International Journal of Computer Vision (2023) 131:552–571
https://doi.org/10.1007/s11263-022-01712-7

Domain-Specific Bias Filtering for Single Labeled Domain Generalization

Junkun Yuan[1] · Xu Ma[1] · Defang Chen[1] · Kun Kuang[1] · Fei Wu[1,2,3] · Lanfen Lin[1]

Received: 27 December 2021 / Accepted: 7 November 2022 / Published online: 24 November 2022

Abstract

Conventional Domain Generalization (CDG) utilizes multiple labeled source datasets to train a generalizable model for unseen target domains. However, due to expensive annotation costs, the requirements of labeling all the source data are hard to be met in real-world applications. In this paper, we investigate a Single Labeled Domain Generalization (SLDG) task with only one source domain being labeled, which is more practical and challenging than the CDG task. A major obstacle in the SLDG task is the discriminability-generalization bias: the discriminative information in the labeled source dataset may contain domain-specific bias, constraining the generalization of the trained model. To tackle this challenging task, we propose a novel framework called Domain-Specific Bias Filtering (DSBF), which initializes a discriminative model with the labeled source data and then filters out its domain-specific bias with the unlabeled source data for generalization improvement. We divide the filtering process into (1) feature extractor debiasing via k-means clustering-based semantic feature re-extraction and (2) classifier rectification through attention-guided semantic feature projection. DSBF unifies the exploration of the labeled and the unlabeled source data to enhance the discriminability and generalization of the trained model, resulting in a highly generalizable model. We further provide theoretical analysis to verify the proposed domain-specific bias filtering process. Extensive experiments on multiple datasets show the superior performance of DSBF in tackling both the challenging SLDG task and the CDG task.

Keywords Domain generalization · Visual recognition · Single labeled multi-source data · Bias filtering · Semantic feature projection

1 Introduction

Deep learning based *supervised learning* (SL) and *semi-supervised learning* (SSL) have made great progress in recent years (LeCun et al., 2015; Yang et al., 2021). However, their success heavily relies on the independent and identically distributed (i.i.d.) assumption (Vapnik, 1992), while the training (source) and test (target) datasets are usually sampled from different distributions in real-world applications, which is known as *dataset shift* (Quionero-Candela et al., 2009). To address this problem, *domain adaptation* (DA) (Ben-David et al., 2010) and *domain generalization* (DG) (Blanchard et al., 2011) are formulated and many effective methods (Wang et al., 2021; Lin et al., 2020; Xu et al., 2021; Wang et al.,

Communicated by Wanli Ouyang.

Junkun Yuan and Xu Ma have contributed equally to this work.

✉ Kun Kuang
kunkuang@zju.edu.cn

Junkun Yuan
yuanjk@zju.edu.cn

Xu Ma
maxu@zju.edu.cn

Defang Chen
defchern@zju.edu.cn

Fei Wu
wufei@zju.edu.cn

Lanfen Lin
llf@zju.edu.cn

[1] College of Computer Science and Technology, Zhejiang University, Hangzhou, China

[2] Shanghai Institute for Advanced Study of Zhejiang University, Shanghai, China

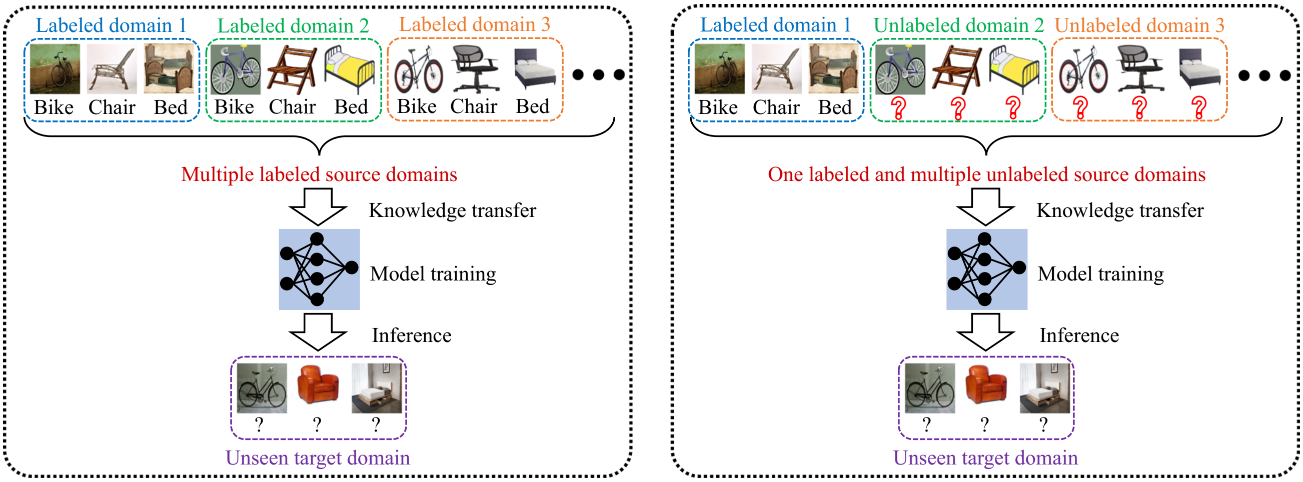
[3] Shanghai AI Laboratory, Shanghai, China

2020c; Ding & Fu, 2017) are proposed to improve the out-of-domain generalization ability of the model.

Typical research fields of DA, such as *unsupervised domain adaptation* (UDA) (Xu et al., 2021; Zhang et al., 2020b; Li et al., 2020b, c; Long et al., 2018; Zhang et al., 2019b), *multi-source domain adaptation* (MSDA) (Zuo et al., 2021; Peng et al., 2019; Zhang et al., 2015; Zhao et al., 2018; Wang et al., 2020c), and *multi-target domain adaptation* (MTDA) (Chen et al., 2019; Liu et al., 2020; Wang et al., 2020c; Gong et al., 2013; Yu et al., 2018; Gholami et al., 2020) suppose that both the source and the target datasets are available for model training. For each new target domain, they have to re-collect target data and use it to repeat the training process, which is expensive, time-consuming, or even infeasible. For example, an autonomous driving car can not know in advance which environment (i.e., domain), it will enter. DG is thus proposed to learn a generalizable model by incorporating the invariance across multiple labeled source domains without accessing any target data. However, labeling all the source data is laborious, and most of the previous DG methods (Ding & Fu, 2017; Balaji et al., 2018; Dou et al., 2019; Wang et al., 2020a; Zhao et al., 2020; Matsuura and Harada, 2020) do not make full use of the information contained in massive unlabeled data. Then, a more practical problem arises: Is it possible to perform domain generalization with only one labeled source dataset as well as multiple unlabeled source datasets? For example, we may train a skin lesion classification model (Li et al., 2020a) by using a labeled skin lesion dataset from a central hospital. Meanwhile, we would like to further improve the generalization ability of the model by employing abundant data from other local hospitals, but the additional data may be unlabeled due to expensive annotation costs.

In this paper, in addition to the Conventional Domain Generalization (CDG) task with multiple labeled source domains, we further investigate a more practical task, namely Single Labeled Domain Generalization (SLDG), where only one of the multiple source domains is labeled (see Fig. 1). The single labeled multi-source data puts a serious obstacle in the path of generalization learning, which we call the *discriminability-generalization bias*: the discriminative information in the labeled source domain may contain domain-specific bias, constraining the out-of-domain generalization ability of the trained model. Thus, how to train a discriminative model while removing its domain-specific bias for guaranteeing generalization is the key to solve this challenging task.

To address this problem, we propose a novel framework called Domain-Specific Bias Filtering (DSBF) for the SLDG task. Specifically, it initializes a discriminative model with the labeled source data and then filters out domain-specific bias in the initialized model with the unlabeled source data for generalization improvement, corresponding to a model initialization stage and a bias filtering stage, respectively. The bias filtering stage consists of (1) feature extractor debiasing via k-means clustering-based semantic feature re-extraction and (2) classifier rectification through attention-guided semantic feature projection. Our method DSBF unifies the exploration of labeled and unlabeled source data to enhance the discriminability and generalization of the trained model, resulting in a highly generalizable model, as verified by theoretical analyses. Extensive experiments on multiple datasets consistently show the superior performance of the proposed DSBF framework for the SLDG task. More-



(a) Conventional Domain Generalization (CDG)

(b) Single Labeled Domain Generalization (SLDG)

Fig. 1 Comparison between the CDG (**a**) and the introduced SLDG tasks (**b**) for visual recognition. The latter is more practical for dealing with the problem of high annotation costs in real-world applications, yet challenging, because only one of the multiple source datasets is labeled, which may lead to a serious problem of discriminability-generalization bias.

over, we also verify the effectiveness of it for the CDG task with multiple labeled source domains.

Our main contributions are summarized as: (1) We investigate a practical and challenging generalization task, namely Single Labeled Domain Generalization (SLDG) with only one source domain being labeled, towards the real scenarios where massive unlabeled data is available for generalizable model training. (2) We propose a novel framework called Domain-Specific Bias Filtering (DSBF) to tackle the SLDG task by unifying the exploration of the labeled and the unlabeled source data, which consists of model initialization and bias filtering that enhances the discriminability and generalization ability of the model, respectively. (3) We verify the proposed method DSBF with theoretical analyses. Extensive experiments on multiple datasets consistently show its superior performance in tackling the SLDG task. Our method can be easily extended to the CDG task and also achieves state-of-the-art performance.

2 Related Work

2.1 Supervised and Semi-Supervised Learning

In recent years, deep learning based supervised learning (SL) has been widely employed in a variety of applications (LeCun et al., 2015). It considers the principle of empirical risk minimization (ERM) (Vapnik, 1992) that a model with low empirical risk on a labeled training dataset is supposed to generalize well on a test dataset. Due to the expensive annotation costs, lots of recent works (Tarvainen & Valpola, 2017; Sohn et al., 2020; Yasarla et al., 2021; Wang et al., 2020b) focus on semi-supervised learning (SSL) (Yang et al., 2021) that utilizes both the labeled and the unlabeled data for model training. For example, Tarvainen and Valpola (2017) train a student model with a classification cost on the labeled data, and use a consistency cost to make the outputs of the student and a teacher be consistent on the unlabeled data for effectively capturing discriminative information. Although the general SSL methods make full use of both the labeled and the unlabeled data for training discriminative models, they assume that all the datasets are sampled from the same distributions, which may make the trained models suffer from significant performance degradation on the test (target) datasets in real scenarios. In comparison, the SLDG task that we investigate aims to train a generalizable model using both the labeled and the unlabeled source data, for better generalization on unknown target domains with different statistical distributions.

2.2 Domain Adaptation

Unsupervised domain adaptation (UDA) (Ben-David et al., 2010; Bellitto et al., 2021; Chen et al., 2021; Dai et al., 2020; Gong et al., 2014; Ho & Gopalan, 2014); (Hoffman et al., 2014; Huang et al., 2021) (Kan et al., 2014; Li et al., 2021; Shen et al., 2021; Sindagi & Srivastava, 2017; Xu et al., 2016; Yamada et al., 2014; Zhao et al., 2021; Zheng & Yang, 2021) is a prevailing direction to DA that addresses the dataset shift between a labeled source domain and an unlabeled target domain. Considerable progress has been made in UDA. A large proportion of them reduces divergence between the source and target domains via adversarial learning (Zhang et al., 2020b; Li et al., 2020b; Ganin et al., 2016; Long et al., 2017, 2018; Saito et al., 2018; Zhang et al., 2019b) or directly minimizing domain discrepancy with a metric like Maximum Mean Discrepancy (MMD) (Li et al., 2020c; Long et al., 2015, 2017). These methods may fail to leverage the available multiple source datasets, leading to insufficient generalization learning.

Increasing works (Zuo et al., 2021; Peng et al., 2019; Zhang et al., 2015; Zhao et al., 2018; Wang et al., 2020c) thus focus on the multi-source domain adaptation (MSDA) (Ben-David et al., 2010) task, where multiple labeled source datasets from different domains are provided for model adaptation. For example, some works (Zuo et al., 2021; Wang et al., 2020c) present an attention-based strategy to reduce domain divergence in the semantic feature space by using the multiple source datasets and an elaborate attention module. Multi-target domain adaptation (MTDA) is another research field of DA, which extends UDA to multiple (Gong et al., 2013; Gholami et al., 2020; Yu et al., 2018; Wang et al., 2020c; Chen et al., 2019; Yu et al., 2018), continuous (Gong et al., 2019; Mancini et al., 2019a; Wu et al., 2019), and latent (Hoffman et al., 2012; Xiong et al., 2014; Mancini et al., 2019b; Liu et al., 2020) target domains. Among them, Chen et al. (2019) introduce blending-target domain adaptation (BTDA) that aims to adapt the model to a mixed target distribution where the multi-target proportions are unobservable. Liu et al. (2020) assume the target domain is a compound of multiple homogeneous domains without domain labels and employ model predictions as the pseudo labels of the unlabeled data to enable a curriculum learning process. Although annotation costs of the target dataset are avoided in the above DA researches, the requirements of re-collecting target data and training model for each new target domain still hinder their applications in real scenarios. In contrast, we aim to train a generalizable model that can directly generalize to unseen target domains in the investigated SLDG task. Note that despite both the MTDA task and our SLDG task assume one labeled dataset and multiple unlabeled datasets, MTDA mainly aims to improve the performance of the model on the seen unlabeled target domains (which can be used for both training and inference), but SLDG aims to improve the performance of the model on unseen target domains (which can only be used for inference).

2.3 Domain Generalization

Recently, domain generalization (DG) (Blanchard et al., 2011) attracts great interest, which learns to extract domain invariance from multiple labeled source datasets and trains a generalizable model to unseen target domains. Since the DG task is similar to meta-learning (Schmidhuber, 1987), some works (Balaji et al., 2018; Dou et al., 2019; Li et al., 2019) employ a meta-learning-based strategy that trains the model on a meta-train dataset and continues to improve the model generalization on a meta-test dataset, both the datasets are constructed from the available labeled multi-source data. Meanwhile, a lot of effort has gone into data augmentation techniques (Carlucci et al., 2019; Wang et al., 2020a). The latent idea of these works is the augmented data generates various new domains, and the models trained on these generated domains could be more generalizable. Similar to DA, some recent DG works (Zhao et al., 2020; Matsuura and Harada, 2020; Zhou et al., 2020) use adversarial learning to learn discriminative and domain-invariant semantic feature representations that can be applied to different domains. Other strategies like normalization (Seo et al., 2020; Zhou et al., 2021c) and else (Huang et al., 2020; Qiao et al., 2020; Yuan et al., 2021b; Ding & Fu, 2017; Yuan et al., 2021a) are also considered in the DG research fields. These methods may require fully labeled multi-source data, which is hard to be satisfied due to the high annotation costs.

