IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 20 June 2024, accepted 3 July 2024, date of publication 9 July 2024, date of current version 19 July 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3425602

RESEARCH ARTICLE

Ladder Curriculum Learning for Domain Generalization in Cross-Domain Classification

XIAOSHUN WANG[1], SIBEI LUO[1], AND YIMING GAO[2]

[1]Huzhou Key Laboratory of Green Energy Materials and Battery Cascade Utilization, School of Intelligent Manufacturing, Huzhou College, Huzhou, Zhejiang 313000, China

[2]School of Intelligent Manufacturing, Huzhou College, Huzhou, Zhejiang 313000, China

Corresponding author: Xiaoshun Wang (wangxiaoshun999@hotmail.com)

This work was supported by the Huzhou Science and Technology Program Project under Grant 2022GZ19.

ABSTRACT Domain generalization seeks to acquire a domain-invariant representation from various source domains, thereby enabling a model to achieve robust generalization across previously unseen target domains. Most existing domain generalization methods for cross-domain classification tasks typically train models using examples randomly presented from all source domains. This may lead to training instability due to the presence of conflicting gradients, thus affecting the model's generalization ability. Recently, curriculum learning has been successfully applied in domain generalization. However, we find that existing methods only focus on domain shift and ignore intra-domain category shift, which still leads to gradient conflict problems and affects the model's generalization ability. To address the aforementioned challenges, we put forward a novel and general methodology known as ladder curriculum learning (LCL) as a solution to the above problem. Specifically, we deliver the source domain data in stages according to the order from easy to difficult. We focus not only on the inter-domain data sorted from easy to difficult, known as inter-domain curriculum learning, but also on the intra-domain data sorted from easy to difficult, known as intra-domain curriculum learning. Through the combined effects of inter-domain curriculum learning and intra-domain curriculum learning, our proposed LCL method can effectively address the optimization problem concerning conflicting gradient directions. Experiments conducted on widely used public datasets show that the LCL method can significantly improve baseline methods, with an improvement margin of up to 1.5%. Through experiments, we also find that the LCL method can be successfully applied to existing domain generalization methods, further enhancing the network's generalization capability with an average improvement rate of 1%.

INDEX TERMS Conflicting gradients, domain generalization, ladder curriculum learning.

I. INTRODUCTION

With the rapid development of deep learning, deep neural networks (DNN) have become the primary solution for applications in various fields such as image classification and image recognition. DNN typically assume that the training and testing data are independently and identically distributed (i.i.d.) [1]. This assumption, however, is not applicable in numerous real-world scenarios. For example, when utilizing segmentation models trained on sunny days to handle rainy and foggy environments [2], or attempting to recognize art paintings using models trained on photographs [3], an unavoidable decrease in performance is often observed in such out-of-distribution deployment scenarios. The answers to these questions all hinge on the machine learning models' ability to effectively address a fundamental challenge, known as the domain shift [4] problem. This problem pertains to the distributional shift between a given set of training (source) data and a different set of test (target) data [5].

The associate editor coordinating the review of this manuscript and approving it for publication was Mohammad J. Abdel-Rahman.

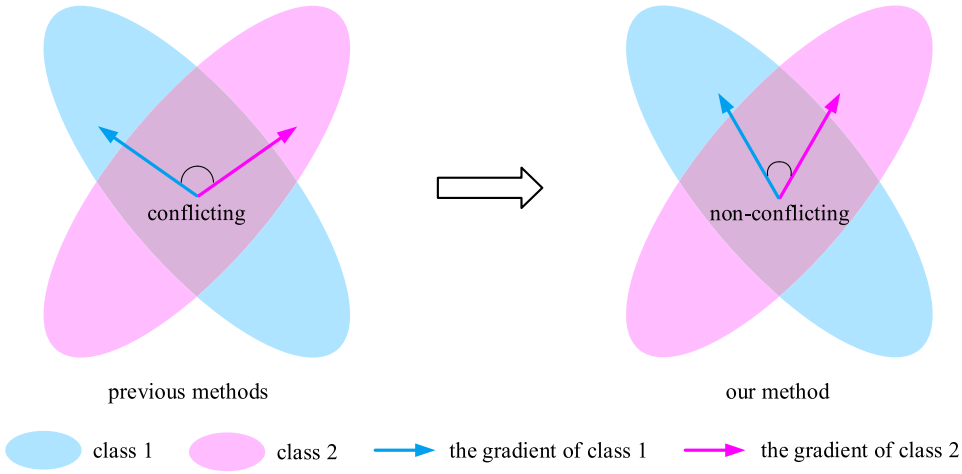

FIGURE 1. Comparison between existing methods and our proposed approach in terms of design philosophy. Existing methods focus solely on domain shift while neglecting category shift, which may result in conflicting gradients during the optimization process, making it challenging for the network parameters to follow consistent optimization directions. This instability could hinder the learning of domain-invariant representations. In contrast, our proposed approach emphasizes category shift and aims to minimize the generation of conflicting gradients, enabling the network model to follow a consistent optimization direction.

Domain generalization (DG), which focuses on the task of generalizing a predictive model across distinct domains, specifically targets the challenges posed by non-i.i.d. supervised learning scenarios. The primary objective of domain generalization is to train a predictive model by harnessing labelled data from multiple source domains and enhancing its capacity to generalize effectively to an unseen target domain. In this context, a domain is defined as a joint probability distribution $P(x, y)$. Domain generalization has been extensively investigated in diverse applications, including person re-identification [6], object recognition [7], and fault diagnosis [8].

In the DG context, researchers have conventionally employed pre-trained weights from the ImageNet dataset [9] to train their models. This transfer learning approach proves to be pragmatic, especially when dealing with limited data. However, this model exhibits a bias towards the ImageNet dataset even before the training process. This phenomenon becomes notably pronounced in multi-domain datasets with well-represented domain shift. Therefore, it is crucial to pay attention to the sequence of training data from multiple domains, as arbitrarily changing the order can cause conflicting gradient issues, affecting the model's generalization ability. This study aims to address the conflicting gradient problem to improve the model's generalization ability.

Curriculum learning [10] involves the sequential presentation of data that progresses from easy-to-learn to difficult-to-learn during the training of deep learning models. This approach is implemented in the context of a deep learning model that emulates a person's 18-year knowledge trajectory, encompassing knowledge from beginner to university-level understanding. Currently, there have been studies applying the idea of curriculum learning to DG. As an illustration, the researchers in literature [11] introduce a novel approach known as inter-domain curriculum learning (IDCL), which employs a curriculum learning-based training strategy in the context of DG by effectively utilizing domain shift. Within IDCL, the training process entails the sequential presentation of weak domain-shift data followed by strong domain-shift data. However, we find that existing methods only focus on domain shift and ignore category shift within domains, which still lead to conflicting gradient issues and affect the model's generalization ability, as shown in Figure 1 (left). Therefore, to address the aforementioned issues, we propose Ladder Curriculum Learning (LCL) to minimize the generation of conflicting gradients, as illustrated in Figure 1 (right).

We metaphorically regard domain data as a course and category data as the content of chapters. We believe it is important to consider not only the difficulty between courses but also the difficulty within course chapters. Based on the above idea, we introduce the concepts of inter-domain curriculum learning and intra-domain curriculum learning in domain generalization. Inter-domain curriculum learning refers to the ordering of multiple source domain data from easy to difficult, while intra-domain curriculum learning refers to the ordering of category data within a source domain from easy to difficult. Considering both inter-domain and intra-domain curriculum learning, we propose the LCL method.

