. T. Shen, “Meta distribution alignment for generalizable person re-identification,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 2487–2496.

[13] P. Zhang, H. Dou, Y. Yu, and X. Li, “Adaptive cross-domain learning for generalizable person re-identification,” in *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XIV*. Springer, 2022, pp. 215–232.

[14] K. Han, C. Si, Y. Huang, L. Wang, and T. Tan, “Generalizable person re-identification via self-supervised batch norm test-time adaption,” in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 36, no. 1, 2022, pp. 817–825.

[15] X. Huang and S. Belongie, “Arbitrary style transfer in real-time with adaptive instance normalization,” in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 1501–1510.

[16] H. Nam and H.-E. Kim, “Batch-instance normalization for adaptively style-invariant neural networks,” *Advances in Neural Information Processing Systems*, vol. 31, 2018.

[17] S. Liao and L. Shao, “Interpretable and generalizable person re-identification with query-adaptive convolution and temporal lifting,” in *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XI 16*. Springer, 2020, pp. 456–474.

[18] X. Pan, P. Luo, J. Shi, and X. Tang, “Two at once: Enhancing learning and generalization capacities via ibn-net,” in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 464–479.