Qiao et al. (2020) present to perform domain generalization with only one labeled source domain, and design a meta-learning scheme with an auto-encoder for model training. Besides, some data augmentation (Volpi et al., 2018; Carlucci et al., 2019; Wang et al., 2020a) and gradient-based (Huang et al., 2020) methods may also be extended to the one-labeled-source setting. However, they fail to leverage the unlabeled data, which might be abundant in real scenarios, to further boost the out-of-distribution generalization of the model.

Therefore, in addition to the Conventional Domain Generalization (CDG) setting, we further investigate a more practical task called Single Labeled Domain Generalization (SLDG). The challenging SLDG task only assumes one source dataset to be labeled, and other unlabeled source datasets are further exploited to improve the out-of-distribution generalization of the model.

A related task is the Semi-Supervised Domain Generalization (SSDG) (Zhou et al., 2021a). Both the SSDG and our SLDG tasks aim to train a generalizable model using partially-labeled source data. However, they are different in the following aspects. (1) Problem definition: SSDG assumes that partial samples are labeled in each source domain; but our SLDG task considers that only one source domain is labeled while other domains are totally unlabeled. (2) Solution direction: based on the different definitions, Zhou et al. (2021a) perform semi-supervised training under the consideration of domain shift by extending FixMatch via uncertainty and style consistency learning; but we learn a discriminative model from the labeled dataset and then filter out bias and boost generalization using the unlabeled datasets, corresponding to the model initialization and bias filtering stages, respectively. (3) Application scenario: SSDG focuses on the scenario that multiple partially-labeled datasets are given for generalization learning; but our SLDG task is introduced towards the scenario that a labeled dataset is given for learning a predictive yet biased model, meanwhile, multiple semantically-relevant but unlabeled datasets are available for further boosting its out-of-distribution generalization performance.

2.4 Attention Mechanism

Attention (Bahdanau et al., 2015) is first introduced in natural language processing for deciding which parts of a sentence should be paid more attention to. It is widely applied in various fields (Wang et al., 2017; Zhang et al., 2019a; Fu et al., 2019; Devlin et al., 2018). Self-attention/intra-attention (Vaswani et al., 2017) is a specific form of the attention mechanism, which learns a representation of a sequence by reweighting its different positions according to their importance. A general process of the self-attention consists of three steps: (1) Getting the embeddings of *query*, *key*, and *value* from the original sequence. (2) Obtaining normalized weights by calculating the similarity between the query and the key. (3) Weighting the value. For example, Fu et al. (2019) capture rich contextual dependencies in both spatial and channel dimensions by using a position attention module and a channel attention module, selectively aggregating spatial and channel features for obtaining more effective representations. In the inter-domain attention module of our method DSBF, we let the key-value pairs be constructed from the semantic features of one source domain and the query be constructed from the semantic features of other source domains. In this way, the similar semantic information is automatically enhanced for generalization improvement.

3 Problem Setup

In Conventional Domain Generalization (CDG) task, we may assume that there are K labeled multi-source datasets $\{\mathcal{D}^j\}_{j=1}^K$ with n^j samples in the j-th dataset, i.e., $\mathcal{D}^j = \{(\boldsymbol{x}_i^j, y_i^j)\}_{i=1}^{n^j}$. Any information of the target domain $\mathcal{D}^{K+1}$ is not provided during the model training process. The source datasets $\mathcal{D}^1, ..., \mathcal{D}^K$ and the target dataset $\mathcal{D}^{K+1}$ are sampled from different distributions $P(X^1, Y^1), ..., P(X^K, Y^K)$, $P(X^{K+1}, Y^{K+1})$, respectively, which are defined on input and label joint space $\mathcal{X} \times \mathcal{Y}$. The goal of the CDG task is to use the fully labeled source datasets $\{\mathcal{D}^j\}_{j=1}^K$ to train a predictive model that can perform well on the unseen target dataset $\mathcal{D}^{K+1}$.

In this paper, we further introduce a more challenging task, i.e., Single Labeled Domain Generalization (SLDG). SLDG also aims to improve the generalization performance on the unseen target domain, but only the first source dataset $\mathcal{D}^1 = \{(\boldsymbol{x}_i^1, y_i^1)\}_{i=1}^{n^1}$ is assumed to be labeled, the others $\{\mathcal{D}^j = \{\boldsymbol{x}_i^j\}_{i=1}^{n^j}\}_{j=2}^K$ are supposed to be unlabeled.

The main challenge in the SLDG task is the discriminability-generalization bias. That is, when we use the labeled source data to train a discriminative model for object recognition, the domain-specific bias in the labeled source data would mislead the model, constraining its generalization performance on other domains. Therefore, it is vital to train a discriminative model using the labeled source data while removing its domain-specific bias for generalization improvement.

4 Methodology

We address the SLDG task by proposing Domain-Specific Bias Filtering (DSBF). It initializes a discriminative model using the labeled source data and filters out domain-specific bias in the initialized model using the unlabeled source data for generalization improvement, which corresponds to a *model initialization* stage and a *bias filtering* stage. The bias filtering consists of (1) *feature extractor debiasing* using the unlabeled data and its pseudo labels obtained via k-means clustering and (2) *classifier rectification* through attention-guided semantic feature projection. Our framework and algorithm are shown in Fig. 2 and Algorithm 1, respectively. We then introduce the details of the two stages of the DSBF method in the following.

Algorithm 1 Domain-Specific Bias Filtering

```
Require: A labeled source dataset $\mathcal{D}^1$, unlabeled source datasets $\{\mathcal{D}^j\}_{j=2}^K$, a backbone $g$ and a network $b$ with parameter $\theta_g$ and $\theta_b$ of the feature extractor, a classifier $c$ with parameter $\theta_c$, semantic feature projection networks $\{v_j\}_{j=2}^K$ with parameters $\{\theta_{v_j}\}_{j=2}^K$, initialization/filtering iterations $M/N$.
Ensure: Well-trained $\hat{g}$, $\hat{b}$, and $\hat{c}$ for inference.
Initialize SGD optimizers and parameters;
for $iter = 1$ to $M$ do // model initialization
    Sample a batch data from $\mathcal{D}^1$;
    Update $\theta_g, \theta_b, \theta_c$ by minimizing Eq. (1);
end for
for $iter = 1$ to $N$ do // bias filtering
    Sample a batch data from $\mathcal{D}^j$, $j = 1, ..., K$;
    Get pseudo labels via Eq. (2–4);
    Update $\theta_g, \theta_b$ by minimizing Eq. (5–6);
    Update $\{\theta_{v_j}\}_{j=2}^K$ by minimizing Eq. (7);
    Update $\theta_c$ by minimizing Eq. (8).
end for
```

4.1 Model Initialization

To initialize a discriminative model, we use the labeled source data $\mathcal{D}^1$ to pretrain the feature extractor $b \circ g$ and the classifier c for learning to extract semantic features of the data and classifying the extracted features to the corresponding categories, respectively. The used cross-entropy classification loss of the labeled source data for initializing the model, i.e., $b \circ g$ and c, is

$$\mathcal{L}_{CL} = -\sum_{r=1}^{C} \boldsymbol{y}_r^1 \log f_r^b(\boldsymbol{x}^1), \tag{1}$$

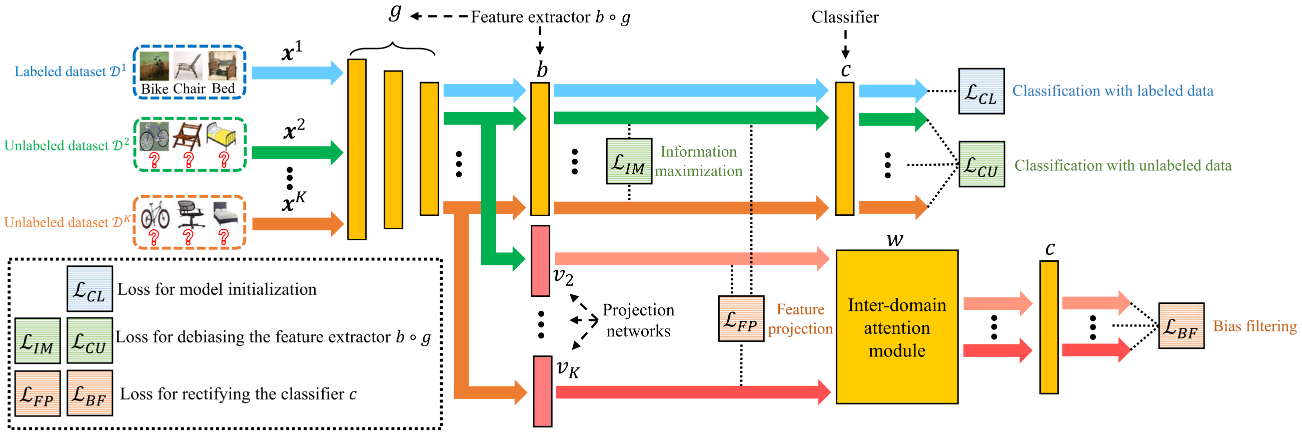

g
Feature extractor $b \circ g$
Classifier
b
c
Labeled dataset $\mathcal{D}^1$
Bike Chair Bed
Unlabeled dataset $\mathcal{D}^2$
Unlabeled dataset $\mathcal{D}^K$
?
x^1
x^2
x^K
$\mathcal{L}_{CL}$
Classification with labeled data
$\mathcal{L}_{IM}$
Information maximization
$\mathcal{L}_{CU}$
Classification with unlabeled data
v_2
v_K
Projection networks
$\mathcal{L}_{FP}$
Feature projection
w
Inter-domain attention module
c
$\mathcal{L}_{BF}$
Bias filtering
$\mathcal{L}_{CL}$ Loss for model initialization
$\mathcal{L}_{IM}$ $\mathcal{L}_{CU}$ Loss for debiasing the feature extractor $b \circ g$
$\mathcal{L}_{FP}$ $\mathcal{L}_{BF}$ Loss for rectifying the classifier c

Fig. 2 Our proposed Domain-Specific Bias Filtering (DSBF) framework. The whole model consists of a feature extractor $b \circ g$, a classifier c, projection networks $\{v_j\}_{j=2}^K$, and an attention module w. We employ $\{v_j\}_{j=2}^K$ and w only for classifier rectification in training. After training, we only use the trained $\hat{c} \circ \hat{b} \circ \hat{g}$ for inference on out-of-distribution target domains.

where $f^b := c \circ b \circ g$ outputs softmax classification of the data, and f^b_r is the rth dimension output of f^b. $\boldsymbol{y}^j_r$ is the r-th dimension of one-hot encoding of the labels $y^j \in \{1, ..., C\}$ of domain j, where the correct class is "1", otherwise is "0".

After model initialization, the feature extractor $b \circ g$ and the classifier c are pretrained to extract semantic features of the data and use them for classification, respectively. However, they may be misled by the domain-specific bias of the labeled source data. Thus, we utilize the unlabeled data to filter out the domain-specific bias in the initialized model for generalization improvement.

4.2 Bias Filtering

The bias filtering consists of feature extractor debiasing and classifier rectification. In feature extractor debiasing, we aim to train the feature extractor using the unlabeled source data for reducing the bias of the feature extractor towards the labeled source data, and hence obtaining a more robust model. Since the unlabeled source data does not have ground-truth labels, we exploit k-means clustering to obtain the pseudo labels and use them to train the feature extractor for effective semantic feature re-extraction. In classifier rectification, we project the semantic features of the unlabeled source data to the semantic features of the labeled source data, and then use the projection features to predict the labels of the labeled source data. Since the projection features only contain the bias of the unlabeled source data (because it is obtained by feeding the features of the unlabeled source data to the projection networks) while the labels only contain the bias of the labeled source data, optimizing the classifier with supervised loss can debias it, which is verified by the theoretical analyses in Sect. 4.3. We introduce an inter-domain attention module to further capture the similarities among domains and boost generalization performance.

4.2.1 Feature Extractor Debiasing

We obtain pseudo labels of the unlabeled data to facilitate the following feature extractor debiasing and classifier rectification processes. Inspired by recent works (Kang et al., 2019; Liang et al., 2020) on deep clustering (Caron et al., 2018), we adopt k-means clustering to assign pseudo labels $\{\hat{\boldsymbol{y}}^j\}_{j=2}^K$ for the unlabeled source datasets $\{\boldsymbol{x}^j\}_{j=2}^K$. Specifically, we first get a centroid $\boldsymbol{a}^{(0)}_{(j,r)}$ of each class r for the semantic features of each unlabeled domain j by softly assigning each sample $\boldsymbol{x}^j$ to it with model prediction-based score $f^b_r(\boldsymbol{x}^j)$, that is

$$\boldsymbol{a}^{(0)}_{(j,r)} = \frac{\sum_{\boldsymbol{x}^j} f^b_r(\boldsymbol{x}^j) b \circ g(\boldsymbol{x}^j)}{\sum_{\boldsymbol{x}^j} f^b_r(\boldsymbol{x}^j)}. \tag{2}$$

The centroid $\boldsymbol{a}^{(0)}_{(j,r)}$ represents the semantic feature distribution of each class r in domain j, which is used to assign the initial pseudo label d^j for the samples $\boldsymbol{x}^j$, that is

$$d^j = \arg\min_r \mathrm{dist}(b \circ g(\boldsymbol{x}^j), \boldsymbol{a}^{(0)}_{(j,r)}), \tag{3}$$

where $\mathrm{dist}(\cdot, \cdot)$ measures the cosine distance. Similarly, we then get updated centroid $\boldsymbol{a}^{(1)}_{(j,r)}$ and final pseudo labels $\hat{y}^j$ by

$$\begin{aligned} \boldsymbol{a}^{(1)}_{(j,r)} &= \frac{\sum_{\boldsymbol{x}^j} 1(d^j = r) b \circ g(\boldsymbol{x}^j)}{\sum_{\boldsymbol{x}^j} 1(d^j = r)}, \\ \hat{y}^j &= \arg\min_r \mathrm{dist}(b \circ g(\boldsymbol{x}^j), \boldsymbol{a}^{(1)}_{(j,r)}). \end{aligned} \tag{4}$$

The pseudo labels $\hat{y}^j$ can be transformed into one-hot encoding $\hat{\boldsymbol{y}}^j$. We consider the ideal form of the softmax outputs of c should be like one-hot encoding for each sample, and be distinct for samples from different classes. Therefore, we improve the clustering performance by optimizing g, b with information maximization constraint (Kundu et al., 2020; Liang et al., 2020) loss:

$$\mathcal{L}_{IM} = \frac{1}{K-1} \sum_{j=2}^{K} \left\{ \underbrace{\sum_{r=1}^{C} t_r \log t_r}_{\text{diverse class}} \underbrace{- \mathbb{E}[\sum_{r=1}^{C} u_r \log u_r]}_{\text{concentrated sample}} \right\}, \tag{5}$$

where $t_r = \mathbb{E}[f^b_r(\boldsymbol{x}^j)]$ and $u_r = f^b_r(\boldsymbol{x}^j)$. The first term on the r.h.s. of Eq. (5), i.e., negative expected entropy of population, makes the outputs of c diverse at the class level. The second term on the r.h.s. of Eq. (5), i.e., expected entropy of individual, makes the outputs of c be concentrated at the sample level. Through minimizing $\mathcal{L}_{IM}$, we encourage the unlabeled samples with closer distance group together, meanwhile, the samples far away are further separated. It improves the clustering performance and allows us to obtain more accurate pseudo labels for bias filtering.

