Style Variable and Irrelevant Learning for Generalizable Person Re-identification

KAI LV, Beijing Key Laboratory of Traffic Data Analysis and Mining, School of Computer and Information Technology, Beijing Jiaotong University, China
HAOBO CHEN and CHUYANG ZHAO, Beihang University, China
KAI TU, JUNRU CHEN, YADONG LI, and BOXUN LI, MEGVII Technology, China
YOUFANG LIN*, Beijing Key Laboratory of Traffic Data Analysis and Mining, School of Computer and Information Technology, Beijing Jiaotong University, China

Domain Generalization person Re-identification (DG-ReID) has gained much attention recently due to the poor performance of supervised re-identification on unseen domains. The goal of domain generalization is to develop a model that is insensitive to domain bias and can perform well across different domains. In this paper, we conduct experiments to verify the importance of style factors in domain bias. Specifically, the experiments are to affirm that style bias across different domains significantly contributes to domain bias. Based on this observation, we propose Style Variable and Irrelevant Learning (SVIL) to eliminate the influence of style factors on the model. Specifically, we employ a Style Jitter Module (SJM) that enhances the style diversity of a specific source domain and reduces the style differences among various source domains. This allows the model to focus on identity-relevant information and be robust to style changes. We also integrate the SJM module with a meta-learning algorithm to further enhance the model’s generalization ability. Notably, our SJM module is easy to implement and does not add any inference cost. Our extensive experiments demonstrate the effectiveness of our approach, which outperforms existing methods on DG-ReID benchmarks.

CCS Concepts: • Information systems → Information retrieval; Retrieval models and ranking.

Additional Key Words and Phrases: Re-identification, Domain Generalization, Style Learning, Style Jitter

1 INTRODUCTION

Person Re-identification (ReID) [6, 8, 51] plays a crucial role in intelligent surveillance systems as it aims to identify individuals across non-overlapping cameras under various scenes and views. The application of deep learning techniques [14, 19, 39, 40, 62] has significantly improved the performance of supervised ReID models, where the model is trained and tested on the same dataset [4, 16, 37, 48, 54, 61, 63, 65, 75]. However, these models suffer from a severe performance drop when applied to an unseen domain due to domain bias. To address this
issue, research on unsupervised domain adaptation and Domain Generalization ReID (DG-ReID) has gained considerable attention in recent years. In the unsupervised domain adaptation task, images from the target domain are available, but the labels are unknown. Approaches such as [10, 12, 25, 53] adapt the source-trained model and fine-tune it using unlabeled target data. On the other hand, in DG-ReID, both the images and labels are invisible, which makes the task more challenging but also more practical.

Existing DG-ReID methods can be categorized into three main approaches, i.e., adversarial learning-based methods, normalization-based methods, and meta-learning-based methods. Adversarial learning-based methods [35, 55] employ the adversarial auto-encoder module and discrimination task to extract domain-invariant features. However, they may suffer from unbalanced training between multiple tasks. Normalization-based methods [3, 23, 28] leverage instance normalization to separate identity-relevant and identity-irrelevant features. However, these methods typically focus on individual samples and do not consider the relationship between multiple samples, both intra-domain and inter-domain. Meta-learning-based methods [2, 9, 27] partition the training dataset into meta-train and meta-test subsets to simulate domain bias and optimize the model through "learning-to-learn". While the goal of meta-learning is to prevent overfitting, as training progresses, the model may eventually overfit as it encounters enough data, including that in the meta-test subset. Most of the DG-ReID methods mentioned above are based on conventional DG tasks that focus on classification tasks and differ from DG-ReID tasks. Typically, various datasets are defined as different domains in conventional DG tasks. These datasets may include real image and painting picture datasets, which have obvious domain bias. However, in DG-ReID tasks, only real image datasets are used, and the images from different datasets (domains) are captured from distinct cameras and regions.

This paper investigates the factors that contribute to domain bias in DG-ReID and aims to address this issue. Our initial hypothesis is that domain bias in DG-ReID may be caused by differences in scene, season, camera angle, and style, such as illumination and hue. To verify the impact of style factors in domain bias, we conduct an exploratory experiment as shown in Fig. 1. We replace the mean and variance statistics of images in the source domain (Market 1501 [72]) with those from the target domain (DukeMTMC-reID [49, 74]), which are considered as style representations in transfer learning tasks [20, 36, 57]. The results in Fig. 1 (b) demonstrate that training
the model on the source domain with the same style as the target domain leads to better ReID performance, confirming that style factors are a significant contributor to domain bias. However, in DG-ReID, we cannot access the style information of the target domain. Hence, we need to eliminate the influence of style factors to reduce domain bias in DG-ReID.

Motivated by our observations, we propose a Style Variable and Irrelevant Learning (SVIL) method for DG-ReID. Unlike multi-source DG classification tasks, DG-ReID has different label spaces across various source domains, indicating that each identity only exists in its specific domain, and identity and domain are strongly related. As illustrated in Fig. 1 (c), the identity (the cycle one) can be easily distinguished by relying on domain information (D1). This highlights the existence of style bias among distinct identities, wherein models often excessively rely on stylistic cues rather than recognizing identity-related details. To address this issue, SVIL enhances the style diversity of the specific source domain and reduces the style differences of different source domains, enabling the model to focus on identity-relevant information and be insensitive to style variations. Specifically, we design a Style Jitter Module (SJM) that leverages style memory to store identity style statistics in all source domains. For each identity in the mini-batch, we generate a new style statistic based on all the stored statistics, where the new style is a weighted combination of the negative sample styles in all domains, based on identity relationship modeling. The original style is replaced by the new style to remove the relationship between the original style and identity. To better distinguish hard negative samples, we use a hard identity emphasis strategy to strengthen the importance of the styles of hard samples during style generation. Additionally, we perform cross-domain identity emphasis to avoid the information of identities within the same domain dominating the generated styles. Our SJM module is an inference cost-free module without learnable parameters. It results in variable styles between training samples, making negative samples harder.

In addition, we notice that previous methods [5, 23, 71] employ a unified loss function on multiple domains to learn feature representations. They omit the strong relationship between identity and domain caused by various label spaces. To weaken the relationship, we propose domain-agnostic loss and domain-specific loss to jointly learn robust features. A meta collaborative training procedure is employed as well to maximize the effectiveness of our SJM module, where the meta-learning algorithm is organically combined. The SJM module can enlarge and continuously change the domain bias between meta-train and meta-test, successfully delaying the model from overfitting. The combination of these two sub-method produces a synergy effect, leading to a style-irrelevant model to generalize to the unseen target domain.

We also perform the proposed SVIL on the single-source DG-ReID task to confirm its effectiveness. Specifically, because images captured under different cameras have stylistic differences, we divide a single dataset into multiple sub-datasets according to cameras. Then we employ SVIL on the single-source DG-ReID task to learn the style-insensitive model as we do on the multi-source DG-ReID task. In summary, the major contributions of our works are summarized as follows:

- Identifying the influence of style factors in domain generation re-identification (DG-ReID) through observation and experiments. We then introduce the Style Variable and Irrelevant Learning (SVIL) method as a solution. SVIL, comprising the Style Jitter Module (SJM) and meta collaborative training, effectively mitigates the impact of style variations.
- Proposing Style Jitter Module (SJM) to diversify the styles attributed to identities, thereby increasing the variety of feature styles. By doing so, the model is forced to focus on identity-relevant information while ignoring style-related factors.
- Applying a meta-collaborative training procedure to further reduce the influence of style variations. This procedure synergizes with SJM, and can further improve the effectiveness of our approach.
- Validating the effectiveness of our proposed approach by demonstrating its superior performance in both multi-source and single-source DG-ReID tasks compared to existing state-of-the-art methods.
The rest of this paper is organized as follows. Section 2 reviews and discusses the related works. Section 3 first introduces the overall framework of the proposed method, and then elaborates the designed style jitter module and the training procedure. Section 4 presents the experimental results. Finally, Section 5 draws the conclusion.

2 RELATED WORKS

2.1 Re-identification

Re-identification (ReID) tasks have gained significant attention due to their wide application in urban surveillance and intelligent transportation [24, 39, 41, 68]. Both person ReID and vehicle ReID have achieved remarkable performance through supervised learning approaches. For example, Sun et al. [54] proposed a circle loss function for effective deep feature learning. Chu et al. [4] introduced a novel viewpoint-aware metric learning approach tailored for vehicle Re-ID. Similarly, He et al. [16] designed a pure transformer framework for ReID tasks, achieving notable results. Other attempts like [37, 48, 61, 65] have also shown great performance using supervised learning.