This study focuses on cross-domain classification tasks. Therefore, in the implementation of LCL, the data used for training is arranged in ascending order based on data shift. Generally, it can be sorted according to the distance from the domain where the backbone is pre-trained. We calculate the distances between the source domain data and ImageNet data, as well as between the category data within the source domain and ImageNet data, using cosine similarity. These two distances are referred to as domain-level distance and

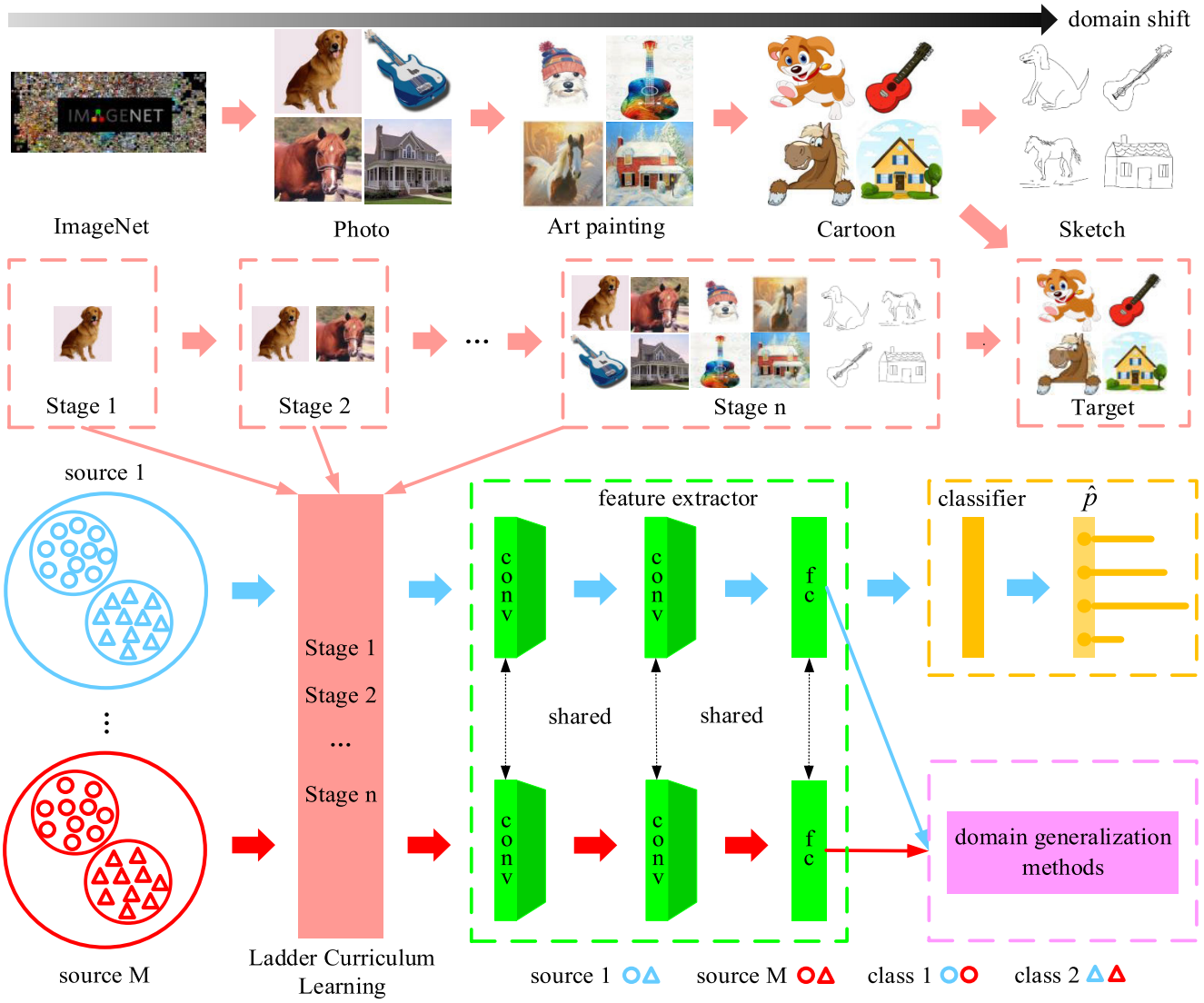

FIGURE 2. Generic network structure. Taking the PACS dataset as an example, we meticulously partition multiple source domain data with different domain shifts, sorting the data from easy to difficult in each stage to alleviate the issue of training instability caused by conflicting gradients. In the network below Figure2, we implement the proposed LCL method at the input end of the network. Figure2 depicts unique domain distributions, denoted by different colors, and distinct classes, represented by various shapes.

category-level distance, respectively. Based on these distances, we strictly sort the source domain data from easy to difficult to alleviate training instability caused by conflicting gradient directions during the optimization process.

This study provides the following contributions:

(1) We introduce an innovative and advanced curriculum learning-based training strategy in the domain generalization (DG) context, capitalizing on domain shift and category shift.

(2) LCL considers both inter-domain curriculum learning and intra-domain curriculum learning, not only focusing on the sorting of inter-domain data but also emphasizing the sorting of intra-domain data. This is a comprehensive application of the curriculum learning concept in domain generalization.

(3) LCL can be successfully applied to existing domain generalization methods to further improve generalization performance. Through experiments, it is verified that LCL can be successfully applied to multi-class datasets, and the proposed method has broad application prospects.

The structure of the remainder of this paper is as follows: Section II introduces related research; Section III discusses our proposed method in detail; Section IV reports the experimental results and provides a detailed analysis of these results; Section V summarizes the work of this paper.

II. RELATED WORK

The problem of generalizing predictive models from several source domains to an unseen target domain has been explored to some extent in the fields of machine learning and computer vision [12], [13]. Muandet et al. [14] formally introduce the term "domain generalization" to refer to this problem and enhance a classifier trained on source domains for use with an unseen target domain by proposing the Domain-Invariant

Component Analysis (DICA) approach. Specifically, DICA identifies a feature transformation by minimizing the distributional variance among various source marginal distributions while preserving the functional relationship between input and output variables. Subsequently, aligning the distributions of multiple source domains has emerged as a fundamental solution for addressing domain generalization [15], [16], [17], [18].

One of the most commonly utilized methods in domain generalization (DG) is domain-invariant representation learning. Mahajan et al. [19] elucidate the importance of modelling within-class variations and propose a matching-based algorithm for situations where base objects are observed, and an approximate objective when objects are not observed. Lv et al. [20] advances a cutting-edge method from a causal perspective to extract causal factors from inputs and reconstruct invariant causal mechanisms. Li et al. [21] explicitly address disentangled features on shape by refining network structures and augmentation techniques. Additionally, learning schemes achieve significant advancements from various angles with transferable architecture. Cha et al. [22] endeavor to discover flat minima to narrow the domain generalization gap by devising a strategy of stochastic weight averaging densely to mitigate overfitting. Zhang et al. [23] develop a multi-view regularized meta-learning algorithm that utilizes multiple optimization trajectories to determine optimal directions for model updates. Wang et al. [24] explore implicitly aligning the gradient directions between the perturbed loss and the empirical risk to enhance the optimization objective. Dai et al. [25] propose to conduct distribution exploration in the subset of uncertainty sharing the same semantic factors with the training domain. A special subset of methods based on representation learning are those grounded in causality, which introduce causal invariance. As an example, PAIR [26] learns causal invariance by introducing a multi-objective optimization approach to effectively balance empirical and invariant risk minimization.

In addition to the methods mentioned above, domain generalization can also be achieved through other approaches [27], [28], [29], [30], [31], [32], [33]. Ensemble methodsin deep learning have garnered attention due to their efficacy in enhancing model performance, robustness, and generalization. Bayesian ensemble method [27] offer a probabilistic framework for amalgamating predictions while accounting for model uncertainties. Ensemble distillation methods [28] aim to bridge the performance disparity between large and small models through techniques grounded in ensemble learning. Approaches hinging on meta-learning [29], [30], [31] acquire the ability to simulate domain shifts through the employment of an episode-based training paradigm. Besides, methods rooted in self-challenge, exemplified by RSC [32], compel the model to glean a comprehensive representation by discarding prevailing features elicited within the training dataset.Furthermore, Gao et al. [33] employ meta-learning to discover a reusable white-box loss function, employing the Implicit Function Theorem (IFT) to compute gradients of the target domain performance concerning the source domain loss parameters.