After clustering, we obtain the pseudo labels of the unlabeled source data. To debias the feature extractor $b \circ g$, we present to retrain it with the average classification loss of all the unlabeled source datasets, thus re-extract the semantic feature of the source data, that is,

$$\mathcal{L}_{CU} = -\frac{1}{K-1} \sum_{j=2}^{K} \sum_{r=1}^{C} \hat{\boldsymbol{y}}^j_r \log f^b_r(\boldsymbol{x}^j). \tag{6}$$

By minimizing the classification loss of both the labeled and unlabeled data, i.e., $\mathcal{L}_{CL}$ and $\mathcal{L}_{CU}$, the feature extractor $b \circ g$ is trained to reduce its bias towards the labeled source domain. Despite that the feature extractor may still be affected by the domain-specific factors of all the source domains, we argue that the process of feature extractor debiasing could allow us to obtain more effective domain-agnostic semantic features from data, facilitating the classifier rectification process with semantic feature projection. Different from Liang et al. (2020), we do not choose to optimize the classifier using the pseudo labels since it could yield adverse effects in our experiments.

4.2.2 Classifier Rectification

As the feature extractor is debiased, we employ the generated semantic features of the unlabeled source to filter out the domain-specific bias in the classifier. Specifically, we first project the semantic features of the unlabeled sources, i.e., $g(\boldsymbol{x}^j)$, $j = 2, ..., K$, to the semantic features of the labeled source, i.e., $b \circ g(\boldsymbol{x}^1)$, with the semantic feature projection networks $\{v_j\}_{j=2}^K$. To improve the class-level domain invariance, we perform conditional projection, i.e., projecting the semantic features of the unlabeled sources to the semantic features of the labeled source which are in the same class by aligning the true labels $\boldsymbol{y}^1$ and the pseudo labels $\{\hat{\boldsymbol{y}}^j\}_{j=2}^K$. Thus, we minimize a feature projection loss to optimize the projection networks $\{v_j\}_{j=2}^K$:

$$\mathcal{L}_{FP} = \frac{1}{K-1} \sum_{j=2}^{K} s^j \left(b \circ g\left(\boldsymbol{x}^1\right) - v_j \circ g(\boldsymbol{x}^j\right)^2, \qquad (7)$$

where $s^j = 1(\boldsymbol{y}^1 = \hat{\boldsymbol{y}}^j)$, i.e., if $\boldsymbol{y}^1 = \hat{\boldsymbol{y}}^j$, then $s^j = 1$, else $s^j = 0$. Through semantic feature projection, the semantic invariance in data is contained in the projection semantic features $v_j \circ g(\boldsymbol{x}^j)$. Then we use it to rectify/optimize the classifier c by minimizing the bias filtering loss, which is the average classification loss of the projections $v_j \circ g(\boldsymbol{x}^j)$:

$$\mathcal{L}_{BF} = -\frac{1}{K-1} \sum_{j=2}^{K} \sum_{r=1}^{C} s^j \boldsymbol{y}_r^1 \log f_r^{v_j}(\boldsymbol{x}^j), \qquad (8)$$

where $f^{v_j} := c \circ w \circ v_j \circ g$ outputs the softmax classification of the projections, and $f_r^{v_j}$ is the r-th dimension output of f^{v_j}. An inter-domain attention module w is designed to enhance the similarities of semantic information among domains, which will be introduced in the following. By minimizing Eq. (8), the classifier c uses invariant semantic information contained in the projections to filter out the domain-specific bias and capture invariant correlation between the features and the labels. In Sect. 4.3, we provide theoretical insights to make it clearer and more specific.

In order to further facilitate the bias filtering process, we put forward an inter-domain attention module to enhance the domain similarities as shown in Fig. 3. Let B be the batchsize and D be the semantic feature dimension, we feed the outputs of the projection networks $\{v_j\}_{j=2}^K$ with size $B \times D$ to embedding networks $\{a_j\}_{j=2}^K$ and get semantic feature embeddings $\{\boldsymbol{o}_j\}_{j=2}^K$ with size $B \times D$. Our goal is to obtain the re-weighted $\{\boldsymbol{o}_j\}_{j=2}^K$, i.e., $\{\boldsymbol{q}_j\}_{j=2}^K$, based on the aggregated inter-domain similarities among $\{\boldsymbol{o}_j\}_{j=2}^K$, i.e., $\{\boldsymbol{z}_j\}_{j=2}^K$, for more effective bias filtering and generalization boost.

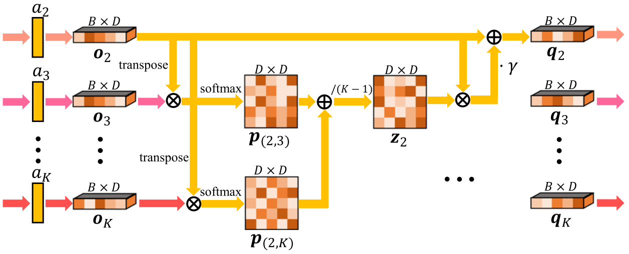


Fig. 3 The proposed inter-domain attention module. It first generates semantic feature embeddings $\{\boldsymbol{o}_j\}_{j=2}^K$ with embedding networks $\{a_j\}_{j=2}^K$, then weights them according to the inter-domain similarities and obtain weighted semantic features $\{\boldsymbol{q}_j\}_{j=2}^K$ for classifier calibrating. Domain invariance is enhanced automatically in this process.

We begin by taking a domain m as an example, where $m \in \{2, ..., K\}$. Note that we denote m as a chosen domain, and denote j as the other domains used to calculate inter-domain similarities and attention. We first get the normalized inter-domain similarity matrices $\{\boldsymbol{p}_{(m,j)}\}_{j=2}^K$ of $\boldsymbol{o}_m$ by multiplying the transpose of $\boldsymbol{o}_m$ with $\{\boldsymbol{o}_j\}_{j=2}^K$:

$$\boldsymbol{p}_{(m,j)} = \frac{\exp\left((\boldsymbol{o}_m)^\top \boldsymbol{o}_j\right)}{\sum_{j=2}^{K} \exp\left((\boldsymbol{o}_m)^\top \boldsymbol{o}_j\right)}, \; j = 2, ..., K, \qquad (9)$$

where $\boldsymbol{p}_{(m,j)}$ is the inter-domain similarity matrix of domain m and j with size $D \times D$. Each position of $\boldsymbol{p}_{(m,j)}$ represents the similarities between the corresponding position of $\boldsymbol{o}_m$ and $\boldsymbol{o}_j$. Since $\{\boldsymbol{o}_j\}_{j=2}^K$ is learned from the projection of each unlabeled source to the labeled source, and $\{\boldsymbol{p}_{(m,j)}\}_{j=2}^K$ extract the common semantic information between $\boldsymbol{o}_m$ and $\{\boldsymbol{o}_j\}_{j=2}^K$, averaging $\{\boldsymbol{p}_{(m,j)}\}_{j=2}^K$ encourages the aggregation of the common semantic information, that is,

$$\boldsymbol{z}_m = \frac{1}{K-1} \sum_{j=2}^{K} \boldsymbol{p}_{(m,j)}. \qquad (10)$$

Each position of $\boldsymbol{z}_m$ represents the overall response of the projection of other domains to the projection of domain m, which also indicates the common semantic information of each position of them. Then we get the re-weighted semantic features $\boldsymbol{q}_m$ by multiplying $\boldsymbol{o}_m$ with $\boldsymbol{z}_m$, and perform an element-wise sum operation with $\boldsymbol{o}_m$, that is,

$$\boldsymbol{q}_m = \alpha \cdot \boldsymbol{o}_m \boldsymbol{z}_m + \boldsymbol{o}_m, \tag{11}$$

where α is a parameter initialized as 0 and trained to provide suitable weight. It is updated with the model parameters together (we add a small perturbation from a uniform distribution $U(0, 1)$ to it to make it trained stably). In this way, through the calculation of all the embedding semantic features $\{\boldsymbol{o}_j\}_{j=2}^K$, we can obtain all the re-weighted semantic features $\{\boldsymbol{q}_j\}_{j=2}^K$, which is re-weighted by semantic similarities among the source domains. This inter-domain attention module encourages the learning of the common semantic information of the semantic feature projection. It effectively assists the bias filtering and improve the generalization performance as verified in the experiments.

Remark Note that Fig. 3 is simplified. In our experiments, each network in $\{a_j\}_{j=2}^K$ is composed of three subnetworks that output the embeddings of query $\{\boldsymbol{o}_j^{query}\}_{j=2}^K$, key $\{\boldsymbol{o}_j^{key}\}_{j=2}^K$, and value $\{\boldsymbol{o}_j^{value}\}_{j=2}^K$, respectively. Eq. (9) is calculated with the key part of $\boldsymbol{o}_m$ and query part of $\boldsymbol{o}_j$, i.e., $\boldsymbol{o}_m^{key}$ and $\boldsymbol{o}_j^{query}$. While Eq. (11) is calculated with the value part of $\boldsymbol{o}_m$, i.e., $\boldsymbol{o}_m^{value}$.

4.2.3 Optimization Details

To illustrate the optimization process clearly, we merge the optimization losses to a loss for stage 1, i.e. $\mathcal{L}_{S1}$, and a loss for stage 2, i.e. $\mathcal{L}_{S2}$, that is,

$$\begin{aligned}\mathcal{L}_{S1} &= \mathcal{L}_{CL}\\ \mathcal{L}_{S2} &= \lambda(\mathcal{L}_{IM} + \mathcal{L}_{CU}) + \gamma(\mathcal{L}_{FP} + \mathcal{L}_{BF})\end{aligned} \tag{12}$$

In the first stage, we initialize the model with the classification loss $\mathcal{L}_{CL}$ on the labeled source data. In the second stage, we debias the feature extractor through the classification loss $\mathcal{L}_{CU}$ on the unlabeled source data with the pseudo labels obtained via k-means clustering and the information maximization loss $\mathcal{L}_{IM}$. We rectify the classifier with the feature projection loss $\mathcal{L}_{FP}$ and the bias filtering loss $\mathcal{L}_{BF}$. λ and γ are the hyper-parameters for balancing the feature extractor debiasing and the classifier rectification processes.

The CDG task In the CDG task, since the groud-truth labels of all the source data are given, we directly employ them for training instead of obtaining pseudo labels through the clustering.

4.3 Theoretical Insights

In the SLDG task, since only one source dataset is labeled, we put forward to rectify the classifier by performing the semantic feature projection. For simplicity, we denote the semantic features extracted from data X^j as $H^j \in \mathbb{R}^{d_h}$, and let it be composed of domain-invariant factor $U \in \mathbb{R}^{d_h}$ and domain-specific factor/bias $L^j \in \mathbb{R}^{d_h}$, that is,

$$H^j = (\boldsymbol{\phi}^j)^\top U + (\boldsymbol{\eta}^j)^\top L^j,\ j = 1, ..., K, \tag{13}$$

where $\boldsymbol{\phi}^j \in \mathbb{R}^{d_h \times d_h}$ and $\boldsymbol{\eta}^j \in \mathbb{R}^{d_h \times d_h}$ are coefficient matrices, which may change across the domains.

Inspired by the ability of the human in robust visual object recognition that no matter how the domain/environment changes, human can always accurately identify the class of the recognized image (Zhang et al., 2020a). We assume that there is an invariant correlation β between the semantic features H^j and the corresponding labels $Y^j \in \mathbb{R}$, meanwhile, the labels Y^j may also be biased by the domain-specific factor L^j, that is,

$$Y^j = \beta^\top H^j + (\psi^j)^\top L^j,\ j = 1, ..., K, \tag{14}$$

where $\beta \in \mathbb{R}^{d_h}$ and $\psi^j \in \mathbb{R}^{d_h}$ are coefficient vectors. β stays unchanged but ψ^j changes across the domains. Note that we assume $\mathbb{E}[L^j] = 0$ for $j = 1, ..., K$. The main assumption is summarized:

Assumption 1 The semantic features H^j and the labels Y^j in each domain j satisfy Eqs. (13) and (14) respectively, where only the domain-invariant factor U and the correlation β stay unchanged across domains. The domain-specific and domain-invariant factors are pairwise independent, i.e., $U \perp L^k$ and $L^j \perp L^k$ for $j, k \in \{1, ..., K\}$ and $j \neq k$.

Our goal is to identify the latent correlation β between the features and the labels. Let m and n be an unlabeled and a labeled source domain, respectively. We first project the semantic features H^m of the unlabeled source data to the semantic features H^n of the labeled source data with a mapping matrix $\boldsymbol{\mu} \in \mathbb{R}^{d_h \times d_h}$, that is, $\hat{\boldsymbol{\mu}} = \mathbb{E}[H^m(H^m)^\top]^{-1}\mathbb{E}[H^m(H^n)^\top]$. Then we use the projection semantic features, i.e., $\hat{H}^n = \hat{\boldsymbol{\mu}}^\top H^m$, to fit the labels Y^n of the labeled source and estimate the correlation $\hat{\beta} = \mathbb{E}[\hat{H}^n(\hat{H}^n)^\top]^{-1}[\hat{H}^n(Y^n)^\top]$, i.e., classifier rectification. We have the theorem.