In Lifelong Person Re-Identification (LReID), a variety of innovative methods have been proposed to tackle the challenges of domain-incremental learning and generalization across both seen and unseen domains. In [45], Meta Reconciliation Normalization (MRN) is a novel normalization technique that optimizes the balance between domain-dependent and independent statistics, enhancing LReID performance through a meta-learning framework without the need for data replay. In [44], the authors emphasized the AKA framework, focusing on continuous learning and generalization across multiple domains, with significant improvements demonstrated over competitors in generalizing evaluation. Similarly, Pu et al. [46] presented a framework that integrates Adaptive Knowledge Accumulation (AKA) and differentiable Ranking Consistency Distillation (RCD) to mitigate catastrophic forgetting while boosting generalization capabilities. Lastly, Jin et al. [22] explored Cloth-Changing ReID (CC-ReID), employing gait recognition to develop cloth-agnostic identity matches, proposing a dual-stream architecture that enhances the ReID model’s robustness against clothing changes. Together, these studies highlight advancements in addressing both the static and dynamic challenges within the field of ReID.

Despite their successes, these supervised learning methods have paid little attention to exploring invariant information across domains, thereby limiting their effectiveness in practical applications. This has led to the emergence of Unsupervised Domain Adaptation (UDA) methods [10, 12, 25, 53] aimed at addressing the differences between the source domain and target domain. For instance, Ge et al. [12] proposed an MMT framework to handle noise labels in the clustering process. Song et al. [53] employed a self-training scheme to iteratively minimize the loss functions. However, these UDA methods require a sufficient amount of unlabeled training data from the target domain, making them impractical in many real-world scenarios.

To overcome this limitation, the domain generalization task has been established, wherein models are trained to generalize to unseen domains without relying on target domain data.

2.2 Domain Generalization

Domain Generalization (DG) has received increasing attention in the research community [42, 57, 79]. These methods generally assume that the source and target domains have the same label space. However, in DG-ReID, the label spaces are non-overlapping, making conventional DG methods unsuitable for direct application.

Adversarial learning has emerged as a popular method for DG-ReID [35, 55]. For instance, Lin et al. [35] proposed a multi-dataset feature generalization network to learn a universal domain-invariant feature representation and generalize it to unseen camera systems. Normalization-based methods are also mainstream solutions in DG-ReID. For example, Jin et al. [23] designed a Style Normalization and Restitution (SNR) module to enhance the generalization capabilities of networks. Choi et al. [3] proposed the MetaBIN framework, which prevents models from overfitting to given source styles and improves generalization capability to unseen domains.
In addition, Model-Agnostic Meta-Learning (MAML) [9] has become popular for DG-ReID. Zhao et al. [71] proposed a memory-based framework to solve DG-ReID, employing the meta-learning algorithm. Similarly, Dai et al. [5] employed a vote experts approach under the meta-learning framework. In our method, the SJM module leverages the strengths of meta-learning, producing a synergy effect.

2.3 Style Transfer

In the field of style transfer, it has been shown that the statistical information of features extracted by deep neural networks can represent the style of an image [11, 26, 32]. Gatys et al. [11] used second-order statistical features as the optimization goal to achieve style transfer. Li et al. [32] found that other statistics, such as mean and variance, of Batch Normalization (BN) layers contain the traits of different domains. Ulyanov et al. [58] utilized Instance Normalization (IN) to obtain the statistical characteristics of features. Dumoulin et al. [7] proposed Conditional Instance Normalization (CIN), which learns pairs of \( \alpha \) and \( \beta \) parameters during training. When both the image content and decoding network are the same, adopting different pairs of \( \alpha \) and \( \beta \) will yield different styles of transfer results. In IN and CIN, the network can learn the affine transformation parameters \( \alpha \) and \( \beta \). Additionally, to remove the two learnable parameters, AdaIN [20] was proposed. It directly replaces the two parameters with the mean and standard deviation of the style feature. Tan et al. [56] proposed a style interleaved learning framework that uses dual forward propagations and a single backward propagation per iteration to prevent overfitting to source domain styles. Li et al. [31] introduced a novel training strategy combining style-aware hard-negative sampling with dynamic style mixing to improve metric learning and increase style diversity, significantly boosting performance in domain generalizable person re-identification tasks. In our approach, SVIL enhances the model’s generalization capabilities by effectively and efficiently mitigating the impact of variations in domain styles.

Overall, the proposed SVIL approach aims to tackle the challenges of DG-ReID by effectively exploring invariant information across domains and leveraging the strengths of meta-learning and style transfer for improved performance.

3 METHOD

The purpose of our method is to narrow the domain gap caused by style variations in different domains. By increasing the style diversity in the training stage, the network will be insensitive to style changes between different domains and pay more attention to identity-relevant information. In this way, we can obtain a more generalizable model that can be deployed directly to a new unseen domain.

3.1 Overview

The main pipeline of our method is illustrated in Fig. 2 (a). In the train stage, we can access \( K \) source domains \( \mathcal{D} = \{ \mathcal{D}_k \}_{k=1}^{K} \), where \( \mathcal{D}_k = \{ (x^k_i, y^k_i) \}_{i=1}^{N_k} \) and \( (x^k_i, y^k_i) \) is a image-label pairs belong to \( \mathcal{D}_k \). \( N_k \) is the number of images in the source domain \( \mathcal{D}_k \). For each sample \( x^k_i \), its label \( y^k_i \) comes from the specific label space \( \mathcal{Y}_k \). By combining all source domains (termed global domain), we can obtain its global label \( y^g_i \in \mathcal{Y} \), where \( \mathcal{Y} \) is the global label space. We sample \( B \) images \( x^k_i \) from each domain \( \mathcal{D}_k \), where \( B = P \cdot K \), meaning \( K \) images from each of \( P \) person identities. Then all images form a mini-batch of size \( B \cdot K \) for all domains (\( P \cdot K \) identities in total). They are fed into the backbone network \( f(\cdot) \) (e.g., ResNet-50). After the Global Average Pooling (GAP), we design \( K \) domain-specific classifiers \( \phi_k(\cdot) \) and a domain agnostic classifier \( \phi_a(\cdot) \) to learn the feature representations.

We plug the SJM module between stage-1 and stage-2. During the training phase, half of the images for each identity pass through the SJM module, resulting in a modification of the style information in the feature representations at stage-1. Meanwhile, the style of the other half of the images remains unchanged. This process leads to a transformation of the style information within different feature representations. Subsequently, all of
these modified features are forwarded through stage-2 and propagated through the network until reaching the final layers. We incorporate the domain-specific loss $L_{\text{spec}}$ on the features and predictions specific to each domain. To encourage the model to become less sensitive to domain-specific variations, we employ a domain-agnostic loss $L_{\text{agno}}$ on the features and predictions across all domains. Two losses jointly prompt the network to ignore the style variation of features and focus on identity-relevant information.

Furthermore, during the training process, we apply a meta-learning algorithm in conjunction with the proposed SJM module, resulting in a synergistic effect that enhances the overall performance of our method (further details in Section 3.4). It is worth noting that our SJM module is only used in training and will be discarded in testing.

### 3.2 Style Jitter Module

Images of different domains in DG-ReID are generally captured from different cameras and scenes, leading to various styles. To address the style variations in DG-ReID, we propose the SJM module to decrease the model’s sensitivity to these style variations. The SJM module takes a hidden representation from a specific domain as input and produces a stylized representation, in which the cross-domain style information is transferred.

Specifically, we denote backbone network as $f(x) = f_m(g_m(x))$, where $g_m$ is to map the input data $x$ to the hidden representation $F = g_m(x) \in \mathbb{R}^{H \times W \times C}$ after the $m$-th stage (e.g., $m = 1$ in Fig. 2) and $f_m$ denotes the
network that maps the feature $g_m(x)$ to the feature vector after the GAP layer. In a mini-batch including $P \cdot K$ identities and each identity including $K'$ images, we can obtain their maps at $m$-th hidden stage. For each identity, we select $\frac{K'}{2}$ representations $F$ and diverse them by the SJM module. As shown in Fig. 2 (b), we first compute the channel-wise mean and standard deviation $\mu(F), \sigma(F) \in \mathbb{R}^C$ by the style operation as follows:

$$
\mu(F) = \frac{1}{HW} \sum_{h=1}^{H} \sum_{w=1}^{W} F_{hw},
$$

$$
\sigma(F) = \sqrt{\frac{1}{HW} \sum_{h=1}^{H} \sum_{w=1}^{W} (F_{hw} - \mu(F))^2 + \epsilon}.
$$

These feature statistics can capture informative characteristics of the specific domain and be viewed as style representation according to previous works [20]. Then, we construct a identity style memory to store the style representations of each identity, where the cross-domain style representations are generated.