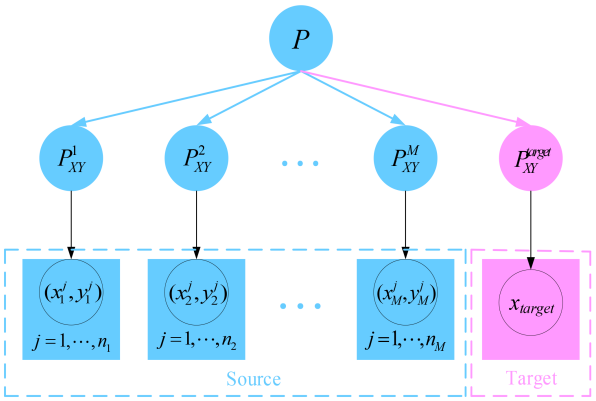

FIGURE 3. Principles of domain generalization.

III. METHODOLOGY

A. NOTATIONS

Consider $\mathcal{X}$ as representing a set of input elements, where $\mathcal{Y}$ designates the output space. Within a given domain, datasets are drawn from a distribution. This phenomenon is denoted as $S = \{(x_i, y_i)\}_{i=1}^{n} \sim P_{XY}$, wherein $x \in \mathcal{X} \subset \mathbb{R}^d$, $y \in \mathcal{Y} \subset \mathbb{R}$ represents the label, while P_{XY} signifies the joint distribution of the input sample and output label. The corresponding random variables are represented by X and Y [34].

Illustrated in Figure 3, within the context of domain generalization, we are presented with a collection of M source domains denoted as $S_{source} = \{S^i \mid i = 1, \cdots, M\}$, where $S^i = \{(x_j^i, y_j^i)\}_{j=1}^{n_i}$ represents the i-th specific domain. The difference in the joint distributions exists between every pair of domains, denoted as: $P^i_{XY} \neq P^j_{XY}$, $1 \leq i \neq j \leq M$. The primary objective underlying domain generalization pertains to the acquisition of a robust and broadly applicable predictive function denoted as $h : \mathcal{X} \to \mathcal{Y}$ [34]. This function is to be learned from the training data originating from the M source domains, with the ultimate aim of minimizing predictive error when applied to an entirely novel target domain, S_{target} (It is important to note that S_{target} cannot be accessed in training and $P^{target}_{XY} \neq P^i_{XY}$ *for* $i \in \{1, \cdots M\}$) :

$$\min_h \mathbb{E}_{(x,y)\in S_{target}}[\ell(h(x), y)$$

where $\mathbb{E}$ is the expectation and $\ell(\cdot, \cdot)$ is the loss function.

B. GENERIC NETWORK STRUCTURE

We propose a universal network architecture, the network structure utilized in this research is depicted in Figure 2. This network structure can be applied to current mainstream domain generalization methods, such as representation learning, learning strategy, and so on. The proposed generic network structure consists of three key elements: the LCL method, the feature extractor, and the classifier. LCL is used to feed data in stages, and then DG methods are applied

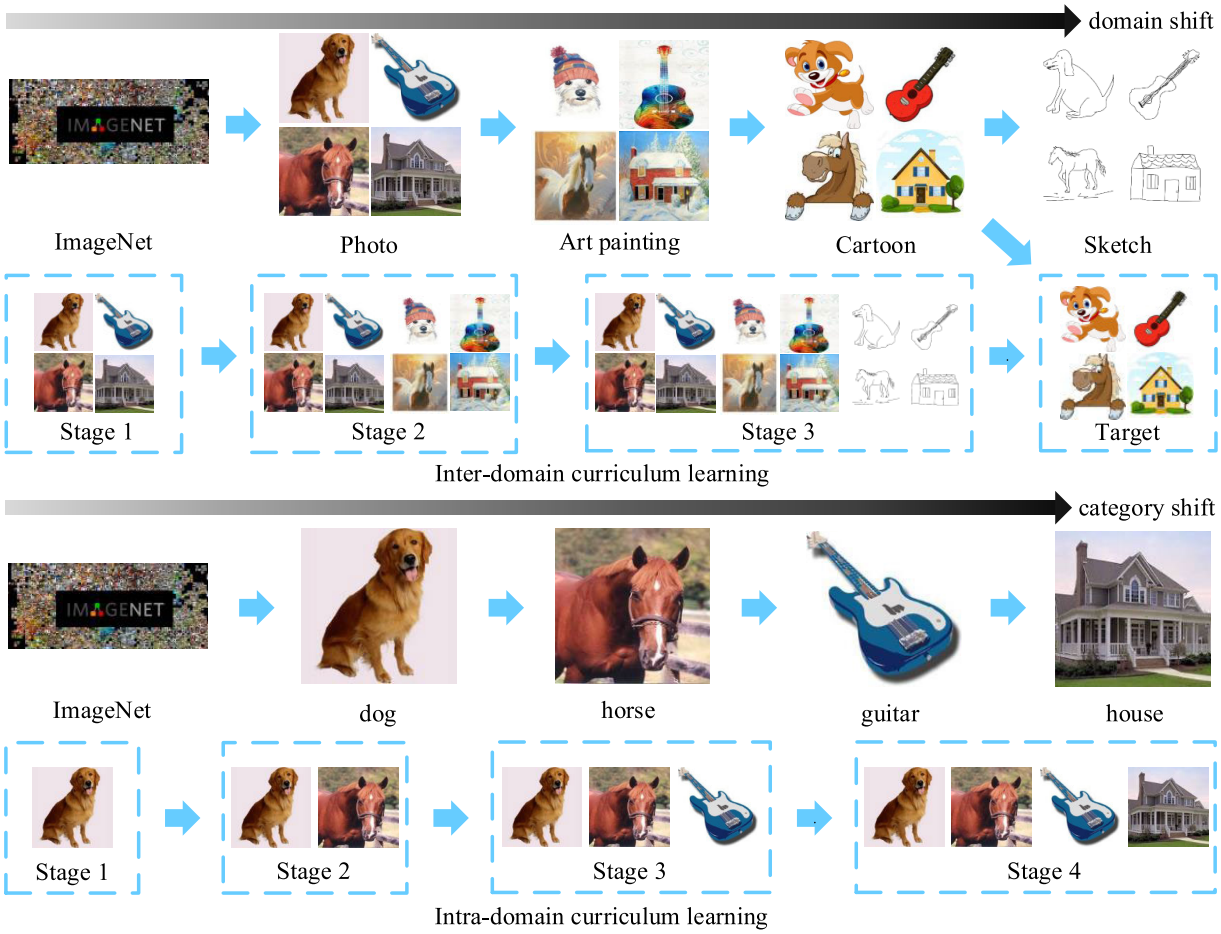

FIGURE 4. Comparison of inter-domain curriculum learning and intra-domain curriculum learning. Inter-domain curriculum learning focuses on sorting the inter-domain data as a whole from easy to hard, while intra-domain curriculum learning focuses on sorting the intra-domain category data from easy to hard.

to the embedded data extracted by the feature extractor to facilitate the establishment of a correctly aligned embedding space. Finally, these acquired embeddings are input into the classifier, which adeptly discriminates target images by ensuring robust alignment among classes within the learned embeddings.

C. LADDER CURRICULUM LEARNING

In DG, we find that almost all methods use random data for training, however, this approach can produce conflicting gradients. Therefore, we propose the LCL method as an innovative strategy to be implemented at the input end. Specifically, the source domain data needs to be sorted through inter-domain curriculum learning and intra-domain curriculum learning, and then each stage's data is sequentially fed into the network for training.

Based on previous research findings, it has been observed that as the domain shift from the specific photo domain becomes stronger, the model's test accuracy decreases. Specifically, in the PACS dataset, the test accuracy is relatively high for photos, followed by art paintings, cartoons, and sketches. Researchers commonly adopt pre-trained weights from ImageNet for domain generalization (DG) tasks instead of training models from scratch. As a result, the model exhibits bias towards the photo domain even before training, leading to a degradation in generalization performance. This decline is particularly evident in datasets where domain shift is prominently represented, such as PACS, which has been actively utilized in recent DG studies.