Theorem 1 *Suppose there are n samples from each domain, then $\hat{\beta}$ is a consistent and unbiased estimator of the true correlation β, i.e., $\hat{\beta} = \beta + O_p\left(\frac{1}{\sqrt{n}}\right)$ and $\mathbb{E}[\hat{\beta}] = \beta$.*

Proof Assume that we sample n examples from each domain. Let $\mathbf{H}^m \in \mathbb{R}^{n \times d_h}$ be the matrix where ith row is the observation $\boldsymbol{h}_i^m \in \mathbb{R}^{d_h}$ of H^m, and other symbols are similarly defined. The first step is to regress $\mathbf{H}^n$ on $\mathbf{H}^m$, i.e., $\hat{\mu} = \left((\mathbf{H}^m)^\top \mathbf{H}^m\right)^{-1}(\mathbf{H}^m)^\top \mathbf{H}^n$. The second step is to regress $\mathbf{Y}^n$ on $\hat{\mathbf{H}}^n = \mathbf{H}^m \hat{\mu}$, i.e., $\hat{\beta} = \left((\hat{\mathbf{H}}^n)^\top \hat{\mathbf{H}}^n\right)^{-1}(\hat{\mathbf{H}}^n)^\top \mathbf{Y}^n$.

By Assumption 1, we have

$$\begin{aligned}\frac{1}{n}(\mathbf{H}^m)^\top\mathbf{L}^n =&\frac{1}{n}\left(\mathbf{U}\boldsymbol{\phi}^m+\mathbf{L}^m\boldsymbol{\eta}^m\right)^\top\mathbf{L}^n\\ =&O_p\left(\frac{1}{\sqrt{n}}\right).\end{aligned}\tag{15}$$

$$\begin{aligned}\frac{1}{n}(\mathbf{H}^n)^\top\mathbf{H}^m =&\frac{1}{n}\left(\mathbf{U}\boldsymbol{\phi}^n+\mathbf{L}^n\boldsymbol{\eta}^n\right)^\top\cdot\left(\mathbf{U}\boldsymbol{\phi}^m+\mathbf{L}^m\boldsymbol{\eta}^m\right)\\ =&\frac{1}{n}(\boldsymbol{\phi}^n)^\top\mathbf{U}^\top\mathbf{U}\boldsymbol{\phi}^m+O_p\left(\frac{1}{\sqrt{n}}\right),\end{aligned}\tag{16}$$

$$\begin{aligned}&\frac{1}{n}(\mathbf{H}^m)^\top\mathbf{H}^m\\ =&\frac{1}{n}\left(\mathbf{U}\boldsymbol{\phi}^m+\mathbf{L}^m\boldsymbol{\eta}^m\right)^\top\cdot\left(\mathbf{U}\boldsymbol{\phi}^m+\mathbf{L}^m\boldsymbol{\eta}^m\right)\\ =&\frac{1}{n}\left((\boldsymbol{\phi}^m)^\top\mathbf{U}^\top\mathbf{U}\boldsymbol{\phi}^m+(\boldsymbol{\eta}^m)^\top(\mathbf{L}^m)^\top\mathbf{L}^m\boldsymbol{\eta}^m\right.\\ &\left.+O_p\left(\frac{1}{\sqrt{n}}\right)\right).\end{aligned}\tag{17}$$

Suppose the minimum eigenvalue of $(\boldsymbol{\phi}^m)^\top\cdot\mathbb{E}[UU^\top]\cdot\boldsymbol{\phi}^m$ is bounded away from 0, we have

$$\begin{aligned}&\left(\frac{1}{n}(\boldsymbol{\phi}^m)^\top\mathbf{U}^\top\mathbf{U}\boldsymbol{\phi}^m+O_p\left(\frac{1}{\sqrt{n}}\right)\right)^{-1}\\ =&\left((\boldsymbol{\phi}^m)^\top\cdot\mathbb{E}[UU^\top]\cdot\boldsymbol{\phi}^m\right)^{-1}+O_p\left(\frac{1}{\sqrt{n}}\right).\end{aligned}\tag{18}$$

Since $(\boldsymbol{\eta}^m)^\top(\mathbf{L}^m)^\top\mathbf{L}^m\boldsymbol{\eta}^m/n$ is positive semidefinite matrices. Hence, the minimum eigenvalue of $(\boldsymbol{\phi}^m)^\top\cdot\mathbb{E}[U(U)^\top]\cdot\boldsymbol{\phi}^m+(\boldsymbol{\eta}^m)^\top\cdot\mathbb{E}[L^m(L^m)^\top]\cdot\boldsymbol{\eta}^m$ is bounded away from 0, then

$$\begin{aligned}&\left(\frac{1}{n}\left((\boldsymbol{\phi}^m)^\top\mathbf{U}^\top\mathbf{U}\boldsymbol{\phi}^m+(\boldsymbol{\eta}^m)^\top(\mathbf{L}^m)^\top\mathbf{L}^m\boldsymbol{\eta}^m\right.\right.\\ &\left.\left.+O_p\left(\frac{1}{\sqrt{n}}\right)\right)\right)^{-1}\\ =&\left((\boldsymbol{\phi}^m)^\top\cdot\mathbb{E}[UU^\top]\cdot\boldsymbol{\phi}^m\right.\\ &\left.+(\boldsymbol{\eta}^m)^\top\cdot\mathbb{E}[L^m(L^m)^\top]\cdot\boldsymbol{\eta}^m\right)^{-1}+O_p\left(\frac{1}{\sqrt{n}}\right).\end{aligned}\tag{19}$$

Therefore, by Eqs. (15–19), we have

$$\begin{aligned}\hat{\beta}=&\left(\left(\hat{\mathbf{H}}^n\right)^\top\hat{\mathbf{H}}^n\right)^{-1}(\hat{\mathbf{H}}^n)^\top\mathbf{Y}^n\\ =&\left((\mathbf{H}^n)^\top\mathbf{H}^m\left((\mathbf{H}^m)^\top\mathbf{H}^m\right)^{-1}\left(\mathbf{H}^m\right)^\top\mathbf{H}^n\right)^{-1}\\ &\cdot(\mathbf{H}^n)^\top\mathbf{H}^m\left((\mathbf{H}^m)^\top\mathbf{H}^m\right)^{-1}(\mathbf{H}^m)^\top\\ &\cdot\left(\mathbf{H}^n\beta+\mathbf{L}^n\psi^n\right)\\ =&\beta+O_p\left(\frac{1}{\sqrt{n}}\right).\end{aligned}$$

We then have $\mathbb{E}[\hat{\beta}]=\beta$. □

Theorem 1 indicates that we can use the semantic features of the unlabeled source to filter out the domain-specific factor/bias of the labeled source and capture the domain-invariant correlation β for more stable domain generalization. Since our theoretical analyses are based on the linear setting, we design the inter-domain attention module to further improve the bias filtering process by learning from the similarities among domains. In this way, the common parts of the features of the unlabeled source domains are enhanced, which helps to further remove the bias and boost the generalization performance.

5 Experiments

We first implement experiments for the introduced Single Labeled Domain Generalization (SLDG) task. We compare our method DSBF with the standard Supervised Learning (SL) algorithm as well as the state-of-the-art algorithms of Semi-Supervised Learning (SSL), Unsupervised Domain Adaptation (UDA), Multi-Target Domain Adaptation (MTDA), and Domain Generalization (DG). To further testify the performance of our method DSBF in domain-specific bias filtering, we then include the comparison with the other DG methods for the Conventional Domain Generalization (CDG) task, where the labels of all the source datasets are provided.

5.1 Setup

5.1.1 Benchmark Datasets

We first adopt two popular benchmark datasets. One is **PACS** (Li et al., 2017) that contains 9,991 images from 7 classes in 4 domains, i.e., Artpaint (*Ar*), Cartoon (*Ca*), Sketch (*Sk*), and Photo (*Ph*). Another one is **Office-Home** (Venkateswara et al., 2017) that consists about 15,500 images of 65 categories over 4 domains, i.e., Art (*Ar*), Clipart (*Cl*), Product (*Pr*), and Real-World (*Rw*). We then perform experiments on a more challenging large-scale dataset called **DomainNet** (Peng et al., 2019). By following (Zhou et al., 2021b), we use four representative domains, i.e., Clipart (*Cl*), Painting (*Pa*), Real (*Re*), and Sketch (*Sk*), for the experiments. In order to further evaluate the performance under the scenarios of more unlabeled source datasets, we construct a new dataset **Office-Caltech-Home**. Specifically, we choose the common classes from Office-Caltech (Gong et al., 2012) and

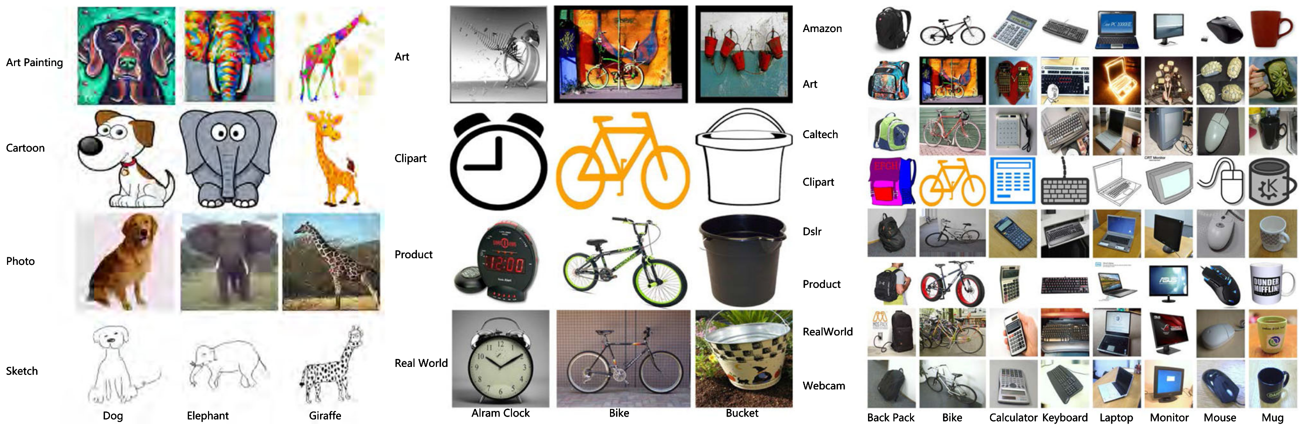


Fig. 4 Example images of the datasets. *Left*: PACS dataset (Li et al., 2017) with four domains, i.e., Art Painting (Ar), Cartoon (Ca), Photo (Ph), and Sketch (Sk). *Middle*: Office-Home dataset (Venkateswara et al., 2017) with four domains, i.e., Art (Ar), Clipart (Cl), Product (Pr), and Real World (Rw). *Right*: our Office-Caltech-Home dataset with eight domains, i.e., Amazon (Am), Art (Ar), Caltech (Ca), Clipart (Cl), Dslr (Ds), Product (Pr), Real World (Rw), and Webcam (We).

Office-Home (Venkateswara et al., 2017) datasets, i.e., backpack, bike, calculator, keyboard, laptop (computer), monitor, mouse, mug, and merge the two datasets to be a new dataset Office-Caltech-Home that has 4,266 images of 8 classes in 8 domains, i.e., Amazon (Am), Webcam (We), DSLR (Ds), Caltech (Ca), Art (Ar), Clipart (Cl), Product (Pr), and Real-World (Rw). We discard DSLR since it only has few images. We use the rest 7 domains with 4,145 images. Example images are shown in Fig. 4.

5.1.2 Baseline Methods

For the experiments of the SLDG task, multiple source datasets are used for model training but only one of them is labeled. The first baseline method is the standard Supervised Learning (SL). ERM (Vapnik, 1992) is employed that minimizes the empirical risk, i.e., cross-entropy loss of classification, on the labeled source dataset. For Semi-Supervised Learning (SSL), both the labeled source dataset and the mixture of the unlabeled source datasets are utilized. We conduct two state-of-the-art SSL methods, i.e., Mean Teacher (Tarvainen & Valpola, 2017) and FixMatch (Sohn et al., 2020), their strategies are related to knowledge distillation (Hinton et al., 2015) and data augmentation, respectively. We also compare DSBF with Unsupervised Domain Adaptation (UDA), where the labeled source dataset and the mixture of the unlabeled source datasets are used as the source dataset and the unlabeled target dataset in the UDA task, respectively. Several representative UDA methods are considered, i.e., DAN (Long et al., 2015), MCD (Saito et al., 2018), and MDD (Zhang et al., 2019b). Besides, Multi-Target Domain Adaptation (MTDA) is considered to use the labeled source dataset and multiple unlabeled source datasets as the labeled source dataset and multiple unlabeled target datasets, respectively. We employ the state-of-the-art MTDA methods, i.e., BTDA (Chen et al., 2019) and OCDA (Liu et al., 2020) as the baselines. Since only one labeled dataset can be utilized in the SLDG task, we compare DSBF with the DG methods which can be extended to this task, including data augmentation based methods JiGen (Carlucci et al., 2019) and GUD (Volpi et al., 2018), a training heuristics method RSC (Huang et al., 2020), and a single domain method M-ADA (Qiao et al., 2020). These works have been introduced in Sect. 2.