Our proposed method is used exclusively during the training phase to improve the learning process and boost the model’s generalization capabilities. Once training is complete, these components are not required for deployment, ensuring that our method does not introduce any additional computational overhead or complexity during inference.

3.2.1 **Identity Style Memory Construction.** The style memory maintains the style representation of all identities. For $|\mathcal{Y}_6|$ identities from all source domains, the memory $\hat{M}$ has $|\mathcal{Y}_6|$ slots, where each slot saves the style features of the corresponding identity. In initialization, we utilize the model to extract style representations for all source samples. Then we initialize the representation $\hat{M}[[j]] = (\hat{\mu}[[j]], \hat{\sigma}[[j]])$ in memory using the average of all representations for $j$-th identity. At each training iteration, we update the memory with the style representations in the current mini-batch, which is formulated as:

$$
\hat{\mu}[[j]] \leftarrow m \cdot \hat{\mu}[[j]] + (1 - m) \cdot \frac{2}{|B_j|} \sum_{x_i \in B_j} \mu(g_m(x_i)),
$$

$$
\hat{\sigma}[[j]] \leftarrow m \cdot \hat{\sigma}[[j]] + (1 - m) \cdot \frac{2}{|B_j|} \sum_{x_i \in B_j} \sigma(g_m(x_i)),
$$

where $g_m(x_i)$ is the input feature $F$, $B_j$ refers to the samples belonging to the $j$-th identity and $|B_j| = \mathcal{K}$ is the number of samples for the $j$-th identity in the current mini-batch. $m \in [0, 1]$ determines the update rate.

3.2.2 **Stylized Feature Generation.** To incorporate information from other domains into the current specific domain, we create new style representations by assigning weights to all identity styles except for the identities corresponding to the input features. To be more specific, we calculate an identity-related weight, denoted as $\alpha \in \mathbb{R}^{|\mathcal{Y}_6|}$, for the input feature through identity relationship modeling (further elaborated in Section 3.3). Subsequently, the new style representation is obtained by combining the features of all identities through weighted fusion as follows:

$$
\mu' = \sum_{j=1}^{|\mathcal{Y}_6|} \alpha_j \hat{\mu}[[j]],
$$

$$
\sigma' = \sum_{j=1}^{|\mathcal{Y}_6|} \alpha_j \hat{\sigma}[[j]].
$$
where $\hat{\mu}$ and $\hat{\sigma}$ are style representations in memory. Then, we replace the style representation with the synthetic one to achieve the domain-agnostic feature, which is formulated as:

$$F' = \sum_{j=1}^{\|Y_6\|} \alpha_j \hat{\sigma}[j] \left( \frac{F - \hat{\mu}(F)}{\sigma'(F)} \right) + \sum_{j=1}^{\|Y_6\|} \alpha_j \hat{\mu}[j]. \quad (7)$$

### 3.3 Identity Relationship Modeling

The identity-related weights control synthetic style representations and influence stylized features. Intuitively, for the features of a specific domain $D_k$, on the one hand, we expect the generated features to contain more style information of other domains $D_{i\neq k}$, which is conducive to learning domain-independent information. On the other hand, the synthesized features should be as difficult as possible, which is beneficial to improve the generalization ability of the model.

#### 3.3.1 Hard Identity Emphasis

We obtain difficult synthetic features by emphasizing the stylistic features of hard identities, i.e., larger identity-related weight $\alpha$. As shown in Fig. 2 (a), we employ a domain-agnostic classifier $\varphi_g(\cdot)$ to classify all features in a mini-batch, where the domain-agnostic loss is adopted as follows:

$$L_{agno} = \frac{1}{K} \sum_{k=1}^{K} \sum_{i=1}^{B} \mathcal{L}_{ce}(\varphi_g(f(x_i^k)), y_i^g) + \mathcal{L}_{tri}(f(x), y^g), \quad (8)$$

where $\mathcal{L}_{ce}$ denotes the cross-entropy loss and $\mathcal{L}_{tri}$ denotes the triplet loss [17]. We normalize the weight $W$ in $\varphi_g(\cdot)$ and the feature vectors $f(x)$, then the cross-entropy loss becomes as follow:

$$\mathcal{L}_{ce} = -\log \frac{\exp(\cos(W_{ij}, f(x^k_i))/\tau)}{\sum_{j'\in|Y_6|} \exp(\cos(W_j, f(x^k_i))/\tau)}, \quad (9)$$

in which $\cos(W_j, f(x)) = \frac{W_j^T f(x)}{|W_j||f(x)|}$ is the cosine similarity between $W_j$ and $f(x)$, $\tau$ is a temperature learnable parameter. Previous study [60] shows that the normalized weight $W_j$ can be viewed as the center of features belonging to identity $j$. Inspired by this, we use the similarity between identity centers to represent the similarity between identities, which is calculated as follows:

$$S = \frac{W \cdot W^T}{||W|| \cdot ||W||}, \quad (10)$$

where $S \in \mathbb{R}^{|Y_6| \times |Y_6|}$ and $S_{ij}$ denotes the cosine similarity between identities $i$ and $j$. Then we can calculate the identity-related weight based on identity similarities.

#### 3.3.2 Identity Similarity Memory

At each training iteration, we cannot directly obtain the identity similarities of the current iteration in the SJM module. Therefore we use the historical cumulative similarity to calculate the weight, which brings the added benefit that cumulative similarity is more representative of true identity similarity than current similarity. More concretely, we build an identity similarity memory $\hat{S}$ to store the historical similarity via momentum updates. At initialization, we set $\hat{S}[j] = (\frac{1}{|Y_6|}, \ldots, \frac{1}{|Y_6|})$ to treat all identities equally and $\hat{S}[j][j] = 0$ to avoid the influence of identity itself.

Then, for each input feature whose identity is $j$, we can obtain the soft identity-related weight based on the similarity memory as follows:

$$\alpha = \text{softmax}(\beta), \quad (11)$$

ACM Trans. Multimedia Comput. Commun. Appl.
where $\beta = \hat{S}[j]$ is the identity-related factor. The identity with high similarity to identity $j$ (i.e., the hard identity) has greater value, and its style representation is paid more attention. We can also focus only on the hardest identity by using the hard weight:

$$
\alpha_i = \begin{cases} 
1, & i = \arg\max_k (\beta) \\
0, & i \neq \arg\max_k (\beta)
\end{cases},
$$

(12)

where the hard weight $\alpha \in \mathbb{R}^{|Y|}$ is a one-hot vector.

According to the weight, we can get the generated stylized features as described in Sec 3.2. All features in the mini-batch are utilized to calculate the losses and update the network via gradient descent, including the weights, $\in_i (\cdot)$. For each identity $9$, we set $i_9 = 0$ and update the identity similarity memory as follows:

$$
\bar{\mathbf{\beta}} [9] \leftarrow \beta (\bar{\mathbf{\beta}} [9] + (1 - \beta) \cdot D_k),
$$

(13)

where $S$ is calculated by the last updated weight $W$.

### 3.3.3 Cross-domain Identity Emphasis

Although identity emphasis can make difficult identities get more attention, most of the difficult identities exist in the same domain, hindering the transfer of cross-domain information. To emphasize the cross-domain identities, we consider the influence of different domains when calculating identity-related weights, i.e., give the cross-domain identities greater weights.

More specifically, we further introduce the relation (distances) between different domains when computing the identity-related factor $\beta$. As shown in Fig. 2 (a), a feature set $A_k = \{f(x_i^k)\}_{i=1}^B$ can be obtained for each source domain $D_k$. Because the feature set $A_k$ contains only a few features and is a subset of the domain $D_k$, its distribution cannot represent the distribution of the entire domain. We model the distance between domains using the cumulative distribution distance between feature sets. Similar to identity similarity saving, we construct a domain distance memory $\mathbf{\hat{D}}$ to store the cumulative MMD distances between various feature sets. At each iteration, the distance between two sets is computed as follows:

$$
\hat{D}_{s,t} = \text{MMD}^2 (A_s, A_t) = \left\| \frac{1}{|A_s|} \sum_{i=1}^B \phi (f(x_i^s)) - \frac{1}{|A_t|} \sum_{i=1}^B \phi (f(x_i^t)) \right\|^2,
$$

(14)

where $\phi (\cdot)$ is a particular representation that maps the feature $f(x)$ into a reproducing kernel Hilbert space. The memory $\mathbf{\hat{D}}$ for $k$-th domain is updated as follows:

$$
\hat{D}[k] \leftarrow m \cdot \hat{D}[k] + (1 - m) \cdot D_k.
$$

(15)

Given the identity $j$ of domain $D_k$, we denote it as $D^{-1}(j) = k$. To enhance the cross-domain identities, the identity-related factor $\beta$ in Eq. 11 and Eq. 12 are redefined as follows:

$$
\beta_i = \hat{S}[j][i] + \hat{D}[D^{-1}(j)][D^{-1}(i)],
$$

(16)

where $\beta \in \mathbb{R}^{|Y|}$. In this way, information about difficult identities and information about cross-domain identities are simultaneously emphasized.