In light of this, we hypothesize that including training data with strong domain shift during the initial training phase might hinder successful learning due to difficulties with weight exploration. Therefore, we consider inter-domain curriculum learning and intra-domaincurriculum learning. Inter-domain curriculum learning provides data with weak domain shift from ImageNet during the initial training to train the model, and then gradually provides all domain data. For example, in the PACS dataset, assuming the target domain is cartoons. As shown in Figure 4 (top), the model first learns photos, then photos and art paintings, and finally photos, art paintings, and sketches. Intra-domain curriculum learning focuses on learning category data from easy to difficult within a specific domain. As shown in Figure 4 (bottom), during the initial training, data with weak category shift from ImageNet

is used to train the model, and then all category data is gradually provided. As shown in Figure 4 (bottom), the model first learns dogs, then dogs and horses, then dogs, horses, and guitars, and finally dogs, horses, guitars, and houses. It is noteworthy that we use cosine similarity to measure the distance between the data from the source domains and the ImageNet data, as shown in Equation (1).

$$cos = \frac{\boldsymbol{a} \cdot \boldsymbol{b}}{\|\boldsymbol{a}\| \, \|\boldsymbol{b}\|} \tag{1}$$

Here, "$\cdot$" denotes the dot product, and "$\|a\|$ " represents the magnitude of the vectors. The range of cosine similarity is $[-1, 1]$, with larger values indicating a smaller angle between the two vectors, meaning the vectors are more similar. Based on these distances, we strictly sort the source domain data from easy to difficult to alleviate training instability caused by conflicting gradient directions during the optimization process.

The process of data input is described as follows. Assuming that D_{all}, as defined in Equation (2), representing a dataset comprising all domains. The sequence of $\{D_1, D_2, \cdots, D_M\}$ denotes the progression in which domain shifts intensify. Additionally, D_i, $1 \leq i \leq M$ in Equation (3) to Equation (5) denotes a domain encompassing all categories, and the sequence of $\{A, B, \cdots, Z\}$ signifies the increasing strength of category shift.

$$D_{all} = \{D_1, D_2, \cdots, D_M\} \tag{2}$$
$$D_1 = \{A_1, B_1, \cdots, Z_1\} \tag{3}$$
$$D_2 = \{A_2, B_2, \cdots, Z_2\} \tag{4}$$
$$\cdots$$
$$D_M = \{A_M, B_M, \cdots, Z_M\} \tag{5}$$

Assuming that the target domain is D_M. As shown in Equations (6), (7), and (8), we progressively incorporate class data in the order from weak to strong shift, while excluding the target domain D_M.

$$S_1 = \{A_1\} \tag{6}$$
$$S_2 = \{S_1, B_1\} \tag{7}$$
$$\cdots$$
$$S_n = \{S_{n-1}, Z_{M-1}\} \tag{8}$$

In the LCL framework, the domains utilized for training are arranged in ascending order based on their domain shift. Typically, source domains can be organized by their distance from a domain where a backbone is pre-trained. Likewise, the categories employed for training are arranged in ascending order according to their category shift within the LCL approach. These categories can be ordered based on their distance from a domain where a backbone is pre-trained. Specifically, we use metric methods to calculate domain-level distance and class-level distance. Based on these two distances, we carefully sort the source domain data from easy to difficult. We firmly believe that the LCL method is still effective even when applied to complex datasets.

D. OVERALL OBJECTIVE AND TRAINING

In order to facilitate the classification task in DG, we employ the cross-entropy loss. This mathematical expression, as depicted in Equation (9), represents the definition of the classification loss.

$$L_c = \frac{1}{n_s} \sum_{j=1}^{n_s} L(f(g(x^j)), y^j) \tag{9}$$

where $L(\cdot,\cdot)$ is the cross-entropy loss, $g(\cdot)$ is the feature extractor, and a softmax over the K classes comes after the classifier $f(\cdot)$. n_s represents the number of source domain images.

Assuming the loss function of a certain domain generalization method is denoted as L_{dg}, the overall objective function is shown as Equation (10).

$$L_{all} = L_c + \alpha L_{dg} \tag{10}$$

We integrate two distinct loss functions: the classification loss and the domain generalization loss. The relative significance of these functions is controlled by the hyperparameters α, which dictate the weight of the domain generalization loss.

IV. EXPERIMENTS

We carry out experiments on three well-established benchmark datasets that are widely employed in the domain generalization research. Firstly, we provide an overview of the datasets used and elaborate on the implementation specifics. Subsequently, we conduct a comprehensive ablation study of the proposed method. Finally, we further evaluate the effectiveness of LCL in domain generalization tasks through its application in state-of-the-art domain generalization methods.

A. DATASETS

Figure 5 illustrates selected sample images from the three benchmark datasets utilized in our experiments.

PACS [3] dataset comprises four domains: Photo, Art Painting (Art), Cartoon, and Sketch. It encompasses a total of 9,991 images belonging to seven classes, namely dog, elephant, giraffe, guitar, horse, house, and person.

Office-31 [35] dataset comprises 4,652 images distributed across 31 categories. These images have been collected from three distinct sources: Amazon, Webcam, and DSLR. DSLR contains high-quality images captured using a digital SLR camera. Amazon includes medium-resolution photos obtained from internet retailers (www.amazon.com). Webcam consists of low-resolution images captured using a web camera. Our proposed method is evaluated on six transfer tasks within the Office-31 dataset, namely A $\rightarrow$ W, D $\rightarrow$ W, W $\rightarrow$ D, A $\rightarrow$ D, D $\rightarrow$ A, and W $\rightarrow$ A.

Office-Home [36] dataset consists of approximately 15,500 images, which are categorized into 65 object classes commonly observed in office and home settings. It comprises four source domains, namely Art, Clipart, Product, and Real World. Notably, the Office-Home dataset boasts the largest number of categories among the three benchmark datasets.

TABLE 1. Classification accuracy of various methods on PACS dataset. The best results are highlighted in bold font. In this table, "Basel." Represents the basel. method trained solely on the source dataset, "$IDCL_{2STAGE}$" and "$IDCL_{3STAGE}$" are both methods proposed in the literature [11].

Network	Method	Photo	Art	Cartoon	Sketch	Average
AlexNet	Basel.	88.4	67.0	69.3	59.3	71.0
	Basel. + $IDCL_{2stage}$	88.7	67.4	70.1	60.1	71.6
	Basel. + $IDCL_{3stage}$	**88.8**	67.5	70.7	60.5	71.9
	Basel. + LCL	88.7	**67.8**	**71.0**	**61.1**	**72.2**
ResNet18	Basel.	96.4	74.3	76.7	68.7	79.0
	Basel. + $IDCL_{2stage}$	96.6	74.8	77.5	69.6	79.6
	Basel. + $IDCL_{3stage}$	**96.8**	74.9	78.3	70.1	80.0
	Basel. + LCL	**96.8**	**75.5**	**78.6**	**70.5**	**80.4**

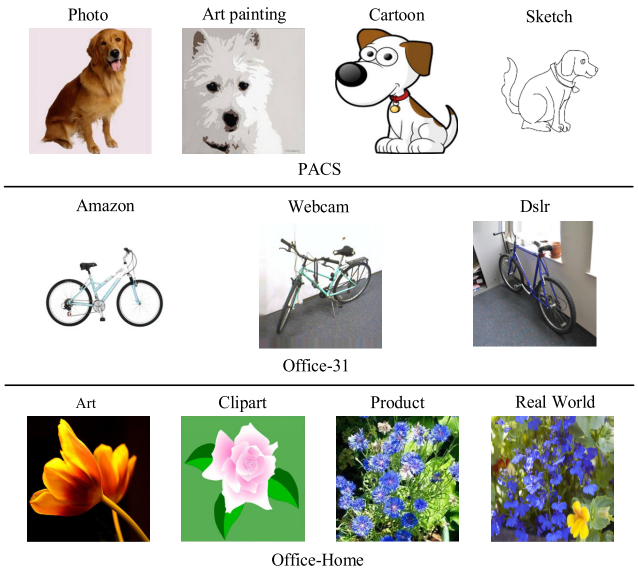

FIGURE 5. The gallery of datasets.

