ECAI 2023
K. Gal et al. (Eds.)

doi:10.3233/FAIA230450

Letting Go of Self-Domain Awareness: Multi-Source Domain-Adversarial Generalization via Dynamic Domain-Weighted Contrastive Transfer Learning

Yuan Ma[a,b,c], **Yiqiang Chen**[a,b,c;*], **Han Yu**[d], **Yang Gu**[a,b,c], **Shijie Wen**[a,b,c] **and Shuai Guo**[a,b,c]

[a]Institute of Computing Technology, Chinese Academy of Sciences, Beijing, China
[b]University of Chinese Academy of Sciences, Beijing, China
[c]Beijing Key Laboratory of Mobile Computing and Pervasive Device, Beijing, China
[d]Nanyang Technological University, Singapore

Abstract. Domain generalization (DG), which aims to learn a model that can generalize to an unseen target domain, has recently attracted increasing research interest. A major approach is to learn domain invariant representations to avoid greedily capturing all the correlations found in source domains caused by empirical risk minimization. Nevertheless, overly emphasizing learning of domain invariant representations might lead to learning overly-compressed domain invariant representations, causing confusion of different classes in a same domain. To address this limitation, we introduce a novel dynamic domain-weighted contrastive loss, which maximizes the subdomain differences between different classes especially those belonging to the same domain, while minimizing the average distance between the points of the convex hull of the aligned source domains. We propose **M**ulti-**s**ource domain-adversarial generalization via dynamic domain-weighted **C**ontrastive **tr**ansfer **l**earning (MsCtrl), a novel domain-adversarial generalization framework, which optimizes the distribution alignment of source and potential target subdomains in an adversarial manner under the "control" of the aforementioned contrastive loss. Extensive experiments based on real-world datasets demonstrate significant advantages of MsCtrl over existing state-of-the-art methods.

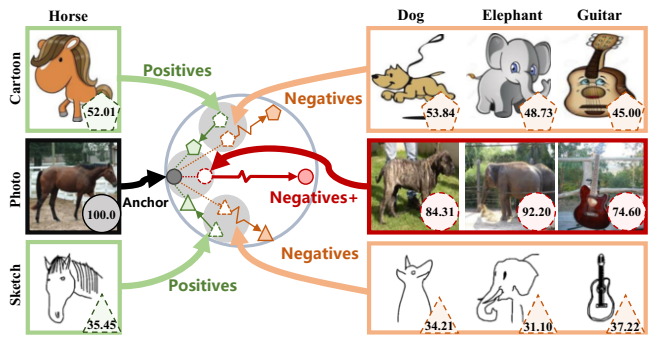

Figure 1. Supervised learning on PACS. The normalized classification responses of each class for the photo horse (the anchor) are shown in the bottom right corner of each subplot. Obviously, the classes that obtained high classification responses from the trained classifier belong to the same domain (photo) as the anchor. However, the domain invariant representations need to break the propinquity of the instances of different classes in the same domain and narrow the instances of the same class in different domains.

1 Introduction

Machine learning technologies have significantly improved how a diverse range of problems are solved in our society. One of the most basic assumptions of traditional machine learning is that training data (the source domain) and test data (the target domain) are independent and identically distributed (i.i.d.). Under this assumption, minimizing the training error optimizes model performance on the test set. However, the target domain collected in the novel circumstances can be different from that of the source domain. Transfer learning has been proposed to address this issue. In addition, labeled data can also be collected from different circumstances, resulting in source domains following distinct data distributions. It has been found that models trained on the training data consists of data that follows multiple distributions with significant variations tend to over-fit, which makes the generalization on unseen target domains a challenging research problem.

* Corresponding Author. Email: yqchen@ict.ac.cn

Domain generalization (DG) has been proposed to address this problem. It seeks to minimize the generalization risk against potential data shifts. A widely adopted DG approach is to learn domain invariant representations [15, 14, 12]. And a number of existing literature [8, 19] has provided theoretical guarantees for this field. Labeled data from different sources are often collected to obtain diverse distributions to support the learning of invariable correlation features (i.e., the essential features) with regard to the target objects. In this way, a robust model which can also be applied to an unseen target domain can be obtained.

Excessive emphasis on the learning of domain invariant features, however, may result in learning overly-compressed domain invariant representations [19]. That is, the representations for different classes from the same domain can be mixed up. This degrades model performance in both sources and targets. While minimizing the domain discrepancy within the same class, it is desirable to maximize the domain discrepancy between different classes simultaneously. Consequently, we prioritize discovering the invariant representations of a

subdomain, which is specified by a set of instances belonging to both the same class and the same domain. The invariant representations of the subdomains with the same class are what we aim to find.