### 3.4 Training Procedure

To make the model focus on domain-independent identity information, we adopt two losses with different effect scopes to supervise the learning of the network. Furthermore, a Model-Agnostic Meta-Learning (MAML) algorithm is applied in the training phase to maximize the effect of the proposed SJM module.
3.4.1 Domain-agnostic Loss. We employ a domain-agnostic loss (Eq. 8) whose effect scopes are all source domains, i.e., global effect. More concretely, global influence refers to giving an anchor sample $x$, its negative samples may come from any domain. By increasing the distance between pairs of negative samples, the model can learn the differences between them. However, because the identity and domain in DG person re-identification are highly correlated, the model can easily distinguish the features of different domains by focusing on domain bias. This makes the model ignore identity-related domain-independent information and reduces its generalization ability. Our SJM module exchanges style information across domains, allowing the model to focus on identity-related information. In addition, we use a domain-specific loss to reduce the model’s attention to domain-related information.

3.4.2 Domain-specific Loss. Different from domain-agnostic loss, the effect scopes of domain-specific loss is a specific domain. The specific loss for domain $D_k$ is formulated as follows:

$$L_{spec}^k = \frac{1}{B} \sum_{i=1}^{B} L_{ce}(\varphi_k(f(x_i^k)), y_i^k) + L_{tri}(f(x_i^k), y_i^k).$$  \hspace{1cm} (17)

Then, the overall specific loss is as follows:

$$L_{spec} = \frac{1}{K} \sum_{k=1}^{K} L_{spec}^k.$$  \hspace{1cm} (18)

It means that the anchor sample and its negative sample must belong to the same domain, i.e., local effect. Then, the model is not affected by domain bias and focuses on the domain-irrelevant information. In addition, SJM brings more styles to the features in the same domain. More hard samples are generated, and the model becomes more discriminative. Finally, the overall training loss is:

$$L_{all} = \lambda L_{agno} + (1 - \lambda) L_{spec}.$$  \hspace{1cm} (19)

3.4.3 Meta Optimizing. To maximize the effectiveness of the SJM module, we apply a Model-Agnostic Meta-Learning (MAML) algorithm to our approach. The MAML adopts the concept of “learning-to-learn”. It splits the source domains into meta-train and meta-test to simulate the domain bias, so as to improve the model generalization. Our SJM module can enlarge the domain bias by enriching the style diversity of meta-train, making the learning process of learning-to-learn harder and further improving the model generalization.

Specifically, we randomly split the $K$ source domains into $K - 1$ meta-train domains and 1 meta-test domain. In the meta-train stage, the images are sampled from meta-train domains and grouped into a mini-batch, denoted as $X_S$. Same as the previous training process, we feed these images into the network with the SJM module $f_S(\cdot; \Theta)$. We can calculate the meta-train loss $L_{mtr}$ (as same as $L_{all}$) based on the stylized changed features. The gradient with respect to the meta-train loss is $\nabla_\Theta L_{mtr}(f_S(X_S; \Theta))$. Then, we optimize the network and achieve an extra updated network:

$$L_{all} = \lambda L_{agno} + (1 - \lambda) L_{spec}.$$  \hspace{1cm} (20)

where $\lambda$ denotes the learning rate of the meta-train optimizer. In the meta-test stage, we sample images from the meta-test domain and from a mini-batch, denoted as $X_T$. Features in the meta-test stage are produced by the updated network $f(\cdot; \Theta')$, where the SJM module is discarded. The meta-test loss $L_{mte}$ is calculated as well, in which $L_{mte}$ is the same as $L_{agno}$ because the specific-domain loss in $L_{all}$ is the same as $L_{agno}$ when there
Algorithm 1: Training Procedure

Input: Source domains \( D = \{ D_k \}_{k=1}^K \); Learning rate hyperparameters \( \alpha, \beta, \gamma \).
Output: Model with the SJM module \( f_s(\cdot; \Theta) \); Model w/o the SJM module \( f(\cdot; \Theta) \); Domain-specific classifiers \( \{ \phi_k(\cdot; W_k) \}_{k=1}^K \); Domain-agnostic classifier \( \phi_g(\cdot; W_g) \).

Initialization: Identity style memory \( \bar{M} \); Identity similarity memory \( \bar{S} \); Domain distance memory \( \bar{D} \).

for iter in iterations do
  Sample \( K - 1 \) domains as meta-train \( D_{mtr} \) and the remaining \( D_t \) as meta-test \( D_{mte} \).
  // For simplicity, we denote \( \phi_k(\cdot; W_k), k \neq t \) and \( \phi_g(\cdot; W_g) \) as \( \phi(\cdot; W) \).
  Meta-training:
  Sample a mini-batch \( X_S \) from \( D_{mtr} \).
  Extract the normal features \( F \) and jittered features \( F' \) by model with the SJM module:
  \[
  F = \{ F; F' \} = f_s(X_S; \Theta);
  \]
  Compute meta-train losses by Eq. 19, where the memory \( \bar{M} \) is updated:
  \[
  L_{mtr} = L_{all}(\phi_s(F^*; W^*), F^*);
  \]
  Compute the gradients w.r.t the original model \( \Theta \) and classifier \( W^* \):
  \[
  \nabla_{\Theta} = \frac{\partial L_{mtr}(X_S; \Theta, W^*)}{\partial \Theta};
  \]
  \[
  \nabla_{W^*} = \frac{\partial L_{mtr}(X_S; \Theta, W^*)}{\partial W^*};
  \]
  Update the original model and meta-train classifiers parameters by:
  \[
  \Theta' \leftarrow \Theta - \alpha \nabla_{\Theta};
  \]
  \[
  W'^* \leftarrow W^* - \alpha \nabla_{W^*};
  \]
  Calculate similarity \( \bar{S} \) with global weight \( W_g^* \) by Eq. 10 and update the memory \( \bar{S} \);
  Calculate domain distances with extracted features \( F^* \) by Eq. 14 and update the memory \( \bar{D} \);
  Meta-testing:
  Sample a mini-batch \( X_T \) from \( D_{mte} \).
  Extract the normal features \( F \) by model w/o the SJM module:
  \[
  F = f(X_T; \Theta', W_t);
  \]
  Compute meta-test losses w/o the SJM module by Eq. 19:
  \[
  L_{mte} = L_{all}(\phi_s(F; W_t), F);
  \]
  Compute the gradient w.r.t the meta-test classifier \( W_t^* \):
  \[
  \nabla_{W_t^*} = \frac{\partial L_{mte}(X_T; \Theta', W_t^*)}{\partial W_t^*};
  \]
  Update the meta-test classifier parameters by:
  \[
  W_t'^* \leftarrow W_t^* - \beta \nabla_{W_t^*};
  \]
  Meta-optimizing
  Update the original model parameters \( \Theta \) by:
  \[
  \Theta \leftarrow \Theta - \gamma \frac{\partial L_{mtr}(X_S; \Theta, W^*) + \beta L_{mte}(X_T; \Theta, W_t^*)}{\partial \Theta};
  \]
end for
exists only one domain. Finally, we utilize the combination of the meta-train and meta-test losses to optimize the original model as follows:

$$\Theta \leftarrow \Theta - \gamma (\nabla_{\Theta} L_{mtr}(f_s(X_i; \Theta)) + \beta \nabla_{\Theta} L_{mte}(f(X_i; \Theta'))),$$

(21)

where $\gamma$ is the learning rate of the meta-test optimizer and $\beta$ is the hyper-parameter to balance the gradient of the meta-train and meta-test losses. The overall training procedure of MAML is illustrated in Algorithm 1.

We randomly stylize images in meta-train domains and keep images in meta-test domains unchanged to mimic real train/test scenarios. The intuition behind MAML is to expose the model to domain shift during training in the hope that the model can avoid overfitting to source domain bias. In conventional methods of applying MAML directly to the meta-train/test domain, the domain bias between meta-train/test domains will gradually decrease after the model witnesses more and more training data in the iterative process. However, our SJM module can effectively mitigate this limitation. More concretely, it introduces style variations to features during meta-training at each iteration, while keeping the style of features constant during meta-testing. This deliberate manipulation of style information makes it more challenging to adapt to the domain gap between meta-training and meta-testing domains. To learn a more generalizable model, the model has to ignore the influence of style variations between meta-train/test domains and pay more attention to the identity-relevant information.