B. PARAMETER SETTING

Our approach is implemented in the PyTorch framework, utilizing the AlexNet [37] architecture or the ResNet [38] architecture as the underlying network. The network is initially pre-trained on the ImageNet dataset. Stochastic gradient descent (SGD) is employed as the optimizer for network training. Following IDCL [11], data augmentation is used in our experiments to improve model generalizability. This is done by randomly cropping, flipping horizontally, jittering color, and changing the intensity. The learning rate gradually decreases, starting from 0.001. The batch size is set to 64. Batch data has two sources: newly added class data and class data that has already participated in the training. We divide the newly added class data and the class data that has already participated in the training in a 1:1 ratio. The number of iterations for the first input category data is set to p, and the number of iterations is increased by p for each category added thereafter. Due to the varying number of categories in the PACS, Office-31, and Office-Home datasets, the iteration counts are set differently for each dataset. The p-values are set to 90, 10, and 3 in the PACS, Office-31, and Office-Home datasets, respectively. The implementation process follows a leave-one-domain-out protocol, i.e., all but one domain dataset is treated as the target domain and the rest of the domain datasets are treated as source domains.

C. ANALYSIS OF EFFECTIVENESS

1) METHODS

This section aims to assess the effectiveness of the proposed method. To conduct this evaluation, three benchmark datasets are employed, namely PACS, Office-31, and Office-Home. For the PACS dataset and the Office-31 dataset, all methods use the AlexNet [37] network and ResNet18 [38] network. Moreover, the scope of this comparison is broadened to encompass the Office-Home datasets, where we use ResNet18 [38] and ResNet50 [38] as the underlying network model. In DG, there are relatively few studies on curriculum learning at present. We only find the $IDCL_{2stage}$ method [11] and the $IDCL_{3stage}$ method [11]. Both $IDCL_{2stage}$ and $IDCL_{3stage}$ are methods of inter-domain curriculum learning, but the two methods differ in how they input data. $IDCL_{3stage}$ inputs data from all stages, while $IDCL_{2stage}$ only inputs data from some stages. It is worth noting that $IDCL_{3stage}$ is designed for datasets with three source domains, but the Office-31 dataset only has two source domain data. Therefore, in the Office-31 dataset, we refer to the method in literature [11] as $IDCL_{2stage}$. By applying the LCL method on various datasets and comparing it with the IDCL method, we aim to comprehensively evaluate the effectiveness and performance of LCL on different datasets and network architectures.

Table 1 presents the outcomes achieved on the PACS dataset, while Tables 2 and 3 display the results on Office-31 and Office-Home, respectively. Throughout these tables, we adhere to the leave-one-domain-out evaluation protocol. Notably, in each column of the tables, the most favorable outcome is denoted in bold formatting.

2) ANALYSIS OF THE PACS DATASET

We conduct a comprehensive comparison between our proposed method, LCL, and the state-of-the-art approaches. Table 1 shows the recognition accuracy in the PACS

TABLE 2. Classification accuracy of various methods on Office-31 dataset. The best results are highlighted in bold font. In this table, "Basel." represents the basel. Method trained solely on the source dataset, "$IDCL_{2STAGE}$" is a methods proposed in the literature [11].

Network	Method	Amazon	Webcam	Dslr	Average
AlexNet	Basel.	43.8	88.4	94.1	75.4
	Basel. + $IDCL_{2stage}$	44.4	89.2	94.5	76.1
	Basel. + LCL	**45.3**	**89.7**	**94.8**	**76.6**
ResNet18	Basel.	55.1	92.6	99.0	82.2
	Basel. + $IDCL_{2stage}$	55.7	93.4	99.3	82.8
	Basel. + LCL	**56.7**	**93.9**	**99.5**	**83.4**

TABLE 3. Classification accuracy of various methods on office-home dataset. The best results are highlighted in bold font. In this table, "Basel." represents the basel. Method trained solely on the source dataset, "$IDCL_{2STAGE}$" and "$IDCL_{3STAGE}$" are both methods proposed in the literature [11].

Network	Method	Art	Clipart	Product	Real-world	Average
ResNet18	Basel.	58.8	48.3	74.2	76.2	64.4
	Basel. + $IDCL_{2stage}$	59.0	48.3	74.6	76.5	64.6
	Basel. + $IDCL_{3stage}$	58.9	48.6	75.1	76.7	64.8
	Basel. + LCL	**59.6**	**49.7**	**76.0**	**77.7**	**65.8**
ResNet50	Basel.	65.0	58.8	78.2	79.3	70.3
	Basel. + $IDCL_{2stage}$	65.3	58.8	78.5	79.7	70.6
	Basel. + $IDCL_{3stage}$	65.2	59.2	79.0	79.9	70.8
	Basel. + LCL	**66.3**	**60.4**	**79.8**	**80.8**	**71.8**

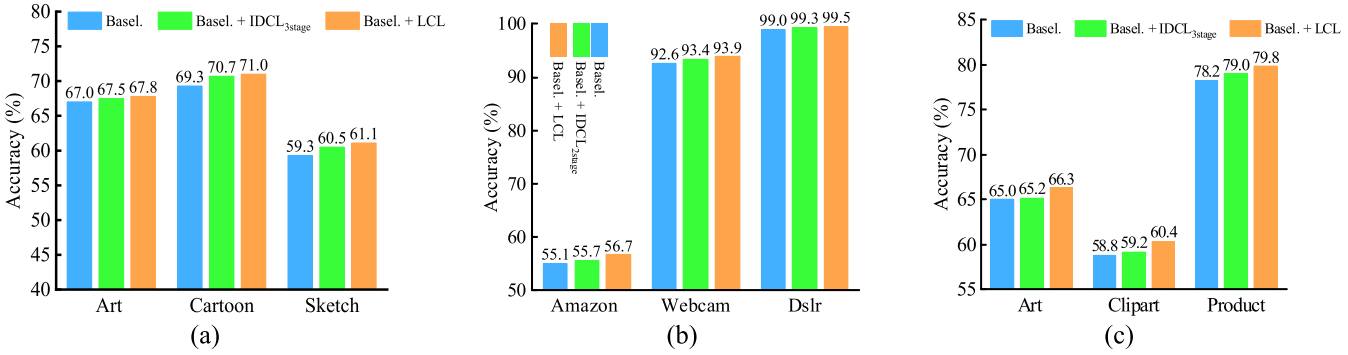

FIGURE 6. Performance of various methods on various datasets and various networks. (a): performance of various methods on PACS dataset using the AlexNet network. (b): performance of various methods on Office-31 dataset using the ResNet18 network. (c): performance of various methods on Office-Home dataset using the ResNet50 network.

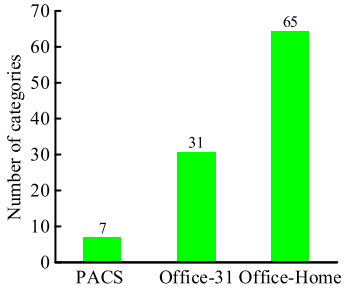

FIGURE 7. Number of categories.

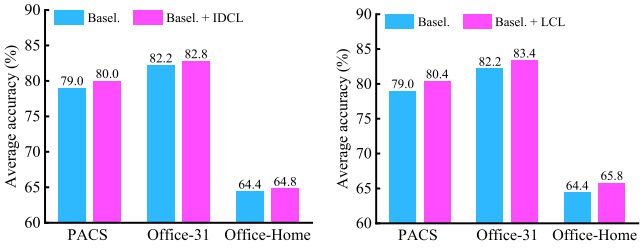

FIGURE 8. Comparison of the average accuracy of various methods.

benchmark. The Basel. is a network modified from AlexNet or ResNet18, without using any domain generalization methods.