5.1.3 Implementation Details

Following (Carlucci et al., 2019; Dou et al., 2019; Huang et al., 2020), we employ the pre-trained ResNet-18 (He et al., 2016) as the feature extractor $b \circ g$ for all the experiments. The architecture of each projection networks $\{v_j\}_{j=2}^{K}$ is a fully-connected layer with 256 units. The classifier is a fully-connected layer with the same units as the image classes. For Algorithm 1, we train the model by SGD optimizer with batchsize 64, learning rate 0.01, momentum 0.9, and weight decay 0.001. In order to achieve efficient and stable training of Eqs. (7–8), in each iteration, we sample data batches from 4 random classes (size of each batch is 16 for each class), we then calculate the loss within each class and obtain the final average loss to update model parameters. The epochs for model initialization and bias filtering are both set to 20, 30, 20, 10 on PACS, Office-Home, Office-Caltech-Home, DomainNet datasets respectively. We split dataset by 0.9/0.1 for training/validation. Note that we report the average classification accuracy of 3 runs with different random seeds for the experiments of the SLDG task. We implement the baseline methods based on their source code and report the results with two decimals. The DAN, MCD, and MDD methods are implemented based on the Transfer Learning Library https://

github.com/thuml/Transfer-Learning-Library which reports the results with one decinal. For the experiments of the CDG task, we cite the results of the baseline DG methods in the related papers. We directly use the groundtruth labels rather than the pseudo labels from clustering in the CDG task. Since all the source datasets are labeled in the CDG task, we are allowed to choose one source dataset as the $\mathcal{D}^1$. In our experiments on PACS dataset, when the target domain is Ar or Ph, we let Sk be the labeled source dataset $\mathcal{D}^1$; and when the target domain is Ca and Sk, we let Ph be the labeled source dataset $\mathcal{D}^1$. For Office-Home dataset, when the target domain is Ar or Cl, we let Pr be the labeled source dataset $\mathcal{D}^1$; and when the target domain is Pr or Rw, we let Cl be the labeled source dataset $\mathcal{D}^1$. In model initialization, we use all the multi-source data for the CDG task, and only use $\mathcal{D}^1$ for the SLDG task. We use default hyper-parameters, i.e., λ and γ are set to 1, in the main experiments, and further analyze their sensitivity later.

5.2 Results for the SLDG Task

Tables 1 and 2 report the results of the SLDG task on PACS and Office-Home datasets, respectively. We first note that the SSL methods, i.e., Mean Teacher and FixMatch, fail badly, which is probably because they rely on the i.i.d. assumption and hence severely overfit the labeled and unlabeled datasets that actually sampled from different domains/distributions. The next observation is that the DG methods, i.e., GUD, JiGen, RSC, M-ADA, show comparable performance to the standard SL method ERM, which is probably because they can not identify the domain invariance well by only utilizing one labeled source data. Since the UDA methods address the dataset shift by using both the labeled and the unlabeled data, they are allowed to learn more effective domain-invariant semantic information and hence perform obviously better. The reason for the worse performance of the MTDA methods, i.e. OCDA and BTDA, may be that they need some strong assumptions. For example, OCDA (Liu et al., 2020) considers a more homogeneous setting that the domain divergence is indistinct, and it directly employs the model predictions of the unlabeled data as pseudo labels for the model training. In comparison, the proposed DSBF method performs the best on 5 and 6 sub-tasks of the 12 sub-tasks on PACS and Office-Home datasets, respectively, and achieves the highest average accuracy which is much higher than other methods on both datasets. We argue that it is because DSBF method makes full use of the unlabeled source data to filter out the domain-specific bias and captures the invariant correlation between the semantic features and the labels, resulting in a well generalizable model for out-of-distribution target data.

We then report the results of the SLDG task on a more challenging large-scale dataset, i.e., DomainNet, in Table 3. It still shows that the semi-supervised methods may fail to learn generalization from data with distribution shift. The DG methods which can only make use of the labeled source data, especially JiGen, RSC, and M-ADA, performs worse than the UDA methods and our method in this label-limited scenario. Despite the UDA methods achieve good performance, they are still surpassed by our method of DSBF since they do not prepare for the generalization on the unseen target domains. We show that DSBF still yields superior out-of-distribution generalization performance on the large-scale dataset.

To further evaluate the generalization performance gain from the unlabeled source data, we consider the scenarios with more domains using Office-Caltech-Home dataset as shown in Fig. 5. We find that the performance improves obviously when only one unlabeled source dataset is used, especially in the third group (on the right of Fig. 5) where the labeled source domain is We and the target domain is Rw, the utilization of the unlabeled source domain Cl significantly improves the accuracy from 80.86% to 93.58%. Moreover, we observe a gradual improvement in performance when given more unlabeled source domains. It indicates that DSBF only needs one unlabeled source data to perform effective domain-specific bias filtering and domain invariance learning. The bias filtering can work better when given more unlabeled source domains, which we attribute to the invariance learning of the multi-source data under the inter-domain attention mechanism.

5.3 Results for the CDG Task

We report the results for the CDG task on PACS and Office-Home datasets in Table 4. We observe that the proposed DSBF method achieves the highest average classification accuracy on both PACS and Office-Home datasets, and performs the best on more than half CDG sub-tasks on Office-Home dataset. DSBF method has excellent performance in effectively training a generalizable model not only in the challenging SLDG task but also in the CDG task, which illustrates the versatility of the proposed domain-specific bias filtering strategy that domain-specific bias of one domain can be filtered out by effectively employing the data of other source domains.

5.4 Analysis

5.4.1 Semantic Invariance Learning

Figure 6 shows the semantic information learned by supervised learning method ERM (Vapnik, 1992) (using only the labeled source data) and our method DSBF (using both the labeled and unlabeled source data). We find that DSBF employs more effective regions of the images for visual recognition, but ERM fails to pay attention to the most effective regions. It demonstrates that DSBF makes full use of the

Table 1 Classification accuracy (%) for the Single Labeled Domain Generalization (SLDG) task on PACS dataset

Methods	Type	Ca→Ar	Sk→Ar	Ph→Ar	Ar→Ca	Sk→Ca	Ph→Ca	Ar→Sk	Ca→Sk	Ph→Sk	Ar→Ph	Ca→Ph	Sk→Ph	Avg.
ERM (Vapnik, 1992)	SL	65.25±2.20	25.88±2.53	**66.65** ± 2.80	57.89 ± 1.09	53.37 ± 4.59	25.53 ± 1.27	51.34 ± 1.85	65.00 ± 3.07	30.52 ± 0.93	**97.01** ± 3.67	87.13 ± 0.29	37.72 ± 1.98	55.27 ± 0.74
Mean Teacher (Tarvainen & Valpola, 2017)	SSL	30.66 ± 1.83	17.43 ± 2.75	28.61 ± 0.80	33.36 ± 1.44	21.50 ± 0.91	28.16 ± 1.53	25.78 ± 2.50	38.20 ± 2.51	29.07 ± 1.77	42.04 ± 2.00	37.07 ± 1.30	16.71 ± 2.45	29.05 ± 0.68
FixMatch (Sohn et al., 2020)		30.03 ± 5.41	14.45 ± 1.69	22.85 ± 0.31	31.66±2.13	24.79 ± 1.50	25.64 ± 3.74	19.64 ± 1.46	39.07 ± 1.28	29.68 ± 1.04	25.87 ± 0.98	33.95 ± 0.83	14.31 ± 1.81	26.00 ± 1.03
DAN (Long et al., 2015)	UDA	62.8 ± 3.0	46.9 ± 3.3	58.3 ± 0.8	68.3 ± 0.8	62.5 ± 3.9	48.2 ± 1.0	55.1 ± 2.7	54.8 ± 0.6	31.3 ± 2.4	93.3 ± 1.9	85.0 ± 0.7	58.9 ± 2.9	60.5 ± 0.8
MCD (Saito et al., 2018)		72.8 ± 0.5	29.7 ± 1.3	66.0 ± 3.6	**69.7** ± 1.0	61.1 ± 4.2	43.6 ± 4.2	49.1 ± 2.6	61.6 ± 3.6	25.2 ± 2.0	95.9 ± 3.2	83.5 ± 4.6	44.2 ± 1.0	58.5 ± 0.4
MDD (Zhang et al., 2019b)		71.7 ± 1.4	40.2 ± 3.0	61.9 ± 3.0	67.0 ± 1.1	58.1 ± 2.2	**62.8** ± 6.3	44.8 ± 2.3	57.1 ± 3.8	33.5 ± 1.3	95.9 ± 2.1	85.1 ± 1.1	55.0 ± 3.7	61.1 ± 0.9
SHOT (Liang et al., 2020)		71.63 ± 1.07	58.37 ± 1.89	49.10 ± 1.27	66.30 ± 1.16	44.50 ± 1.87	57.30 ± 3.26	63.70 ± 0.31	61.40 ± 1.23	58.50 ± 1.88	82.20 ± 2.81	85.20 ± 0.26	62.10 ± 2.74	63.36 ± 0.29
OCDA (Liu et al., 2020)	MTDA	17.33 ± 4.15	18.51 ± 4.34	17.45 ± 0.87	26.07 ± 4.16	15.78 ± 0.32	19.71 ± 5.45	16.47 ± 1.14	19.67 ± 0.46	24.31 ± 2.50	26.43 ± 2.69	24.07 ± 0.66	11.32 ± 3.58	19.76 ± 0.43
BTDA (Chen et al., 2019)		63.58 ± 3.64	47.14 ± 5.65	54.95 ± 1.27	59.15 ± 3.36	35.74 ± 2.71	22.32 ± 0.17	**70.90** ± 2.20	64.79 ± 0.97	31.78 ± 1.69	42.76 ± 1.66	60.93 ± 0.22	48.76 ± 3.10	50.24 ± 0.45
GUD (Volpi et al., 2018)	DG	68.12 ± 0.94	22.75 ± 2.96	66.06 ± 1.06	68.17 ± 0.63	34.68 ± 1.00	26.11 ± 3.88	64.27 ± 2.01	68.44 ± 0.19	52.48 ± 0.26	95.57 ± 1.53	82.75 ± 0.69	36.53 ± 4.93	57.16 ± 0.65
JiGen (Carlucci et al., 2019)		66.60 ± 1.36	23.68 ± 0.97	64.36 ± 0.48	53.91 ± 1.61	33.45 ± 0.67	26.58 ± 0.82	50.24 ± 1.87	62.76 ± 1.48	28.58 ± 2.30	95.75 ± 0.36	84.31 ± 0.31	30.24 ± 1.86	51.71 ± 0.11
RSC (Huang et al., 2020)		64.79 ± 1.62	53.03 ± 2.66	66.50 ± 3.53	67.15 ± 1.97	**66.64** ± 3.89	26.58 ± 1.10	55.46 ± 1.32	**73.96** ± 5.88	44.06 ± 2.02	94.79 ± 0.08	82.10 ± 1.62	47.25 ± 1.56	61.86 ± 0.36
M-ADA (Qiao et al., 2020)		55.18 ± 2.77	24.41 ± 4.71	55.71 ± 2.82	62.33 ± 2.87	58.36 ± 5.28	35.92 ± 3.96	51.95 ± 4.36	72.28 ± 0.99	31.15 ± 3.44	84.43 ± 2.20	71.98 ± 2.15	34.92 ± 3.09	53.22 ± 0.59
DSBF	SLDG	**73.14** ± 0.71	**64.97** ± 5.10	52.93 ± 2.28	66.88 ± 1.38	45.72 ± 2.86	55.72 ± 2.07	68.54 ± 0.59	68.86 ± 1.61	**64.76** ± 2.79	87.26 ± 1.23	**89.28** ± 1.44	**67.40** ± 3.73	**67.12** ± 0.60

A→B represents using A, B, and other domains as the labeled source, target, and unlabeled source domains, respectively. The best results are emphasized in bold

Table 2 Classification accuracy (%) for the Single Labeled Domain Generalization (SLDG) task on Office-Home dataset

Methods	Type	Cl→Ar	Pr→Ar	Rw→Ar	Ar→Cl	Pr→Cl	Rw→Cl	Ar→Pr	Cl→Pr	Rw→Pr	Ar→Rw	Cl→Rw	Pr→Rw	Avg.
ERM (Vapnik, 1992)	SL	44.39 ± 1.66	42.65 ± 4.19	**58.12** ± 3.17	38.83 ± 0.97	33.96 ± 1.20	40.92 ± 0.90	58.14 ± 5.34	56.84 ± 3.88	**74.25** ± 0.57	67.84 ± 2.41	59.61±8.46	65.43 ± 0.14	53.42 ± 0.71
Mean Teacher (Tarvainen & Valpola, 2017)	SSL	8.41 ± 3.68	14.70 ± 5.11	2.84 ± 1.75	8.96 ± 3.59	12.41 ± 1.22	13.56 ± 0.43	2.23 ± 0.35	10.90 ± 1.07	29.29 ± 4.87	4.35 ± 0.64	6.34 ± 0.80	18.56 ± 0.09	11.05 ± 0.84
FixMatch (Sohn et al., 2020)		8.94 ± 1.46	7.50 ± 0.33	3.13 ± 1.06	8.18 ± 0.55	13.13 ± 1.07	16.91 ± 1.53	7.16 ± 6.55	14.69 ± 2.81	20.01 ± 0.24	2.27 ± 0.57	4.36 ± 3.87	21.25 ± 2.94	10.63 ± 0.80
DAN (Long et al., 2015)	UDA	46.6 ± 2.2	43.8 ± 6.1	58.1 ± 2.4	38.3 ± 3.1	33.8 ± 0.4	41.9 ± 2.3	58.8 ± 2.6	57.9 ± 1.1	73.0 ± 3.3	66.6 ± 0.4	59.5 ± 1.7	65.2 ± 0.6	53.6 ± 0.8
MCD (Saito et al., 2018)		42.3 ± 2.3	42.6 ± 0.9	58.1 ± 4.1	35.7 ± 2.7	32.3 ± 4.7	39.1 ± 2.9	56.3 ± 3.3	53.7 ± 4.2	72.6 ± 1.8	65.5 ± 0.9	55.4 ± 3.0	64.8 ± 3.4	51.5 ± 0.8
MDD (Zhang et al., 2019b)		45.9 ± 0.2	47.4 ± 0.3	56.7 ± 1.8	39.4 ± 2.7	34.6 ± 2.6	42.9 ± 2.4	59.6 ± 1.1	59.1 ± 1.9	72.8 ± 0.4	68.1 ± 4.6	61.3 ± 1.3	65.5 ± 1.1	54.4 ± 0.3
SHOT (Liang et al., 2020)		51.30 ± 4.32	48.73 ± 1.53	48.04 ± 0.40	39.98 ± 1.86	37.09 ± 2.73	39.89 ± 2.79	**63.08** ± 7.56	60.46 ± 0.69	63.05 ± 1.60	62.47 ± 3.02	63.44 ± 0.44	60.73 ± 3.51	53.19 ± 0.61
OCDA (Liu et al., 2020)	MTDA	13.26 ± 1.92	11.58 ± 1.97	20.60 ± 0.29	13.57 ± 1.59	25.40 ± 0.77	12.69 ± 0.67	11.04 ± 2.10	24.10 ± 1.91	11.97 ± 2.35	13.34 ± 2.51	18.46 ± 2.37	11.76 ± 5.51	15.65 ± 0.99
BTDA (Chen et al., 2019)		39.27 ± 1.90	**59.97** ± 5.07	50.57 ± 3.08	42.37 ± 0.81	**59.22** ± 0.71	**57.79** ± 0.25	43.95 ± 1.11	48.39 ± 9.33	51.84 ± 2.30	40.14 ± 0.57	47.74 ± 0.17	58.50 ± 0.53	49.98 ± 0.39
GUD (Volpi et al., 2018)	DG	42.31 ± 0.46	31.20 ± 1.02	53.03 ± 3.00	38.92 ± 3.89	35.46 ± 1.33	43.89 ± 1.40	51.79 ± 1.14	50.10 ± 2.06	71.71 ± 1.94	61.56 ± 1.39	53.22 ± 2.00	60.22 ± 0.93	49.45 ± 0.51
JiGen (Carlucci et al., 2019)		41.62 ± 1.92	38.20 ± 3.11	54.31 ± 4.71	36.20 ± 0.26	36.08 ± 0.94	42.29 ± 3.73	44.78 ± 1.31	53.57 ± 5.81	69.63 ± 3.91	55.57 ± 1.00	54.97 ± 0.24	62.54 ± 1.94	49.15 ± 0.33
RSC (Huang et al., 2020)		40.23 ± 0.28	37.58 ± 5.06	55.50 ± 2.30	39.54 ± 0.23	38.35 ± 0.72	46.94 ± 1.98	49.72 ± 3.14	52.29 ± 6.92	72.89 ± 1.33	63.16 ± 3.51	54.88 ± 1.30	57.44 ± 2.20	50.71 ± 0.78
M-ADA (Qiao et al., 2020)		30.08 ± 1.24	23.61 ± 2.65	44.21 ± 1.55	37.55 ± 1.01	33.68 ± 4.60	44.15 ± 0.40	43.10 ± 0.80	30.39 ± 1.40	63.05 ± 0.86	53.22 ± 1.11	44.50 ± 1.83	49.83 ± 5.35	41.45 ± 0.44
DSBF	SLDG	**51.37** ± 0.92	50.30 ± 0.99	56.90 ± 0.23	**43.81** ± 0.01	37.18 ± 0.34	42.43 ± 0.28	61.19 ± 0.59	**61.23** ± 1.29	71.71 ± 1.12	**68.58** ± 0.61	**64.30** ± 0.53	**67.84** ± 0.57	**56.40** ± 0.12