4 EXPERIMENTS

4.1 Datasets and Evaluation Metrics

Following the previous works [5, 21, 23, 52], we conduct our experiments on public person ReID or Pearson-search datasets, including Market1501 [72], DukeMTMC-reID [49, 74], CUHK02 [29], CUHK03 [30], MSMT17 [64], CUHK-SYSU [66] and four small ReID datasets including PRID [18], GRID [38], VIPeR [13], and iLIDs [73].

**Multi-source DG-ReID task.** On the multi-source DG-ReID task, we evaluate our methods on large-scale datasets and small-scale datasets, respectively. On large-scale datasets, we train and evaluate on four large datasets, i.e. Market-1501, DukeMTMC-reID, CUHK03, and MSMT17. Concretely, we choose one of these four datasets as the evaluation dataset and use the remaining three datasets as our training datasets. Different from evaluation strategies on large datasets, two evaluation protocols (termed as protocol-1 and protocol-2) are adopted on small-scale datasets for comparison with previous methods. Both two protocols are evaluated on all small-scale datasets (i.e. PRID, GRID, VIPeR, and iLIDs) and trained on different datasets. Specifically, in protocol-1, we train our model on six large-scale datasets, i.e. Market-1501, DukeMTMC-reID, CUHK02, CUHK03, CUHK-SYSU, and MSMT17. In protocol-2, we train our methods on four datasets, i.e. CUHK03, DukeMTMC-reID, Market-1501, and MSMT17.

**Single-source DG-ReID task.** We have also conducted experiments on single-source DG-ReID tasks. We use two datasets Market-1501 and DukeMTMC-reID in our experiments. In the single-source setting, one of these datasets is used for training and the other one is used for evaluation.

Following the common evaluation metric, we use the Cumulative Matching Characteristic (CMC) at Rank-$k$ and mean Average Precision (mAP) to evaluate the model’s performance on target domains.

4.2 Implementation Details

We adopt ResNet50 [15] and IBN-ResNet50 [43] pretrained on ImageNet as our backbones, respectively. We add a Batch Normalization (BN) layer after the global pooling layer to get the ReID feature. A linear classifier is added after the BN layer to get classification predictions used for calculating cross-entropy loss and triplet loss [17]. We use the above structure as our baseline. Note that this baseline is different from the strong baseline we proposed. Images are resized to 256 × 128, and random cropping and random flipping are utilized as data augmentation. The batch size of each specific domain is set to 128, including 32 identities and 4 images per identity. In the training
stage, we evenly sample mini-batch from each source domain and combine all these batches as the model's input. For optimizing the model, we use Adam optimizer with a weight decay of \(5 \times 10^{-4}\). The learning rate of the meta-train phase and meta-test phase are initialized as \(3.5 \times 10^{-4}\) and are decayed by 0.1 at the 30th and 50th epochs respectively. We train the model on 4 GTX 1080Ti GPUs for 90 epochs. Our SJM module is plugged after the stage-1 of the backbone, and its scale \(s\) is set to 4. In addition, the \(\lambda\) in \(L_{all}\) is 0.1.

Our method is tailored for domain generalization in person re-identification scenarios, obviating the need for supervised fine-tuning and domain adaptation when applied to new domains. Therefore, our method is superior in terms of computational efficiency and scalability, as it can be readily deployed off-the-shelf. However, it's important to note that while our method offers significant advantages, it is still constrained by the source training domain. It may not achieve optimal generalization ability when confronted with large disparities between the target and source domains. In such cases, our method may fall short compared to supervised fine-tuning and domain adaptation methods.

4.3 Comparison with State-of-the-Art methods

We compare our methods with the state-of-the-art methods on the multi-source DG-ReID task on both large-scale datasets and small-scale datasets. We have also conducted experiments on single-source DG-ReID tasks to demonstrate the effectiveness and generalization ability of our methods.

4.3.1 Results on large ReID datasets under multi-source tasks. To demonstrate the effectiveness of our method, we compare it with the state-of-the-art approaches (SOTAs) on the large-scale DG-ReID benchmark, including QAConv [34], SNR [23], M3L [71], RaMoE [5], MixNorm [47], SuA-SpML [70], and OSNet [76]. For simplicity, we denote Market-1501 as M, DukeMTMC-reID as D, CUHK03 as C3, and MSMT17 as MS.

Following the evaluation protocol for DG-ReID in [5, 71], we employ a leave-one-out protocol to partition the four datasets into training and testing sets. Specifically, we select three datasets for training while reserving one as the test dataset. It is important to note that all images within the training datasets are used for model training, irrespective of train/test splits. The results of these experiments are presented in Table 1. For a fair comparison, we report results based on both ResNet50 and IBN-ResNet50 backbones. Under the ResNet50 backbone, our SVIL approach demonstrates significant performance gains over our baseline, achieving improvements of 5.90%, 8.73%, 9.31%, and 7.16% in terms of mAP across the respective datasets. Comparing our SVIL method to the SuA-SpML method under the ResNet50 backbone, our approach outperforms SuA-SpML by 6.50%, 6.93%, 2.61%, and 2.26% in terms of mAP. Under the IBN-ResNet50 backbone, our method outperforms the SuA-SpML method by 3.40%, 3.99%, 0.12%, and 0.22% in terms of mAP.
mAP on datasets C3, D, M, and MS, respectively. Switching to the IBN-ResNet50 backbone, our method maintains its superiority, outperforming the SuA-SpML method by 6.23% and 3.92% in mAP and Rank-1 metrics. These experiments demonstrate that our approach achieves state-of-the-art performance across all target domains, regardless of the backbone architecture used. Furthermore, it consistently outperforms the baseline model by a substantial margin, providing clear evidence of the effectiveness and superiority of our proposed method.

Table 2. Comparison (%) with the state-of-the-arts DG ReID methods on small-scale person ReID benchmarks.

| Method      | Source | Reference | Target: PRID mAP Rank-1 | Target: GRID mAP Rank-1 | Target: VIPeR mAP Rank-1 | Target: iLIDs mAP Rank-1 | Average |
|-------------|--------|-----------|-------------------------|-------------------------|--------------------------|--------------------------|---------|
| DIMN [52]  | CVPR 2019 | 51.95 39.20 41.09 29.28 60.12 51.23 78.39 70.17 | 57.89 47.47 |
| DualNorm [21] | BMVC 2019 | 64.90 60.40 45.70 41.40 58.00 53.90 78.50 74.80 | 61.78 57.62 |
| MMD-AAE [28] | CVPR 2018 | - 57.20 - 47.40 - 58.40 - 84.80 - | - 61.95 |
| SNR [23] | C2+C3+ | CVPR 2020 | 66.50 52.10 47.70 40.20 61.30 52.90 89.90 84.10 | 66.35 57.33 |
| RaMoE [5] | D+M+CS | CVPR 2021 | 67.30 57.70 54.20 46.80 64.60 56.60 90.20 85.00 | 69.08 61.52 |
| MixNorm [47] | TMM 2022 | 59.10 49.20 - - 60.60 50.80 - - | 59.85 50.00 |
| Baseline | - | 63.77 53.00 50.63 39.00 62.76 55.10 84.80 76.67 | 64.88 55.47 |
| SVIL (Ours) | - | 69.39 58.05 56.18 44.20 71.70 63.29 | 88.97 85.30 71.56 62.63 |
| SNR [23] | CVPR 2020 | 60.00 49.00 41.30 30.40 65.00 55.10 91.90 87.00 | 64.55 55.38 |
| DMG-Net [1] | CVPR 2021 | 69.70 59.70 47.20 37.30 70.90 62.30 88.20 83.00 | 69.00 60.60 |
| RaMoE [5] | C3+D+ | CVPR 2021 | 66.80 56.90 53.90 43.40 72.70 63.40 92.30 88.40 | 71.42 63.02 |
| META [67] | M+MS | ECCV 2022 | 71.70 60.10 52.40 46.80 61.50 63.40 83.50 79.20 | 70.90 63.80 |
| Baseline | - | 65.18 55.00 52.88 42.60 62.36 62.19 | 86.03 82.00 68.74 60.45 |
| SVIL (Ours) | - | 75.28 67.00 56.90 48.00 75.30 67.09 | 90.03 86.67 74.38 67.19 |