In AlexNet, we find that both $IDCL_{2stage}$ and $IDCL_{3stage}$ significantly improve the "Basel." method, with average accuracy increased by 0.6% and 0.9% respectively. This indicates that a reasonable arrangement of data in domain generalization is essential. Moreover, we observe that "Basel. + $IDCL_{3stage}$" outperforms "Basel. +$IDCL_{2stage}$" in both task-specific accuracy and average accuracy, indicating that

input data from all stages is more effective than using data from only some stages. Our proposed LCL method is inspired by this, and it inputs data from all stages in order. Furthermore, we observe that "Basel. + LCL" significantly improves the average accuracy of "Basel." by 1.2%, demonstrating its capability in enhancing generalization. Similar conclusions are drawn in ResNet18.

3) ANALYSIS OF THE OFFICE-31 DATASET

Based on the experimental data presented in Table 2, we find that in AlexNet, "Basel. + LCL" achieves a significant improvement in the accuracy of "Basel." by 1.2%, thus demonstrating its effectiveness in enhancing generalization. Furthermore, "Basel. + LCL" significantly improves "Basel." in all tasks. It is worth noting that we obtain similar conclusions in ResNet18, indicating that LCL can effectively enhance network generalization capability, whether in AlexNet or ResNet18. This means that the LCL method can be applied to different networks and has broad application prospects.

4) ANALYSIS OF THE OFFICE-HOME DATASET

Table 3 provides a comprehensive comparison of our proposed method with several state-of-the-art approaches. In ResNet18, the experimental results clearly demonstrate that both "Basel. + LCL" and "Basel. + $IDCL_{3stage}$" outperform "Basel." in terms of average accuracy. This indicates that reasonably arranging data in each stage can significantly improve generalization capability compared to random data input. Furthermore, we apply the LCL method to the deeper ResNet50 and find that "Basel. + LCL" significantly improves "Basel." in all tasks. This shows that even in deeper networks, LCL can enhance the network's generalization ability, further demonstrating the applicability of LCL in different classification networks. Moreover, "Basel. + LCL" achieves higher accuracy across all tasks than "Basel. + $IDCL_{3stage}$". This is because our proposed method not only focuses on inter-domain curriculum learning but also carefully considers intra-domain curriculum learning, thereby mitigating the generation of conflicting gradients as much as possible.

5) FURTHER ANALYSIS

Based on the data in Table 1 to Table 3, we present the accuracy of three tasks on three different datasets in Figure 6. From Figure 6, it can be observed that whether using a shallow network such as AlexNet or a deep network like ResNet18 and ResNet50, "Basel. + LCL" significantly improves the accuracy of "Basel." on all tasks. This demonstrates the applicability of our proposed LCL method in different networks and datasets.

Figure 7 shows the number of classes contained in each domain in the PACS dataset, Office-31 dataset, and Office-Home dataset. We find that each domain in the PACS dataset has relatively few classes, only 7 classes. However, in the Office-31 dataset and Office-Home dataset, each domain contains a relatively large number of classes, exceeding 30 classes. There is a significant difference in the number of classes among the datasets. Based on this, we compare the average accuracy of "Basel. + IDCL" and "Basel. + LCL" on different datasets to reflect their overall performance, as shown in Figure 8 (left) and Figure 8 (right). Please note that "Basel. + IDCL" here refers to the input of data from all stages, representing either "Basel. + IDCL2stage" or "Basel. + IDCL3stage". From Figure 8 (left), it can be observed that compared to "Basel.", "Basel. + IDCL" shows a significant improvement in the PACS dataset, reaching 1.0%. However, in the Office-31 dataset and Office-Home dataset, the improvement of "Basel. + IDCL" is not as significant, only around 0.5%. This indicates that IDCL performs better in datasets with fewer classes but only moderately in datasets with more classes. We believe this is because the IDCL method does not consider the problem of intra-domain class shift. Specifically, in the PACS dataset, each domain has fewer classes, and using the IDCL method is less likely to encounter the problem of conflicting gradients. Therefore, the improvement of "Basel. + IDCL" is more evident in the PACS dataset. However, in the Office-31 dataset and Office-Home dataset, there are more classes, making it more prone to the issue of conflicting gradients when using the IDCL method, leading to less prominent improvement in "Basel. + IDCL" in these datasets. In contrast, from Figure 8 (right), it can be observed that "Basel. + LCL" significantly improves the accuracy of "Basel." on all datasets. This indicates that our proposed LCL method can be successfully applied to datasets with either fewer or more classes.

D. COMPARATIVE ANALYSIS OF THE LATEST METHODS

1) METHODS

In this section, we perform a comprehensive comparison to evaluate the effectiveness of LCL applied in state-of-the-art domain generalization methods. We use a representative benchmark dataset: PACS. We use the ResNet18 network architecture. We apply two domain generalization methods: DSU [39]and DSU++ [40]. To further validate the superiority of LCL, we incorporate the concept of LCL into the DSU method and DSU++ method.Through this evaluation, we aim to assess the effectiveness of LCL implementing in domain generalization methods.

In the DSU method, we implement our proposed approach and report the experimental results in Table 4. Likewise, in the DSU++ method, we implement our proposed approach and report the experimental results in Table 5. In each table, the names of the source domains are omitted according to the leave-one-domain-out evaluation protocol.

2) COMPARATIVE ANALYSIS IN DSU METHOD

DSU is an advanced domain generalization methods proposed by Li et al. In the DSU method, we incorporate LCL to verify its effectiveness. Table 4 presents the experimental results of our approach on the PACS dataset. In the

TABLE 4. Classification accuracy of various methods on PACS dataset. The best results are highlighted in bold font.

Network	Method	Photo	Art	Cartoon	Sketch	Average
ResNet18	DSU [39]	95.8	83.6	79.6	77.6	84.2
	DSU [39] + LCL	**96.8**	**84.9**	**80.7**	**79.1**	**85.4**

TABLE 5. Classification accuracy of various methods on PACS dataset. The best results are highlighted in bold font.

Network	Method	Photo	Art	Cartoon	Sketch	Average
ResNet18	DSU++[40]	96.4	84.5	80.6	79.2	85.2
	DSU++[40] + LCL	**97.2**	**85.3**	**81.2**	**80.1**	**86.0**

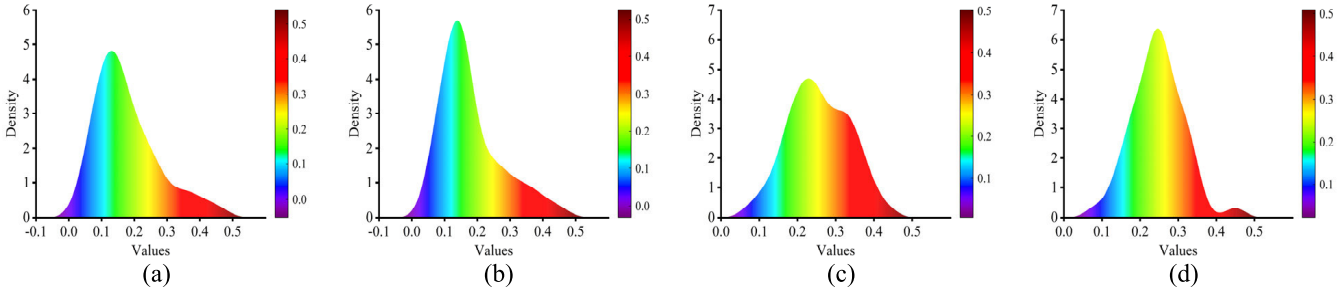

FIGURE 9. Comparison of the shifts of feature statistics (mean and standard deviation) in source domains and target domain using the DSU method. (a): feature mean value of source domains. (b): feature mean value of target domain. (c): feature standard deviation of source domains. (d): feature standard deviation of the target domain.