A→B represents using A, B, and other domains as the labeled source, target, and unlabeled source domains, respectively. The best results are emphasized in bold

Table 3 Classification accuracy (%) for the Single Labeled Domain Generalization (SLDG) task on DomainNet dataset

Methods	Type	Pa→Cl	Re→Cl	Sk→Cl	Cl→Pa	Re→Pa	Sk→Pa	Cl→Re	Pa→Re	Sk→Re	Cl→Sk	Pa→Sk	Re→Sk	Avg.
ERM (Vapnik, 1992)	SL	30.93 ± 1.50	34.73 ± 1.45	40.98 ± 1.39	29.42 ± 1.34	36.33 ± 0.76	32.95 ± 0.53	40.88 ± 0.76	47.04 ± 0.76	40.89 ± 1.20	28.99 ± 1.49	24.56 ± 1.44	25.73 ± 1.28	34.45 ± 0.61
Mean Teacher (Tarvainen & Valpola, 2017)	SSL	8.03 ± 0.45	7.45 ± 1.09	7.23 ± 0.17	6.34 ± 2.58	8.23 ± 0.07	8.34 ± 1.49	8.12 ± 1.26	10.23 ± 3.80	7.34 ± 2.06	7.21 ± 1.17	7.43 ± 0.22	7.03 ± 1.95	7.75 ± 0.27
FixMatch (Sohn et al., 2020)		6.37 ± 0.52	7.01 ± 0.35	5.28 ± 0.10	6.87 ± 0.63	8.40 ± 1.69	9.61 ± 1.90	9.33 ± 1.72	9.95 ± 0.22	8.34 ± 0.48	7.23 ± 0.53	5.67 ± 0.74	5.82 ± 0.68	7.49 ± 0.27
DAN (Long et al., 2015)	UDA	30.5 ± 0.4	40.7 ± 0.2	23.0 ± 0.2	26.0 ± 2.5	40.2 ± 0.5	**45.4** ± 2.3	39.0 ± 1.3	40.4 ± 0.6	32.2 ± 0.5	32.0 ± 2.4	28.2 ± 0.2	27.8 ± 1.5	33.8 ± 0.4
MCD (Saito et al., 2018)		31.0 ± 1.9	40.9 ± 0.3	25.4 ± 2.6	25.4 ± 0.9	39.7 ± 0.6	44.0 ± 1.3	39.1 ± 1.4	42.5 ± 0.4	32.1 ± 0.3	**35.6** ± 1.3	27.7 ± 0.1	30.3 ± 1.9	34.5 ± 0.4
MDD (Zhang et al., 2019b)		34.6 ± 2.2	**41.3** ± 0.4	28.1 ± 3.7	24.6 ± 3.7	**41.4** ± 3.2	44.4 ± 1.5	38.5 ± 3.7	44.2 ± 2.4	32.3 ± 0.1	35.1 ± 1.8	29.3 ± 0.9	**30.6** ± 1.7	35.4 ± 0.4
SHOT (Liang et al., 2020)		30.90 ± 1.57	40.06 ± 0.38	27.07 ± 4.06	27.49 ± 2.76	41.25 ± 2.21	43.99 ± 3.11	34.85 ± 3.10	44.10 ± 1.94	33.59 ± 2.46	33.68 ± 0.36	29.50 ± 1.43	29.48 ± 0.83	34.66 ± 0.19
OCDA (Liu et al., 2020)	MTDA	21.46 ± 2.43	34.19 ± 2.48	20.45 ± 0.84	18.61 ± 1.41	35.18 ± 0.13	21.40 ± 2.19	18.39 ± 0.22	21.51 ± 0.82	21.01 ± 1.27	11.11 ± 0.51	21.34 ± 1.40	27.21 ± 0.63	22.66 ± 0.34
BTDA (Chen et al., 2019)		19.30 ± 3.46	21.36 ± 1.45	24.63 ± 1.28	16.86 ± 0.84	20.86 ± 0.33	18.02 ± 1.30	16.78 ± 0.39	27.81 ± 0.47	19.82 ± 0.33	10.22 ± 0.66	11.46 ± 5.03	23.36 ± 0.80	19.21 ± 0.70
GUD (Volpi et al., 2018)	DG	32.23 ± 0.15	41.23 ± 1.02	26.83 ± 0.68	25.19 ± 3.45	39.95 ± 1.26	44.17 ± 0.97	37.89 ± 3.64	43.85 ± 1.56	34.56 ± 2.93	34.74 ± 0.10	28.60 ± 1.50	29.99 ± 0.92	34.94 ± 0.19
JiGen (Carlucci et al., 2019)		16.83 ± 0.53	32.19 ± 0.33	27.24 ± 0.33	14.19 ± 0.46	35.83 ± 0.31	17.88 ± 1.85	24.37 ± 1.12	32.03 ± 1.76	25.13 ± 0.55	20.82 ± 1.46	21.71 ± 0.69	25.74 ± 1.42	24.50 ± 0.21
RSC (Huang et al., 2020)		14.71 ± 1.49	29.81 ± 2.63	22.02 ± 0.50	12.00 ± 1.31	33.32 ± 0.36	13.94 ± 0.98	21.37 ± 4.53	29.48 ± 0.62	21.63 ± 0.89	17.74 ± 0.34	18.85 ± 1.06	23.13 ± 0.83	21.50 ± 0.28
M-ADA (Qiao et al., 2020)		9.62 ± 0.31	27.46 ± 1.63	17.37 ± 0.02	11.40 ± 1.18	32.19 ± 0.20	13.14 ± 0.80	17.01 ± 0.72	15.04 ± 0.80	14.26 ± 2.50	19.05 ± 2.25	10.16 ± 0.38	19.76 ± 1.63	17.21 ± 0.23
DSBF	SLDG	**35.30** ± 0.31	39.26 ± 0.79	**43.80** ± 0.31	**33.94** ± 0.66	38.50 ± 0.11	34.05 ± 1.30	**44.41** ± 0.43	**48.89** ± 0.82	**46.21** ± 0.15	33.75 ± 0.53	**29.73** ± 0.31	29.96 ± 0.39	**38.15** ± 0.23

A→B represents using A, B, and other domains as the labeled source, target, and unlabeled source domains, respectively. The best results are emphasized in bold

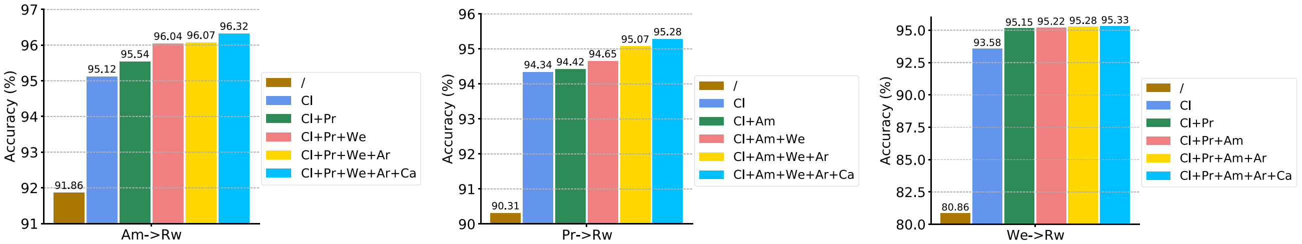


Fig. 5 Accuracy for the SLDG task on Office-Caltech-Home dataset. A→B represents using A, B, and other domains as the labeled source, target, and unlabeled source domains, respectively. We add one unlabeled source dataset each time from the unlabeled source domain set for each experiment. If no unlabeled source dataset is given (marked with " / "), the experiments are implemented in the supervised learning setting, i.e., using ERM (Vapnik, 1992) method.

Table 4 Classification accuracy (%) for the Conventional Domain Generalization (CDG) task on PACS and Office-Home datasets

Methods	PACS					Office-Home				
	Ar	Ca	Ph	Sk	Avg.	Ar	Cl	Pr	Rw	Avg.
DeepAll (Carlucci et al., 2019)	77.85	74.86	95.73	67.74	79.05	52.15	45.86	70.86	73.15	60.51
MMD-AAE (Li et al., 2018)	75.2	72.7	96.0	64.2	77.0	56.5	47.3	72.1	74.8	62.7
RSC (Huang et al., 2020)	83.43	**80.31**	95.99	80.85	85.15	58.42	47.90	71.63	74.54	63.12
CrossGrad (Shankar et al., 2018)	79.8	76.8	96.0	70.2	80.7	58.4	**49.4**	73.9	75.8	64.4
D-SAMs (D'Innocente & Caputo, 2018)	77.33	72.43	95.30	77.83	80.72	58.03	44.37	69.22	71.45	60.77
DSON (Seo et al., 2020)	**84.67**	77.65	95.87	**82.23**	85.11	59.37	44.70	71.84	74.68	62.90
JiGen (Carlucci et al., 2019)	79.42	75.25	**96.03**	71.35	80.51	53.04	47.51	71.47	72.79	61.20
DSBF	84.13 ± 0.12	79.32 ± 0.19	95.77 ± 0.32	81.58 ± 0.18	**85.20** ± 0.02	**60.65** ± 0.13	47.33 ± 0.04	**74.11** ± 0.08	**75.89** ± 0.26	**64.50** ± 0.04

The best results are emphasized in bold

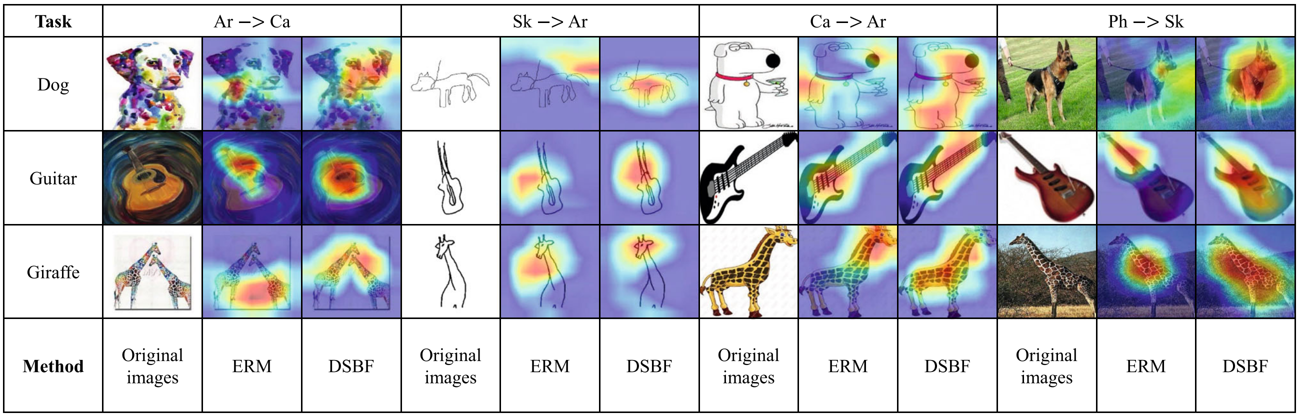


Fig. 6 Grad-CAM visualization (Selvaraju et al., 2017) of the semantic information learned by the supervised learning method ERM (Vapnik, 1992) and the proposed method DSBF. The regions in the darker red are considered more important for the trained model to perform object recognition (Color figure online).