Table 3. Performance (%) comparison with the state-of-the-arts on the single-source DG-ReID task.

| Method      | M→D | D→M |
|-------------|------|------|
| IBN-ResNet [43] | 23.30 | 23.50 |
| OSNet [77] | 25.90 | 24.00 |
| OSNet-IBN [77] | 27.60 | 27.40 |
| CrossGrad [50] | 27.10 | 26.30 |
| QConv [33] | 28.70 | 27.20 |
| L2A-OT [78] | 29.70 | 29.20 |
| OSNet-AIN [77] | 30.50 | 30.60 |
| SNR [23] | 33.60 | 33.90 |
| SuA-SpML [70] | 33.40 | 35.20 |
| SVIL (Ours) | 35.08 | 35.05 |

4.3.2 Results on small ReID datasets under multi-source tasks. Following the previous works [5, 21, 23, 52], we employ two distinct evaluation protocols, Protocol-1 [52] and Protocol-2 [23], to assess the performance of our method across four smaller ReID datasets. To simplify notation, we denote CUHK02 as C2 and CUHKSYSU as CS. The results of these evaluations are shown in Table 2. Under Protocol-1, all images from the datasets M+D+C2+C3+CS are employed for training, irrespective of train/test splits. Subsequently, the trained network is tested on PRID, GRID, VIPeR, and iLIDs datasets, respectively. Our method outputs better results than other methods on PRID, GRID, and VIPeR datasets under mAP and Rank-1. Furthermore, our proposed method showcases substantial improvements over the baseline across all datasets. Under Protocol-2, all images in M+D+C3+MS are utilized for training regardless of train/test splits. In this scenario, our approach not only
Table 4. Ablations studies on different components of our method.

| Backbone          | SJM | MAML | Loss     | C+D+MS→M | C+M+MS→D |
|-------------------|-----|------|----------|----------|----------|
|                   |     |      |          | mAP Rank-1 | mAP Rank-1 |
| ResNet50          | ×   | ×    | L_{spec} | 50.74 77.32 | 52.64 69.52 |
| ResNet50          | ✓   | ×    | L_{spec} | 53.76 80.40 | 53.99 71.50 |
| IBN-ResNet50      | ×   | ×    | L_{spec} | 58.12 81.80 | 57.87 73.43 |
| IBN-ResNet50      | ✓   | ×    | L_{spec} | 58.77 82.36 | 58.39 73.83 |
| ResNet50          | ×   | ✓    | L_{all}  | 56.09 80.29 | 55.74 71.23 |
| ResNet50          | ✓   | ✓    | L_{all}  | 58.18 82.33 | 57.26 73.61 |
| IBN-ResNet50      | ×   | ✓    | L_{all}  | 61.40 83.49 | 60.24 75.58 |
| IBN-ResNet50      | ✓   | ✓    | L_{all}  | 63.95 84.77 | 60.86 76.26 |
| IBN-ResNet50      | ×   | ✓    | L_{all}  | 62.30 83.43 | 60.72 76.26 |
| IBN-ResNet50      | ✓   | ✓    | L_{all}  | 64.95 86.05 | 61.68 77.33 |

Table 5. Ablations studies of the SJM Module on different identity relationship modeling.

| Method            | Cross-domain | C+D+MS→M | C+M+MS→D |
|-------------------|---------------|----------|----------|
|                   |               | mAP Rank-1 | mAP Rank-1 |
| SVIL w/o SJM      | ×             | 62.30 83.43 | 60.72 76.26 |
| SVIL w/ SJM       | ×             | 64.01 85.34 | 61.38 76.91 |
| SVIL w/ SJM       | ✓             | 64.95 86.05 | 61.68 77.33 |

outperforms the baseline by a significant margin but also achieves competitive results in comparison to state-of-the-art approaches.

4.3.3 Results on large ReID datasets under multi-source tasks. Since images within the same dataset often originate from different cameras, resulting in distinct styles, we divide single-source datasets into K sub-datasets based on camera information to simulate a multi-source setting. Specifically, we partition datasets like Market1501 [72] and DukeMTMC-reID [74] into three subsets, ensuring that the camera labels in different subsets do not conflict with each other. This division introduces style variations across various sub-datasets. For simplicity, we exclude the domain-specific loss $L_{spec}$ and retain the domain-agnostic loss $L_{agno}$ with a weight parameter $\lambda = 1$. We then incorporate our method into the single-source DG-ReID task in the same manner as the multi-source task. In Table 3, ‘M → D’ denotes that Market1501 serves as the labeled source training domain, and DukeMTMC-reID is the unseen target domain. Our experimental results demonstrate that our approach achieves competitive performance when compared to state-of-the-art methods. However, there is still room for improvement when applying our method to single-domain problems. Our approach’s division of a single-source domain is somewhat coarse. When subsets are divided solely based on camera labels, the identity label space in each subset may overlap, deviating from the real multi-source DG-ReID problem. On the other hand, the omission of the domain-specific loss is another limiting factor. Both of these factors hinder our method from achieving optimal performance. Nevertheless, the comparison in Table 3 underscores that our method enhances the generalization capability of models, showcasing its potential for broader applicability and performance improvements in more refined multi-source settings.

4.4 Ablation Study

4.4.1 Effectiveness of our SJM module on different backbones. We integrate our SJM (Style-Joint Module) module into two different backbones, namely ResNet-50 [15] and IBN-ResNet-50 [43], and conduct experiments using
Table 6. Study about which stage of IBN-ResNet50 to plug the SJM module.

| Method                  | C+D+MS→M       | C+M+MS→D       |
|-------------------------|----------------|----------------|
| SVIL w/o SJM            | 62.30 83.43    | 60.72 76.26    |
| SJM after stage-0       | 64.44 85.15    | 61.16 77.06    |
| SJM after stage-1       | 64.95 86.05    | 61.68 77.33    |
| SJM after stage-2       | 62.90 84.83    | 59.13 74.46    |
| SJM after stage-3       | 58.34 81.41    | 59.01 75.49    |
| SJM after stage-4       | 50.62 75.00    | 51.60 70.11    |

Fig. 3. Visualization of activation maps of different features on method w/o SJM and method w/ SJM. The maps of the method with SJM pay more attention to identity-relevant discriminative information, e.g. clothes and bags.

various training strategies. The results presented in Table 4 clearly demonstrate that models incorporating the SJM module consistently outperform those without it.

For instance, when comparing ResNet50+SJM to ResNet50, we observe improvements of 3.02% in mAP and 3.08% in Rank-1 on the M target domain. Similarly, IBN-ResNet50+SJM outperforms IBN-ResNet50 by 0.65% in mAP and 0.56% in Rank-1 on the same target domain. It’s worth noting that the improvement in IBN-ResNet50 is relatively modest due to the presence of Instance Normalization (IN) within IBN-ResNet50, which already enhances the style-insensitivity of the model to some extent. As a result, there is partial redundancy in the effects achieved by our SJM module.

In summary, these comparative experiments provide clear evidence of the effectiveness of our SJM module in improving the performance of models across different backbone architectures, highlighting its valuable contribution to addressing style variations in the domain of person re-identification.

4.4.2 Effectiveness of our SJM module with MAML. To demonstrate the effectiveness of combining the SJM module with MAML (Model-Agnostic Meta-Learning). The results presented in Table 4 reveal that the SJM module’s performance improvement is more pronounced when used in conjunction with MAML, compared to when it is used without MAML. As an illustration, consider the scenario C+D+MS→M. In this case, the mAP/Rank-1 of IBN-ResNet50+SJM+MAML outperforms IBN-ResNet50+MAML by 2.55%/1.28%. Moreover, IBN-ResNet50+SJM exhibits an improvement of 0.65% in mAP and 0.56% in Rank-1 compared to IBN-ResNet50. This comparison clearly demonstrates that the combination of the SJM module and MAML produces a synergy effect, leading to
superior performance in domain generalization tasks. The experiments demonstrate the significant performance gains achievable by leveraging the SJM module in conjunction with MAML, showing the potential for enhanced model adaptability and generalization in challenging scenarios.

4.4.3 Effectiveness of our SJM module on different loss functions. The results in Table 4 demonstrate the effectiveness of the SJM module with various losses. For instance, in C+D+MS→M, the improvement of $L_{aggr}$ on IBN-ResNet-50+MAML+$L_{spec}$+SJM is 0.1%/1.37% higher than the improvement on IBN-ResNet-50+MAML+$L_{spec}$ on mAP/Rank-1 metrics. This comparison confirms that our SJM module can make cross-domain samples harder.