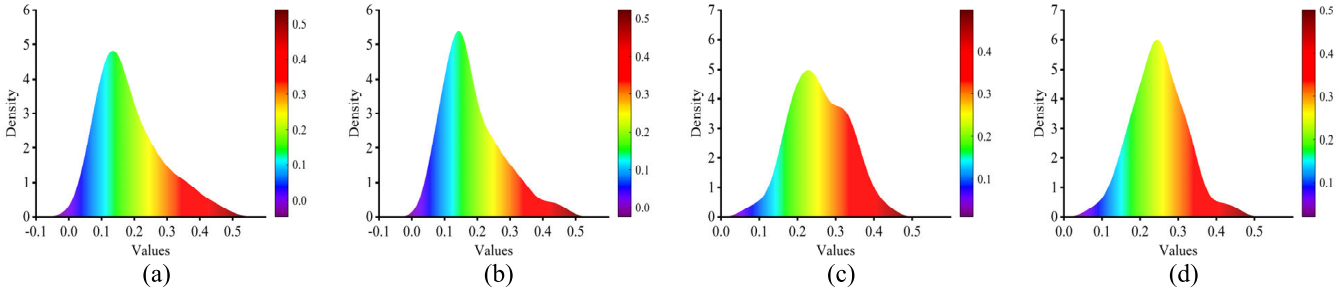

FIGURE 10. Comparison of the shifts of feature statistics (mean and standard deviation) in source domains and target domain using the "DSU + LCL" method. (a): feature mean value of source domains. (b): feature mean value of target domain. (c): feature standard deviation of source domains. (d): feature standard deviation of the target domain.

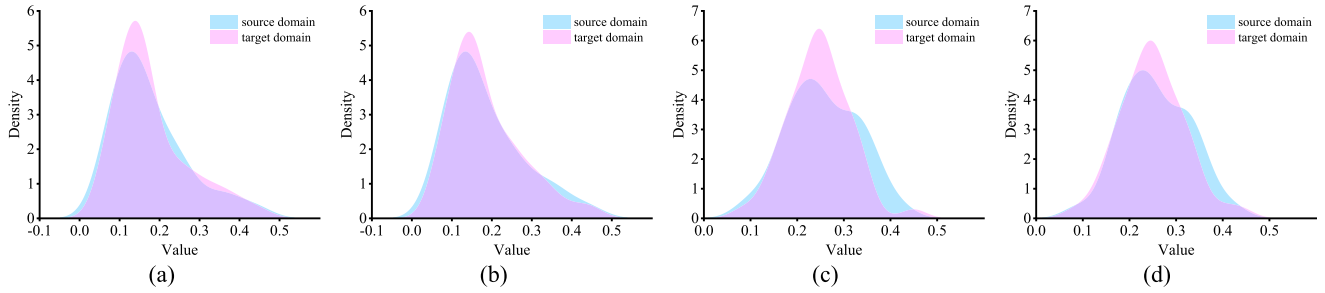

FIGURE 11. Comparison of the DSU method and "DSU + LCL" method in terms of feature statistics (mean and standard deviation) (a): feature mean value of source domains and target domain using the DSU method. (b): feature mean value of source domains and target domain using the "DSU + LCL" method. (c): feature standard deviation of source domains and target domain using the DSU method. (d): feature standard deviation of source domains and target domain using the "DSU + LCL" method.

PACS dataset, LCL significantly improves the DSU method. "DSU + LCL" achieves an average accuracy improvement of 1.2% over "DSU." This demonstrates that a reasonable arrangement of data in the domain generalization method can effectively enhance the network's generalization ability.

3) COMPARATIVE ANALYSIS IN DSU++ METHOD

To further validate the effectiveness of LCL in other domain generalization methods, we incorporate LCL into the advanced domain generalization method DSU++ proposed by Li et al. Table 5 present the experimental results of our

approach on the PACS dataset. Similar to the conclusions in Table 4, "DSU++ + LCL" also significantly improves the DSU++ method, achieving better results than "DSU++" on all tasks, with an average accuracy increase of 0.8%. These results indicate that our proposed method can be successfully applied to existing domain generalization methods.

4) FURTHER ANALYSIS

Within this subsection, we delve into an extensive examination of the impacts generated by the DSU method and the "DSU + LCL" method on intermediate features. For analysis purposes, we conduct quantitative experiments on the PACS dataset, where we treat art painting as the unseen target domain and the remaining domains as source domains.

To elucidate the phenomenon of feature statistical shift, we capture the intermediate features after the second block in ResNet18 and calculate the average feature statistics values for some data in a specific category within both the source domain and the target domain. The distributions of these feature statistics are visually presented in Figure 9 and Figure 10. We find that when using the DSU method, the feature statistics demonstrate an obvious implicit calibration effect, indicating that the DSU method can to some extent improve the differences between feature statistics, but there are still differences between domain distributions. Therefore, we add our proposed LCL method on top of the DSU method. From Figure 10, it can be observed that "DSU + LCL" has a smaller distribution shift compared to DSU. This is because LCL not only focuses on domain shift issues but also considers category shift issues within the domain. LCL improves the training instability caused by conflicting gradients from two aspects. As a result, "DSU + LCL" can help the model obtain robustness to domain shift and category shift. To visually compare the source domain distribution with the target domain distribution, we present the feature statistics of Figure 9 and Figure 10 together in Figure 11. It can be observed from Figure 11 that "DSU + LCL" can further improve the distribution shift issue present in DSU. This once again strongly confirms that the LCL method can be successfully applied to existing domain generalization methods.

V. CONCLUSION

The primary goal of this study is to address the issue of training instability caused by conflicting gradient directions. We propose a method called "Ladder Curriculum Learning" (LCL) to promote domain generalization. The integration of the LCL method yields substantial enhancements in managing the direction of conflicting gradients, a pivotal aspect for bolstering domain generalization, ultimately leading to heightened accuracy on the target dataset. Furthermore, LCL demonstrates its ability to harmoniously complement existing domain generalization approaches. Through the concurrent application of these two approaches, the models attain heightened robustness when faced with diverse domain data. Empirical evaluations on benchmark datasets unequivocally validate the efficacy of our proposed method in advancing the network's generalization capabilities. Although LCL is a successful method for resolving conflicting gradients, some domain generalization methods may not yet fully apply the idea of LCL. We will address this issue in future research.

CONFLICT OF INTEREST STATEMENT

We have no conflicts of interest to declare.

ACKNOWLEDGMENT

This work was supported by the Huzhou Science and Technology Program Project under Grant 2022GZ19.