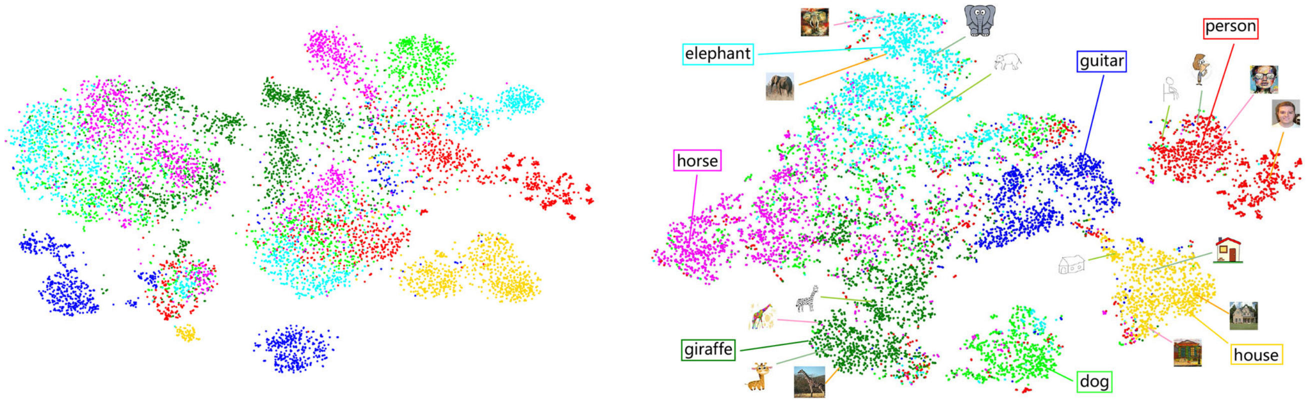


Fig. 7 T-SNE visualization of the distributions of the extracted semantic feature on PACS dataset (Ph→Sk), where each color represent a class. *Left*: After the model initialization stage. *Right*: After the bias filtering stage (Color figure online).

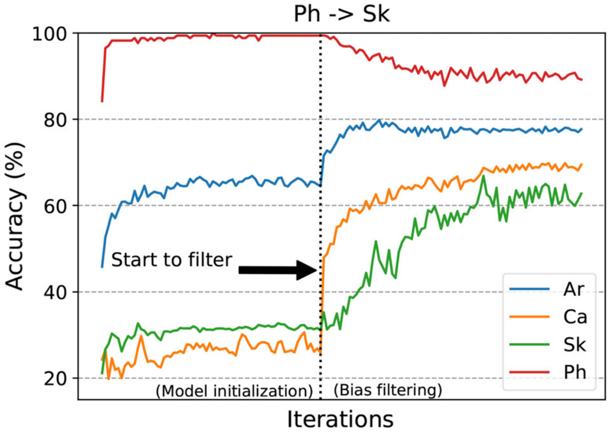


Fig. 8 Classification accuracy on PACS dataset during the model initialization stage and the bias filtering stage (the labeled source domain: Ph; the target domain: Sk; the unlabeled source domains: Ar and Ca).

unlabeled source data to filter out the domain-specific bias in the initialized model and capture the effective semantic information for accurate object recognition.

5.4.2 Semantic Feature Extraction

We then exploit t-SNE (Van der Maaten & Hinton, 2008) to analyze the semantic feature distributions after the model initialization stage and after the bias filtering stage as shown on the left and right of Fig. 7, respectively. It is evident that after bias filtering, DSBF extracts more discriminative semantic features of the data by making the same-class samples gather together. The bias filtering removes the bias in the initialized model and generates a more generalizable model.

5.4.3 Learning Process Tracking

We plot the learning process in Fig. 8. It is observed that: (1) In the model initialization stage, the trained model overfits the domain-specific bias of the labeled source data Ph (red line), and its classification accuracy rises rapidly. (2) In the bias filtering stage, the unlabeled data, i.e., Ar and Ca, are employed to filter out the domain-specific bias in both the feature extractor and classifier, the classification accuracy on the labeled source domain Ph thus drops slowly, while the performance on the unlabeled source domains Ar (blue line) and Ca (orange line) domains, as well as the unseen target domain Sk (green line), improves significantly. It clearly illustrates the learning process of DSBF, which first uses the labeled source data to initialize a discriminative model and then utilizes the unlabeled source data to filter out its bias and rectify the initialized model for improving its generalization ability.

5.4.4 Ablation Study

Table 5 shows the ablation results, where **DEB** is feature extractor debiasing, **REC** is classifier rectification, and **ATT** is the inter-domain attention module in the classifier rectification. We note that all the three parts, i.e., DEB, REC, and ATT, are important for DSBF to achieve the superior performance. We then observe that feature extractor debiasing obviously improves the performance on both datasets. It is probably because feature extractor debiasing trains the ResNet-18 network that has much more parameters for tuning than the one fully-connected (FC) layer of the classifier trained in classifier rectification (note that no matter how many parameters the attention module has, only one FC layer of the classifier is trained and will be used for testing the final performance). The attention shows its effectiveness in domain similarities learning and generalization improvement. It is also observed that DSBF w/o REC w/o ATT is better than SHOT (Liang et al., 2020) which uses the pseudo labels to train the classifier. It indicates that training classifier with pseudo labels could yield adverse effects in the SLDG task.

Table 5 Ablation study of classification accuracy (%) on PACS and Office-Home datasets

DEB		REC		ATT	PACS	Home
$\mathcal{L}_{IM}$	$\mathcal{L}_{CU}$	$\mathcal{L}_{FP}$	$\mathcal{L}_{BF}$			
					61.11	53.62
		✓	✓	✓	61.47	53.64
	✓	✓	✓	✓	65.34	54.34
✓		✓	✓	✓	65.75	55.11
✓	✓				65.99	55.89
✓	✓	✓			66.01	55.90
✓	✓		✓		66.21	55.92
✓	✓	✓	✓		66.99	55.97
✓	✓	✓	✓	✓	**67.12**	**56.40**

DEB feature extractor debiasing; *REC* classifier rectification; *ATT* the inter-domain attention module in the classifier rectification. The best results are emphasized in bold

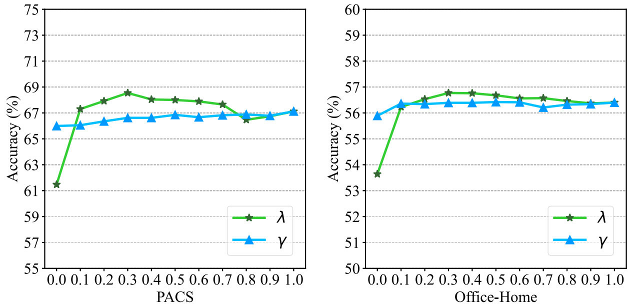


Fig. 9 Sensitivity analysis of the hyper-parameters λ and γ for the SLDG task, which are used for the feature extractor debiasing and the classifier rectification, respectively.

5.4.5 Sensitivity Analysis

We give sensitivity analysis by varying the hyper-parameters λ and γ for the SLDG and the CDG tasks in Figs. 9 and 10, respectively. For the SLDG task, it shows that the model performance is generally stable under different hyper-parameter settings. For the CDG task, the model prefers large value of λ but is insensitive to γ. We argue the reason is that the groud-truth labels are given directly in the CDG task, rather than obtained via clustering in the SLDG task. Thus, for the CDG task, the labels of the unlabeled data have higher reliability and the performance would be better when assigning larger weights, i.e., λ, to the model training with the ground-truth labels.

5.4.6 Clustering-Based Pseudo Labels

We further analyze the performance with different iterations of clustering-based pseudo label assignment. The results are shown in Table 6. We first observe that it is necessary to employ the clustering to obtain more accurate pseudo labels and achieve significantly better generalization performance. The second observation is that no further significant improvement can be achieved by performing more iterations. Therefore, based on this empirical experience, we may use

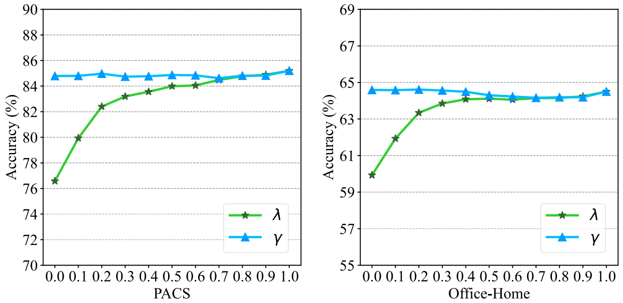


Fig. 10 Sensitivity analysis of the hyper-parameters λ and γ for the CDG task, which are used for the feature extractor debiasing and the classifier rectification, respectively.

Table 6 Average (**Avg.**) classification accuracy (%) with different clustering iterations (**Iter.**) on PACS and Office-Home datasets

Dataset	PACS				Office-Home			
Iter.	0	1	2	3	0	1	2	3
Avg.	58.28	67.12	66.87	**67.51**	53.30	**56.40**	56.16	56.02

Iteration is 0: directly employing the model predictions as the pseudo labels. The best results are emphasized in bold

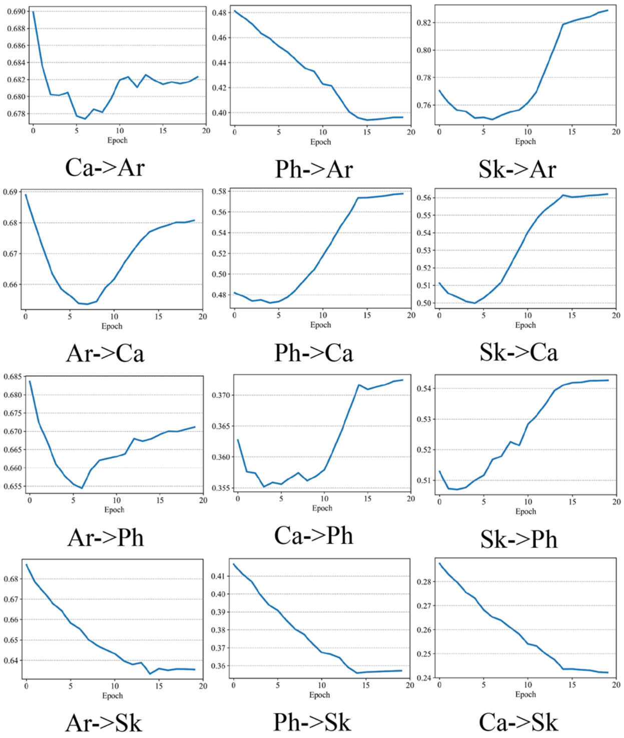


Fig. 11 Changes of the parameter α of the inter-domain attention module during training on the PACS dataset.

clustering to achieve better generalization performance, but it does not need to be iterated several times.

5.5 Trainable Weight Parameter of Attention Module

We show the changes of the parameter α (see Eq. 11) of the attention module in Fig. 11. Interestingly, it is observed that when given the same labeled source domain or the same target domain, the changes of α may show similar trend. For example, the three subfigures with the same labeled source domain of Sk, and the three subfigures with the same target domain of Sk. We argue that the reason for this phenomenon is that our attention module learns from the similarities among domains. When the labeled source domain or the target domain is given, the other domains may contain the similar common information for learning, which leads to the similar trend of α.

6 Conclusion

In this paper, we investigate a practical task to address the real-world problem of high annotation costs for generalizable model learning, i.e., Single Labeled Domain Generalization (SLDG), where only one of the multiple source domains is labeled. To tackle this challenging task, we propose a novel framework called Domain-Specific Bias Filtering (DSBF), which unifies the exploration of the labeled and the unlabeled source data, through a model initialization stage and a bias filtering stage, enhancing discriminability and generalization of the model. Extensive experiments on multiple datasets show the superior performance of DSBF for the SLDG task and the CDG task. In future work, we may extend our work to the scenarios with multimodal data.

Acknowledgements This work was supported in part by National Key Research and Development Program of China (2021YFC3340300), National Natural Science Foundation of China (U20A20387, Nos. 62006207, 62037001), Young Elite Scientists Sponsorship Program by CAST(2021QNRC001), Project by Shanghai AI Laboratory (P22KS00111), the Starry Night Science Fund of Zhejiang University Shanghai Institute for Advanced Study (SN-ZJU-SIAS-0010), Natural Science Foundation of Zhejiang Province (LZ22F020012, LQ21F020020), Fundamental Research Funds for the Central Universities (226-2022-00142, 226-2022-00051), National Key Research and Development Project (2022YFC2504605).

References

Bahdanau, D., Cho, K. & Bengio, Y. (2015). Neural machine translation by jointly learning to align and translate. In *International Conference on Learning Representations (ICLR)*.

Balaji, Y., Sankaranarayanan, S. & Chellappa, R. (2018). Metareg: Towards domain generalization using meta-regularization. In *Advances in Neural Information Processing Systems (NeurIPS),* pp. 998–1008.

Bellitto, G., Proietto Salanitri, F., Palazzo, S., et al. (2021). Hierarchical domain-adapted feature learning for video saliency prediction. *International Journal of Computer Vision (IJCV), 129*(12), 3216–3232.

Ben-David, S., Blitzer, J., Crammer, K., et al. (2010). A theory of learning from different domains. *Machine Learning, 79*(1–2), 151–175.

Blanchard, G., Lee, G., & Scott, C. (2011). Generalizing from several related classification tasks to a new unlabeled sample. *Advances in Neural Information Processing Systems (NeurIPS), 24*, 2178–2186.

Carlucci, F. M., D'Innocente, A., & Bucci, S., et al. (2019). Domain generalization by solving jigsaw puzzles. In *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR),* pp. 2224–2233.

Caron, M., Bojanowski, P., & Joulin, A., et al. (2018). Deep clustering for unsupervised learning of visual features. In *Proceedings of the European conference on computer vision (ECCV),* pp. 132–149.

Chen, Y., Wang, H., Li, W., et al. (2021). Scale-aware domain adaptive faster r-cnn. *International Journal of Computer Vision (IJCV), 129*(7), 2223–2243.

Chen, Z., Zhuang, J., & Liang, X., et al. (2019). Blending-target domain adaptation by adversarial meta-adaptation networks. In *IEEE/CVF conference on computer vision and pattern recognition (CVPR),* pp. 2243–2252.

Dai, D., Sakaridis, C., Hecker, S., et al. (2020). Curriculum model adaptation with synthetic and real data for semantic foggy scene understanding. *International Journal of Computer Vision (IJCV), 128*(5), 1182–1204.

Devlin, J., Chang, M. W., & Lee, K., et al. (2018). Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv.

Ding, Z., & Fu, Y. (2017). Deep domain generalization with structured low-rank constraint. *IEEE Transactions on Image Processing (TIP), 27*(1), 304–313.

Dou, Q., de Castro, D. C., & Kamnitsas, K., et al. (2019). Domain generalization via model-agnostic learning of semantic features. In *Advances in Neural Information Processing Systems (NeurIPS)*.

D'Innocente, A., & Caputo, B. (2018). Domain generalization with domain-specific aggregation modules. In *German Conference on Pattern Recognition,* Springer, pp. 187–198.