4.4.4 Effectiveness of different identity relationship modeling. We conduct experiments to confirm the effectiveness of identity relationship modeling in the SJM module. The results in Table 5 show that only hard identity emphasis can improve the results, and cross-domain identity emphasis can further improve the model performance on the basis.

4.4.5 Effectiveness of which stage to plug the SJM Module. Our SJM module is designed as a plug-and-play component, allowing it to be inserted at various stages within the backbone network. To investigate how the placement of the SJM module affects performance, we conduct experiments using the Strong baseline with an IBN-ResNet backbone. The backbone comprises five stages, with stage-0 consisting of Conv, BN, and Max Pooling layers, while stage-1/2/3/4 represent the remaining convolutional blocks.

In Table 6, the results indicate that our method performs notably better when the SJM module is placed in shallow stages (0 and 1) but exhibits slightly diminished performance in deeper stages (2, 3, and 4). This phenomenon can be attributed to the fact that shallow features contain a greater abundance of low-level details, and applying style transformations to them incurs less loss of semantic information.

Our findings suggest that incorporating the SJM module in shallower stages is generally more effective. The main reason is that shallow-level features inherently possess weaker semantic information, and applying style transformations to them can introduce variations in non-identity-related details, such as background elements. On the other hand, deeper layers of the network are characterized by stronger semantic features but contain less style information. Introducing style perturbations at deeper stages can lead to the loss of identity-related information.
Fig. 5. Examples of ranking results on Market1501. The green and red boxes indicate the correct matchings and the wrong matchings, respectively. (a) is the result of baseline, and (b) is the result of our methods.

4.5 Qualitative Analysis

4.5.1 Visualization of Feature Maps. To better understand the influence of our SJM module, we visualize the intermediate feature maps in the method w/o SJM and method w/ SJM in Fig. 3. Following [23], we obtain each activation map by summarizing the feature maps along channels followed by a spatial $l_2$ normalization. We can observe that the maps of our method pay more attention to discriminate areas. For instance, as shown in the 2nd and 3rd columns in the maps, the method w/ SJM tends to focus more on bags and skirts, which have some discriminative details. Whereas the focus area of the method w/o SJM is scattered, including part of the background and other unimportant areas. These areas may contain more style information. This phenomenon confirms that our SJM module contributes to learning a more style-insensitive and more generalized model.

4.5.2 Visualization of Feature Distributions. In Fig. 4, we visualize the t-SNE [70] distributions of the features on the four benchmarks for baseline and our method. Different colors represent various datasets. Blue, red, and cyan denote the source datasets Duke, CUHK03, and MSMT17, respectively, while green indicates the target dataset Market1051. As shown in Fig. 4 (a), we roughly delineate the boundaries of these features by a black dotted line. We can see the feature distributions of different datasets on the baseline are separately distributed and have visible domain gaps. For example, most of the features of the MSMT17 dataset are concentrated on the left side of Fig. 4 (a). After adopting our method, these domain gaps are significantly diminished. The features of the MSMT17 dataset in Fig. 4 (c) are mixed up with those of other datasets, making it difficult to find clear boundary gaps, i.e., domain gap. Our method effectively blends the features from various domains, resulting in a more cohesive and unified feature distribution.

4.5.3 Visualization of Rank List. Fig. 5 demonstrates 5 pairs of ranking results, i.e., the ranking results of baseline in (a) and the ranking results of our method in (b). It is noticeable that the incorrect matches in the baseline typically involve individuals with similar clothing or background elements, which exhibit comparable styles.
Our method effectively mitigates these mismatches stemming from style variations. This observation serves as compelling evidence confirming the efficacy of our approach.

5 CONCLUSION

In this paper, we introduce the Style Variable and Irrelevant Learning (SVIL) method, which aims to mitigate the impact of style factors on model performance, ultimately yielding a more robust and generalized model. Specifically, we present the Style Random Jitter (SJM) module, designed to enhance the diversity of styles within source domains. This module encourages the model to prioritize identity-related information while disregarding style-related factors. Additionally, we seamlessly integrate the SJM module with a meta-learning algorithm, further enhancing the model’s generalization capabilities. The SVIL method presented in this work represents a significant step towards addressing domain generalization challenges in various applications, demonstrating the potential to improve model performance by mitigating the impact of style variations. Extensive experiments demonstrate the effectiveness and superiority of our method.

REFERENCES

[1] Yan Bai, Jile Jiao, Ce Wang, Jun Liu, Yihang Lou, Xuetao Feng, and Ling-Yu Duan. 2021. Person30K: A Dual-Meta Generalization Network for Person Re-Identification. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2123–2132.
[2] Yogesh Balaji, Swami Sankaranarayanan, and Rama Chellappa. 2018. Metareg: Towards domain generalization using meta-regularization. In Advances in Neural Information Processing Systems.
[3] Seokheon Choi, Taekyung Kim, Minki Jeong, Hyounghseoob Park, and Changick Kim. 2021. Meta Batch-Instance Normalization for Generalizable Person Re-Identification. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 3425–3435.
[4] Ruihang Chu, Yifan Sun, Yadong Li, Zheng Liu, Chi Zhang, and Yichen Wei. 2019. Vehicle Re-Identification With Viewpoint-Aware Metric Learning. In Proceedings of the International Conference on Computer Vision.
[5] Yongxing Dai, Xiaotong Li, Jun Liu, Zekun Tong, and Ling-Yu Duan. 2021. Generalizable person re-identification with relevance-aware mixture of experts. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 16145–16154.
[6] Yuhang Ding, Hehe Fan, Mingliang Xu, and Yi Yang. 2020. Adaptive Exploration for Unsupervised Person Re-identification. ACM Transactions on Multimedia Computing, Communications and Applications 16, 1 (2020), 3:1–3:19.
[7] Vincent Dumoulin, Jonathon Shlens, and Manjunath Kudlur. [n. d.]. A Learned Representation For Artistic Style. In Proceedings of the International Conference on Learning Representations.
[8] Hehe Fan, Liang Zheng, Chenggang Yan, and Yi Yang. 2018. Unsupervised Person Re-identification: Clustering and Fine-tuning. ACM Transactions on Multimedia Computing, Communications and Applications 14, 4 (2018), 83:1–83:18.
[9] Chelsea Finn, Pieter Abbeel, and Sergey Levine. 2017. Model-agnostic meta-learning for fast adaptation of deep networks. In Proceedings of the International Conference on Machine Learning. 1126–1135.
[10] Yang Fu, Yunchao Wei, Guanzhan Wang, Yuqian Zhou, Honghui Shi, and Thomas S Huang. 2019. Self-similarity grouping: A simple unsupervised cross domain adaptation approach for person re-identification. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 6112–6121.
[11] Leon A Gatys, Alexander S Ecker, and Matthias Bethge. 2016. Image style transfer using convolutional neural networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2414–2423.
[12] Yixiao Ge, Dapeng Chen, and Hongsheng Li. [n. d.]. Mutual Mean-Teaching: Pseudo Label Refinery for Unsupervised Domain Adaptation on Person Re-identification. In Proceedings of the International Conference on Learning Representations.
[13] Douglas Grey and Hai Tao. 2008. Viewpoint invariant pedestrian recognition with an ensemble of localized features. In Proceedings of the European Conference on Computer Vision. Springer, 262–275.
[14] Guangxing Han, Xuan Zhang, and Chongrong Li. 2023. One-Shot Unsupervised Cross-Domain Person Re-identification. IEEE Transactions on Circuits and Systems for Video Technology (2023).
[15] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 770–778.
[16] Shuting He, Hao Luo, Pichao Wang, Fan Wang, Hao Li, and Wei Jiang. 2021. TransReID: Transformer-based Object Re-Identification. In Proceedings of the International Conference on Computer Vision.
[17] Alexander Hermans, Lucas Beyar, and Bastian Leibe. 2017. In defense of the triplet loss for person re-identification. arXiv preprint arXiv:1703.07737 (2017).
[18] Martin Hržer, Csaba Beleznai, Peter M Roth, and Horst Bischof. 2011. Person re-identification by descriptive and discriminative classification. In Proceedings of the Scandinavian Conference on Image Analysis. Springer, 91–102.
[45] Nan Pu, Yu Liu, Wei Chen, Erwinn M Bakker, and Michael S Lew. 2022. Meta reconciliation normalization for lifelong person re-identification. In Proceedings of the ACM International Conference on Multimedia. 541–549.

[46] Nan Pu, Zhun Zhong, Nicu Sebe, and Michael S Lew. 2023. A memorizing and generalizing framework for lifelong person re-identification. IEEE Transactions on Pattern Analysis and Machine Intelligence (2023).