REFERENCES

[1] Z. Chen, W. Wang, Z. Zhao, F. Su, A. Men, and Y. Dong, "Instance paradigm contrastive learning for domain generalization," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 34, no. 2, pp. 1032–1042, Jul. 2023.
[2] S. Choi, S. Jung, H. Yun, J. T. Kim, S. Kim, and J. Choo, "RobustNet: Improving domain generalization in urban-scene segmentation via instance selective whitening," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 11575–11585.
[3] D. Li, Y. Yang, Y.-Z. Song, and T. M. Hospedales, "Deeper, broader and artier domain generalization," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Venice, Italy, Oct. 2017, pp. 5543–5551.
[4] K. Wu, L. Li, and Y. Han, "Weighted progressive alignment for multi-source domain adaptation," *Multimedia Syst.*, vol. 29, no. 1, pp. 117–128, Feb. 2023.
[5] J. Yang, Z. Li, C. Li, S. Xie, W. Yu, and S. Li, "Generalizing to unseen domains via PatchMix," *Multimedia Syst.*, vol. 30, no. 1, pp. 1–20, Feb. 2024.
[6] W. Peng, H. Chen, Y. Li, and J. Sun, "Multi-source domain generalization peron re-identification with knowledge accumulation and distribution enhancement," *Int. J. Speech Technol.*, vol. 54, no. 2, pp. 1818–1830, Jan. 2024.
[7] S. Chen, L. Wang, Z. Hong, and X. Yang, "Domain generalization by joint-product distribution alignment," *Pattern Recognit.*, vol. 134, Feb. 2023, Art. no. 109086.
[8] L. Ren, T. Mo, and X. Cheng, "Meta-learning based domain generalization framework for fault diagnosis with gradient aligning and semantic matching," *IEEE Trans. Ind. Informat.*, vol. 20, no. 21, pp. 1–11, May 2023.
[9] J. Deng, W. Dong, R. Socher, L. J. Li, K. Li, and L. Fei-Fei, "ImageNet: A large-scale hierarchical image database," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, 2009, pp. 1–8.
[10] F. Liu, T. Zhang, C. Zhang, L. Liu, L. Wang, and B. Liu, "A review of the evaluation system for curriculum learning," *Electronics*, vol. 12, no. 7, p. 1676, Apr. 2023.
[11] D. Kim, J. Kim, and J. Lee, "Inter-domain curriculum learning for domain generalization," *ICT Exp.*, vol. 8, no. 2, pp. 225–229, Jun. 2022.
[12] Z. Guan, Y. Li, Z. Pan, Y. Liu, and Z. Xue, "RFDG: Reinforcement federated domain generalization," *IEEE Trans. Knowl. Data Eng.*, vol. 36, no. 3, pp. 1285–1298, Mar. 2024.
[13] D. Wang, J. Yang, W. Cui, L. Xie, and S. Sun, "AirFi: Empowering WiFi-based passive human gesture recognition to unseen environment via domain generalization," *IEEE Trans. Mobile Comput.*, vol. 23, no. 2, pp. 1–12, Jun. 2022.
[14] K. Muandet, D. Balduzzi, and B. Scholkopf, "Domain generalization via invariant feature representation," in *Proc. Int. Conf. Mach. Learn.*, 2013, pp. 10–18.
[15] S. Gao, Z. Zhang, and S. Su, "DAWN: Domain generalization based network alignment," *IEEE Trans. Big Data*, vol. 9, no. 3, pp. 1–11, Oct. 2022.
[16] B. Lyu, T. Nguyen, P. Ishwar, M. Scheutz, and S. Aeron, "Barycentric-alignment and reconstruction loss minimization for domain generalization," *IEEE Access*, vol. 11, pp. 49226–49240, 2023.
[17] Q. Qian, J. Luo, and Y. Qin, "Adaptive intermediate class-wise distribution alignment: A universal domain adaptation and generalization method for machine fault diagnosis," *IEEE Trans. Neural Netw. Learn. Syst.*, pp. 1–15, Mar. 2024.
[18] Z. Gao, B. Pan, X. Xu, T. Li, and Z. Shi, "LiCa: Label-indicate-conditional-alignment domain generalization for pixel-wise hyperspectral imagery classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 61, Sep. 2023, Art. no. 5519011.

[19] D. Mahajan, S. Tople, and A. Sharma, "Domain generalization using causal matching," in *Proc. Int. Conf. Mach. Learn. (PMLR)*, 2021, pp. 7313–7324.
[20] F. Lv, J. Liang, S. Li, B. Zang, C. H. Liu, Z. Wang, and D. Liu, "Causality inspired representation learning for domain generalization," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 8036–8046.
[21] J. Li, Y. Li, H. Wang, C. Liu, and J. Tan, "Exploring explicitly disentangled features for domain generalization," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 33, no. 11, pp. 6360–6373, Nov. 2023.
[22] J. Cha, S. Chun, K. Lee, H. C. Cho, S. Park, Y. Lee, and S. Park, "SWAD: Domain generalization by seeking flat minima," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 34, 2021, pp. 22405–22418.
[23] J. Zhang, L. Qi, Y. Shi, and Y. Gao, "MVDG: A unified multi-view framework for domain generalization," in *Proc. Eur. Conf. Comput. Vis.*, 2022, pp. 161–177.
[24] P. Wang, Z. Zhang, Z. Lei, and L. Zhang, "Sharpness-aware gradient matching for domain generalization," 2023, *arXiv:2303.10353*.
[25] R. Dai, Y. Zhang, Z. Fang, B. Han, and X. Tian, "Moderately distributional exploration for domain generalization," 2023, *arXiv:2304.13976*.
[26] Y. Chen, K. Zhou, Y. Bian, B. Xie, B. Wu, Y. Zhang, K. Ma, H. Yang, P. Zhao, B. Han, and J. Cheng, "Pareto invariant risk minimization: Towards mitigating the optimization dilemma in out of-distribution generalization," in *Proc. Int. Conf. Learn. Represent.*, 2023, pp. 1–50.
[27] X. Ren, Z. Mi, T. Cai, C. G. Nolte, and P. G. Georgopoulos, "Flexible Bayesian ensemble machine learning framework for predicting local ozone concentrations," *Environ. Sci. Technol.*, vol. 56, no. 7, pp. 3871–3883, Apr. 2022.
[28] Y. Chen, S. Wang, J. Liu, X. Xu, F. de Hoog, and Z. Huang, "Improved feature distillation via projector ensemble," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 35, 2022, pp. 12084–12095.
[29] K. Chen, D. Zhuang, and J. M. Chang, "Discriminative adversarial domain generalization with meta-learning based cross-domain validation," *Neurocomputing*, vol. 467, pp. 418–426, Jan. 2022.
[30] Y. Chen, Q. Tang, and H. Ma, "Multi-source adaptive meta-learning framework for domain generalization person re-identification," *Soft Comput.*, vol. 28, no. 6, pp. 4799–4820, Mar. 2024.
[31] Y. Dai, X. Li, J. Liu, Z. Tong, and L.-Y. Duan, "Generalizable person re-identification with relevance-aware mixture of experts," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 16140–16149.
[32] Z. Huang, H. Wang, E. P. Xing, and D. Huang, "Self-challenging improves cross-domain generalization," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2020, pp. 124–140.
[33] B. Gao, H. Gouk, Y. Yang, and T. Hospedales, "Loss function learning for domain generalization by implicit gradient," in *Proc. Int. Conf. Mach. Learn.*, 2022, pp. 7002–7016.
[34] J. Wang, C. Lan, C. Liu, Y. Ouyang, T. Qin, W. Lu, Y. Chen, W. Zeng, and P. Yu, "Generalizing to unseen domains: A survey on domain generalization," *IEEE Trans. Knowl. Data Eng.*, vol. 35, no. 8, pp. 8052–8072, Jun. 2022.
[35] K. Saenko, B. Kulis, M. Fritz, and T. Darrell, "Adapting visual category models to new domains," in *Proc. Eur. Conf. Comput. Vis.*, 2010, pp. 213–226.
[36] H. Venkateswara, J. Eusebio, S. Chakraborty, and S. Panchanathan, "Deep hashing network for unsupervised domain adaptation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 5385–5394.
[37] A. Krizhevsky, I. Sutskever, and G. Hinton, "ImageNet classification with deep convolutional neural networks," in *Proc. 25th Int. Conf. Neural Inf. Process. Syst.*, 2012, pp. 84–90.
[38] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 770–778.
[39] X. Li, Y. Dai, Y. Ge, J. Liu, Y. Shan, and L.-Y. Duan, "Uncertainty modeling for out-of-distribution generalization," 2022, *arXiv:2202.03958*.
[40] X. Li, Z. Hu, J. Liu, Y. Ge, Y. Dai, and L.-Y. Duan, "Modeling uncertain feature representation for domain generalization," 2023, *arXiv:2301.06442*.

XIAOSHUN WANG received the degree in pattern recognition and intelligent systems. He is currently a Lecturer. His research interests include domain generalization and domain adaptation.

SIBEI LUO received the Ph.D. degree. He is currently a Lecturer. He has achieved several significant contributions. His research interests include artificial intelligence and image processing.

YIMING GAO is currently pursuing the degree. His research interest includes artificial intelligence. He has received multiple awards.