Fu, J., Liu, J., & Tian, H., et al. (2019). Dual attention network for scene segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR),* pp. 3146–3154.

Ganin, Y., Ustinova, E., Ajakan, H., et al. (2016). Domain-adversarial training of neural networks. *The Journal of Machine Learning Research (JMLR), 17*(1), 2096–2030.

Gholami, B., Sahu, P., Rudovic, O., et al. (2020). Unsupervised multi-target domain adaptation: An information theoretic approach. *IEEE Transactions on Image Processing (TIP), 29*, 3993–4002.

Gong, B., Shi, Y., & Sha, F., et al. (2012). Geodesic flow kernel for unsupervised domain adaptation. In *2012 IEEE conference on computer vision and pattern recognition,* IEEE, pp. 2066–2073.

Gong, B., Grauman, K., & Sha, F. (2013). Reshaping visual datasets for domain adaptation. In *Advances in Neural Information Processing Systems (NIPS)*.

Gong, B., Grauman, K., & Sha, F. (2014). Learning kernels for unsupervised domain adaptation with applications to visual object recognition. *International Journal of Computer Vision (IJCV), 109*(1), 3–27.

Gong, R., Li, W., & Chen, Y., et al. (2019). Dlow: Domain flow for adaptation and generalization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR),* pp. 2477–2486.

He, K., Zhang, X., & Ren, S., et al. (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR),* pp. 770–778.

Hinton, G., Vinyals, O., & Dean, J. (2015). Distilling the knowledge in a neural network. arXiv preprint arXiv:1503.02531

Ho, H. T., & Gopalan, R. (2014). Model-driven domain adaptation on product manifolds for unconstrained face recognition. *International Journal of Computer Vision (IJCV), 109*(1–2), 110–125.

Hoffman, J., Kulis, B., & Darrell, T., et al. (2012). Discovering latent domains for multisource domain adaptation. In *European Conference on Computer Vision (ECCV),* Springer, pp. 702–715.

Hoffman, J., Rodner, E., Donahue, J., et al. (2014). Asymmetric and category invariant feature transformations for domain adaptation. *International Journal of Computer Vision (IJCV), 109*(1–2), 28–41.

Huang, Y., Wu, Q., Xu, J., et al. (2021). Unsupervised domain adaptation with background shift mitigating for person re-identification. *International Journal of Computer Vision (IJCV), 129*(7), 2244–2263.

Huang, Z., Wang, H., & Xing, E. P., et al. (2020). Self-challenging improves cross-domain generalization. In *Proceedings of the European Conference on Computer Vision (ECCV),* pp. 124–140.

Kan, M., Wu, J., Shan, S., et al. (2014). Domain adaptation for face recognition: Targetize source domain bridged by common subspace. *International Journal of Computer Vision (IJCV), 109*(1–2), 94–109.

Kang, G., Jiang, L., & Yang, Y., et al. (2019). Contrastive adaptation network for unsupervised domain adaptation. In *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR),* pp. 4888–4897.

Kundu, J. N., Venkat, N., & Babu, R. V., et al. (2020). Universal source-free domain adaptation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR),* pp. 4544–4553.

LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *Nature, 521*(7553), 436–444.

Li, D., Yang, Y., & Song, Y. Z., et al. (2017). Deeper, broader and artier domain generalization. In *Proceedings of the IEEE international conference on computer vision (ICCV),* pp. 5542–5550.

Li, D., Zhang, J., & Yang, Y., et al.(2019). Episodic training for domain generalization. *Proceedings of the IEEE international conference on computer vision (ICCV),* pp. 1446–1455.

Li, H., Pan, S. J. & Wang, S., et al. (2018). Domain generalization with adversarial feature learning. In *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR),* pp. 5400–5409.

Li, H., Wang, Y. & Wan, R., et al. (2020a). Domain generalization for medical imaging classification with linear-dependency regularization. In *Advances in neural information processing systems (NeurIPS).*

Li, H., Wan, R., Wang, S., et al. (2021). Unsupervised domain adaptation in the wild via disentangling representation learning. *International Journal of Computer Vision (IJCV), 129*(2), 267–283.

Li, R., Cao, W., Wu, S., et al. (2020). Generating target image-label pairs for unsupervised domain adaptation. *IEEE Transactions on Image Processing (TIP), 29,* 7997–8011.

Li, Y., Hu, W., Li, H., et al. (2020). Aligning discriminative and representative features: An unsupervised domain adaptation method for building damage assessment. *IEEE Transactions on Image Processing (TIP), 29,* 6110–6122.

Liang, J., Hu, D., Feng, J., (2020). Do we really need to access the source data? Source hypothesis transfer for unsupervised domain adaptation. In *International conference on machine learning (ICML),* PMLR.

Lin, S., Li, C. T., & Kot, A. C. (2020). Multi-domain adversarial feature generalization for person re-identification. *IEEE Transactions on Image Processing (TIP), 30,* 1596–1607.

Liu, Z., Miao, Z. & Pan, X., et al. (2020). Open compound domain adaptation. In *IEEE/CVF conference on computer vision and pattern recognition (CVPR),* pp. 12, 403–12, 412.

Long, M., Cao, Y. & Wang, J., et al. (2015). Learning transferable features with deep adaptation networks. In *International conference on machine learning (ICML),* PMLR, pp. 97–105.

Long, M., Zhu, H. & Wang, J., et al. (2017). Deep transfer learning with joint adaptation networks. In *International conference on machine learning (ICML), PMLR,* pp. 2208–2217.

Long, M., Cao, Z. & Wang, J., et al. (2018). Conditional adversarial domain adaptation. In *Advances in neural information processing systems (NeurIPS),* pp. 1640–1650.

Mancini, M., Bulo, S. R., Caputo. B., et al. (2019a). Adagraph: Unifying predictive and continuous domain adaptation through graphs. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (CVPR),* pp. 6568–6577.

Mancini, M., Porzi, L. & Bulo. S. R., et al. (2019b). Inferring latent domains for unsupervised deep domain adaptation. *IEEE transactions on pattern analysis and machine intelligence (TPAMI).*

Matsuura, T., Harada, T. (2020). Domain generalization using a mixture of multiple latent domains. In *Proceedings of the AAAI conference on artificial intelligence (AAAI).*

Peng, X., Bai, Q. & Xia, X., et al. (2019). Moment matching for multi-source domain adaptation. In *Proceedings of the IEEE international conference on computer vision (ICCV),* pp. 1406–1415.

Qiao, F., Zhao, L., Peng, X. (2020). Learning to learn single domain generalization. In *Proceedings of the IEEE conference on computer vision and pattern recognition,* pp. 12,556–12,565.

Quionero-Candela, J., Sugiyama, M. & Schwaighofer, A., et al. (2009). *Dataset shift in machine learning*. The MIT Press.

Saito, K., Watanabe, K. & Ushiku, Y., et al. (2018). Maximum classifier discrepancy for unsupervised domain adaptation. In *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR),* pp. 3723–3732.

Schmidhuber, J. (1987). *Evolutionary principles in self-referential learning, or on learning how to learn: The meta-meta-... hook.* PhD thesis, Technische Universität München.

Selvaraju, R. R., Cogswell, M., Das, A., et al. (2017). Grad-cam: Visual explanations from deep networks via gradient-based localization. In *Proceedings of the IEEE international conference on computer vision (ICCV),* pp. 618–626.

Seo, S., Suh, Y., Kim, D., et al. (2020). Learning to optimize domain specific normalization for domain generalization. In *European conference on computer vision (ECCV).*

Shankar, S., Piratla, V., Chakrabarti, S., et al. (2018). Generalizing across domains via cross-gradient training. *International conference on learning representation (ICLR).*

Shen, Z., Huang, M., Shi, J., et al. (2021). Cdtd: A large-scale cross-domain benchmark for instance-level image-to-image translation and domain adaptive object detection. *International Journal of Computer Vision (IJCV), 129*(3), 761–780.

Sindagi, V. A., & Srivastava, S. (2017). Domain adaptation for automatic oled panel defect detection using adaptive support vector data description. *International Journal of Computer Vision (IJCV), 122*(2), 193–211.

Sohn, K., Berthelot, D., Carlini, N., et al. (2020). Fixmatch: Simplifying semi-supervised learning with consistency and confidence. In *Advances in neural information processing systems (NeurIPS).*

Tarvainen, A., Valpola, H. (2017). Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results. In *Advances in neural information processing systems (NeurIPS).*

Van der Maaten, L., & Hinton, G. (2008). Visualizing data using t-sne. *Journal of Machine Learning Research (JMLR), 9*(11), 2579–2605.

Vapnik, V. (1992). Principles of risk minimization for learning theory. In *Advances in neural information processing systems (NeurIPS)*, pp. 831–838.

Vaswani, A., Shazeer, N. & Parmar, N., et al. (2017). Attention is all you need. In *Advances in neural information processing systems (NeurIPS),* pp. 5998–6008.

Venkateswara, H., Eusebio, J. & Chakraborty, S., et al. (2017). Deep hashing network for unsupervised domain adaptation. In *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR),* pp. 5018–5027.

Volpi, R., Namkoong, H., Sener, O., et al. (2018). Generalizing to unseen domains via adversarial data augmentation. *Advances in neural information processing systems (NeurIPS).*

Wang, F., Jiang, M. & Qian, C., et al. (2017). Residual attention network for image classification. In *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR),* pp. 3156–3164.

Wang, J., Cheng, M. M. & Jiang, J. (2021). Domain shift preservation for zero-shot domain adaptation. *IEEE transactions on image processing (TIP).*

Wang, S., Yu, L., Li, C., et al. (2020a). Learning from extrinsic and intrinsic supervisions for domain generalization. In *Proceedings of the European conference on computer vision (ECCV).*

Wang, X., Kihara, D., Luo, J., et al. (2020). Enaet: A self-trained framework for semi-supervised and supervised learning with ensemble transformations. *IEEE Transactions on Image Processing (TIP), 30*, 1639–1647.

Wang, Y., Zhang, Z., Hao, W., et al. (2020). Attention guided multiple source and target domain adaptation. *IEEE Transactions on Image Processing (TIP), 30*, 892–906.

Wu, Z., Wang, X. & Gonzalez, J. E., et al. (2019). Ace: Adapting to changing environments for semantic segmentation. In *Proceedings of the IEEE/CVF international conference on computer vision (ICCV),* pp. 2121–2130.

Xiong, C., McCloskey, S. & Hsieh, S. H., et al. (2014). Latent domains modeling for visual domain adaptation. In *Proceedings of the AAAI conference on artificial intelligence (AAAI).*

Xu, H., Yang, M., Deng, L., et al. (2021). Neutral cross-entropy loss based unsupervised domain adaptation for semantic segmentation. *IEEE Transactions on Image Processing (TIP), 30*, 4516–4525.

Xu, J., Ramos, S., Vázquez, D., et al. (2016). Hierarchical adaptive structural svm for domain adaptation. *International Journal of Computer Vision (IJCV), 119*(2), 159–178.

Yamada, M., Sigal, L., & Chang, Y. (2014). Domain adaptation for structured regression. *International Journal of Computer Vision (IJCV), 109*(1–2), 126–145.

Yang, X., Song, Z. & King, I., et al. (2021). A survey on deep semi-supervised learning. arXiv preprint arXiv:2103.00550

Yasarla, R., Sindagi, V. A. & Patel, V. M. (2021). Semi-supervised image deraining using gaussian processes. *IEEE transactions on image processing (TIP).*

Yu, H., Hu, M. & Chen, S. (2018). Multi-target unsupervised domain adaptation without exactly shared categories. arXiv:1809.00852

Yuan, J., Ma, X. & Chen, D., et al. (2021a). Collaborative semantic aggregation and calibration for separated domain generalization. arXiv e-prints pp arXiv–2110

Yuan, J., Ma, X. & Kuang, K., et al. (2021b). Learning domain-invariant relationship with instrumental variable for domain generalization. arXiv preprint arXiv:2110.01438

Zhang, C., Zhang, K. & Li, Y. (2020a). A causal view on robustness of neural networks. In *Advances in neural information processing systems (NeurIPS).*

Zhang, H., Goodfellow, I. & Metaxas, D., et al. (2019a). Self-attention generative adversarial networks. In *International conference on machine learning (ICML), PMLR,* pp. 7354–7363.

Zhang, K., Gong, M. & Schölkopf, B., et al. (2015). Multi-source domain adaptation: A causal view. In *AAAI conference on artificial intelligence (AAAI),* pp. 3150–3157.

Zhang, Y., Liu, T. & Long, M., et al. (2019b). Bridging theory and algorithm for domain adaptation. In *International conference on machine learning (ICML).*

Zhang, Y., Wei, Y., Wu, Q., et al. (2020). Collaborative unsupervised domain adaptation for medical image diagnosis. *IEEE Transactions on Image Processing (TIP), 29*, 7834–7844.

Zhao, H., Zhang, S., Wu, G., et al. (2018). Adversarial multiple source domain adaptation. *Advances in Neural Information Processing Systems (NeurIPS), 31*, 8559–8570.

Zhao, S., Gong, M. & Liu, T., et al. (2020) Domain generalization via entropy regularization. In *Advances in neural information processing systems (NeurIPS).*

Zhao, S., Li, B. & Xu, P., et al. (2021). Madan: Multi-source adversarial domain aggregation network for domain adaptation. *International Journal of Computer Vision (IJCV),* pp. 1–26.

Zheng, Z., & Yang, Y. (2021). Rectifying pseudo label learning via uncertainty estimation for domain adaptive semantic segmentation. *International Journal of Computer Vision (IJCV), 129*(4), 1106–1120.

Zhou, K., Yang, Y. & Hospedales, T., et al. (2020). Learning to generate novel domains for domain generalization. In *European conference on computer vision (ECCV),* pp. 561–578.

Zhou, K., Loy, C. C. & Liu, Z. (2021a). Semi-supervised domain generalization with stochastic stylematch. arXiv preprint arXiv:2106.00592

Zhou, K., Yang, Y., Qiao, Y., et al. (2021). Domain adaptive ensemble learning. *IEEE Transactions on Image Processing, 30*, 8008–8018.

Zhou, K., Yang, Y. & Qiao, Y., et al. (2021c). Domain generalization with mixstyle. In *International conference on learning representations (ICLR).*

Zuo, Y., Yao, H., & Xu, C. (2021). Attention-based multi-source domain adaptation. *IEEE Transactions on Image Processing, 30*, 3793–3803.

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.