[47] Lei Qi, Lei Wang, Yinghuan Shi, and Xin Geng. 2022. A Novel Mix-Normalization Method for Generalizable Multi-Source Person Re-Identification. IEEE Transactions on Multimedia (2022), 1–12.

[48] Xuelin Qian, Yanwei Fu, Yu-Gang Jiang, Tao Xiang, and Xiangyang Xue. [n. d.]. Multi-scale Deep Learning Architectures for Person Re-identification. In Proceedings of the International Conference on Computer Vision. 5409–5418.

[49] Ergys Ristani, Francesco Solera, Roger S. Zou, Rita Cucchiara, and Carlo Tomasi. 2016. Performance Measures and a Data Set for Multi-target, Multi-camera Tracking. In Eur. Conf. Comput. Vis. Worksh., Vol. 9914. 17–35.

[50] Shiv Shankar, Vihari Piratla, Soumen Chakrabarti, Siddhartha Chaudhuri, Preethi Jyothi, and Sunita Sarawagi. 2018. Generalizing Across Domains via Cross-Gradient Training. In Proceedings of the International Conference on Learning Representations.

[51] Hao Sheng, Kai Ly, Yang Liu, Wei Ke, Weifeng Lyu, Zhang Xiong, and Wei Li. 2020. Combining pose invariant and discriminative features for vehicle reidentification. IEEE Internet of Things Journal 8, 5 (2020), 3189–3200.

[52] Jifei Song, Yongxin Yang, Yi-Zhe Song, Tao Xiang, and Timothy M Hospedales. 2019. Generalizable person re-identification by domain-invariant mapping network. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 719–728.

[53] Liangchen Song, Cheng Wang, Lefei Zhang, Bo Du, Qian Zhang, Chang Huang, and Xinggang Wang. 2020. Unsupervised domain adaptive re-identification: Theory and practice. Pattern Recognition 102 (2020), 107173.

[54] Yifan Sun, Changmiao Cheng, Yuhan Zhang, Chi Zhang, Liang Zheng, Zhongdao Wang, and Yichen Wei. 2020. Circle Loss: A Unified Perspective of Pair Similarity Optimization. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition.

[55] Masato Tamura and Tomokazu Murakami. 2019. Augmented hard example mining for generalizable person re-identification. arXiv preprint arXiv:1910.05280 (2019).

[56] Wentao Tan, Changxing Ding, Pengfei Wang, Mingming Gong, and Kui Jia. 2023. Style Interleaved Learning for Generalizable Person Re-identification. IEEE Transactions on Multimedia (2023).

[57] Zhiquan Tang, Yunhe Gao, Yi Zhu, Zhi Zhang, Mu Li, and Dimitris Metaxas. 2021. CrossNorm and SelfNorm for Generalization under Distribution Shifts. In Proceedings of the International Conference on Computer Vision.

[58] Dmitry Ulyanov, Andrea Vedaldi, and Victor Lempitsky. 2017. Improved texture networks: Maximizing quality and diversity in feed-forward stylization and texture synthesis. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 6924–6932.

[59] Laurens Van der Maaten and Geoffrey Hinton. 2008. Visualizing data using t-SNE. Journal of Machine Learning Research 9, 11 (2008).

[60] Feng Wang, Xiang Xian, Jian Cheng, and Alan Loddon Yuille. 2017. NormFace: L2 Hypersphere Embedding for Face Verification. In Proceedings of the ACM International Conference on Multimedia. 1041–1049.

[61] Hao Wang, Yitong Wang, Zheng Zhou, Xing Ji, Di Hong Yang, Jingchao Zhou, Zhifeng Li, and Wei Liu. 2018. CosFace: Large Margin Cosine Loss for Deep Face Recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition.

[62] Shuai Wang, Zhihao Wu, Xiaobo Hu, Youfeng Lin, and Kai Li. 2023. Skill-based hierarchical reinforcement learning for target visual re-identification via memory-based multi-source meta-learning for person re-identification. arXiv preprint arXiv:2105.05280 (2021).

[63] Longhui Wei, Shiliang Zhang, Wen Gao, and Qi Tian. 2018. Person transfer gan to bridge domain gap for person re-identification. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 719–728.

[64] Zhikang Wang, Lihuo He, Xiaoguang Tu, Jian Zhao, Xinbo Gao, Shengmei Shen, and Jiashi Feng. 2021. Robust video-based person re-identification by hierarchical mining. IEEE Transactions on Circuits and Systems for Video Technology 32, 12 (2021), 8179–8191.

[65] Wentao Tan, Changxing Ding, Pengfei Wang, Mingming Gong, and Kui Jia. 2023. Style Interleaved Learning for Generalizable Person Re-identification. IEEE Transactions on Multimedia (2023).

[66] Nan Pu, Zhun Zhong, Nicu Sebe, and Michael S Lew. 2023. A memorizing and generalizing framework for lifelong person re-identification. In Proceedings of the International Conference on Computer Vision and Pattern Recognition. 3415–3424.

[67] Lin Wu, Chunhua Shen, and Anton van den Hengel. 2016. PersonNet: Person Re-identification with Deep Convolutional Neural Networks. CoRR abs/1601.07235 (2016).

[68] Tong Xiao, Shuang Li, Bochao Wang, Liang Lin, and Xiaoqiang Wang. 2017. Joint detection and identification feature learning for person search. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 3415–3424.

[69] Boqiang Xu, Jian Liang, Lingxiao He, and Zhenan Sun. 2021. Mimic embedding via adaptive aggregation: learning generalizable person re-identification. In Proceedings of the European Conference on Computer Vision. Springer. 372–388.

[70] Ming Ye, Jianbing Shen, Gaojie Lin, Tao Xiang, Ling Shao, and Steven CH Hoi. 2021. Deep learning for person re-identification: A survey and outlook. IEEE Transactions on Pattern Analysis and Machine Intelligence (2021).

[71] Shijie Tu, Feng Zhu, Dapeng Chen, Rui Zhao, Haoshin Chen, Shixiang Tang, Jinguo Zhu, and Yu Qiao. 2021. Multiple Domain Experts Collaborative Learning: Multi-Source Domain Generalization For Person Re-Identification. CoRR arXiv:2105.12355 (2021).

[72] Lei Zhang, Zhipu Liu, Wensheng Zhang, and David Zhang. 2023. Style Uncertainty Based Self-Paced Meta Learning for Generalizable Person Re-identification. IEEE Transactions on Image Processing 32 (2023), 2107–2119.

[73] Yuyang Zhao, Zhun Zhong, Fengxiang Yang, Zhihong Lao, Yaojin Lin, Shaoli Li, and Nicu Sebe. 2021. Learning to generalize unseen domains via memory-based multi-source meta-learning for person re-identification. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 6277–6286.
[72] Liang Zheng, Liyue Shen, Lu Tian, Shengjin Wang, Jingdong Wang, and Qi Tian. 2015. Scalable person re-identification: A benchmark. In Proceedings of the International Conference on Computer Vision. 1116–1124.
[73] Wei-Shi Zheng, Shaogang Gong, and Tao Xiang. 2009. Associating Groups of People.. In Proceedings of the British Machine Vision Conference, Vol. 2. 1–11.
[74] Zhedong Zheng, Liang Zheng, and Yi Yang. 2017. Unlabeled samples generated by gan improve the person re-identification baseline in vitro. In Proceedings of the International Conference on Computer Vision. 3754–3762.
[75] Xian Zhong, Tianyou Lu, Wexin Huang, Mang Ye, Xuemei Jia, and Chia-Wen Lin. 2021. Grayscale enhancement colorization network for visible-infrared person re-identification. IEEE Transactions on Circuits and Systems for Video Technology 32, 3 (2021), 1418–1430.
[76] Kaiyang Zhou, Yongxin Yang, Andrea Cavallaro, and Tao Xiang. 2019. Omni-scale feature learning for person re-identification. In Proceedings of the International Conference on Computer Vision. 3702–3712.
[77] Kaiyang Zhou, Yongxin Yang, Andrea Cavallaro, and Tao Xiang. 2021. Learning Generalisable Omni-Scale Representations for Person Re-Identification. IEEE Transactions on Pattern Analysis and Machine Intelligence PP (2021).
[78] Kaiyang Zhou, Yongxin Yang, Timothy M. Hospedales, and Tao Xiang. 2020. Learning to Generate Novel Domains for Domain Generalization. In Proceedings of the European Conference on Computer Vision, Vol. 12361. 561–578.
[79] Kaiyang Zhou, Yongxin Yang, Yu Qiao, and Tao Xiang. 2021. Domain Generalization with MixStyle. In Proceedings of the International Conference on Learning Representations.