And as shown in Fig.1, instances from the same domain show high similarities, making them closer to each other than other classes. Nevertheless, there is a strong need to "push" representation points (the red circle in Fig.1) from different classes further apart within the same domain. In addition to the fact that the representations of the same domain are close, the overfitting of positive pairs in the original contrastive learning [30] leads to the class-domain-coupling of the positive pairs, which is also the key to the existence of the overly-compressed problem. From the perspective of information theory, for the goal of obtaining the best classification accuracy on the unknown target domain, the domain attribute of the source instances is irrelevant noise.

With this motivation in consideration, we propose a dynamic domain-weighted contrastive loss function that eliminates the class-domain-coupling by selecting appropriate positive pairs and the self-domain awareness of the anchor by focusing on the contrast between the negative instances in the self-domain. Geometrically, this enhancement minimizes the average distance between elements of the convex hull of the aligned subdomain, making it more appropriate for domain generalization tasks.

Moreover, we propose the **M**ulti-**s**ource domain-adversarial generalization via **C**ontrastive **tr**ansfer **l**earning (MsCtrl) approach with the aim to train the task classifiers and a feature extractor in an adversarial manner under the **"control"** of minimizing the convex hull of the aligned subdomain distributions (through the aforementioned improved contrastive loss) and the source risk. Because the target domain is unknown in DG tasks and it is impossible to directly quantify the representation discrepancy between the source and target domains, we perform image augmentation on the source domain to mimic the potential target domain instances to appropriately evaluate MsCtrl's performance and implement extrapolation.

In summary, the main contributions of our work are three-fold:

- Firstly, we propose a new dynamic domain-weighted contrastive loss. Compared to the conventional contrastive loss, the dynamic domain-weighted contrastive loss significantly reduce the diameter of the convex hull of the aligned subdomain after letting go of the self-domain awareness, achieving stricter feature alignment at the subdomain-wise.
- Secondly, we propose MsCtrl, a novel multi-source domain-adversarial generalization framework. The distribution of source domains and potential target domain are aligned by adjusting task-specific decision boundaries against the "control" of minimizing source risk and multi-source dynamic domain-weighted contrastive loss. Furthermore, theoretical analysis offers the rationality of our method from another perspective.
- We evaluate the proposed MsCtrl method on three domain generalization benchmarks and demonstrate the validity of each component of the model through analysis and ablation studies. The results show that MsCtrl is highly competitive compared to other existing state-of-the-art methods.

2 Related Work

2.1 Unsupervised Domain Adaptation

Unsupervised domain adaptation (UDA) aims to transfer knowledge learned from a labeled domain to an unlabeled target domain. Inspired by Generative Adversarial Networks (GANs) [6], classic adversarial domain adaptation methods such as DANN [5], ADDA [27] and RADA [33] attempt to learn domain invariant representations that the domain discriminator cannot distinguish. MCD [22] uses the adversarial training procedure of two classifiers with mini-max discrepancy on the target samples and generates the feature with minimum discrepancy to achieve desirable performance.

2.2 Multi-Domain Generalization

When there are multiple source domains with different distributions, it becomes a multi-source domain adaptation (MSDA)[32] or domain generalization (DG) [31] problem. The inability to obtain target domain distribution makes DG more challenging. DGs attempt to learn a model with strong generalization ability from multiple source domains that can adapt properly to invisible target domains through various strategies.

2.2.1 Domain randomization

Domain randomization methods [9, 11, 12, 36] direct the model to learn domain invariant features by randomizing or stylizing the source instances to simulate the wider distribution related to the source domains.

2.2.2 Domain adversarial learning

The adversarial method in UDA has also inspired many DG methods [17, 38]. Since the target domain is not known, the existing adversarial domain alignment methods generally align multiple source distributions. However, enforced general domain alignment in traditional adversarial methods might learn overly-compressed domain invariant representations with ambiguous category classification boundaries. Our method is free from overcompression since we directly define a dynamic weighted loss based on contrastive learning [35, 3] to achieve subdomain alignment. In UDAs, subdomain alignment can often achieve better adaptive performance than the conventional approaches of aligning the whole domain [39]. However, in DGs, as the target domain information is lacking, it has been proven that general domain invariant representations are helpful to increase the span of latent representations [19] notably for the distant target domain. Therefore, subdomain alignment is more difficult to generalize to the unseen domains. The present effort is an attempt to simulate the potential target domain instances by augmenting the source instances to achieve model extrapolation through adversarial training of multi-source domain and potential targets.

2.3 Contrastive Learning

Contrastive learning aims to learn a feature extractor that maps views of the same input to similar features while mapping other views to distinguishable features [3]. A more relevant line of contrastive learning is how to select positive [26, 24] and negative pairs [20, 1]. And in this paper, we bridge the gap between contrastive learning and transfer learning and explore the optimal positive and negative pairs selection for the contrastive transfer learning setting.

3 The Proposed MsCtrl Method

3.1 Notations

We start with the problem definition. We consider the classification problem of the source and unseen target domains with the same

label set $\mathcal{Y} = \{y_i\}_{i=1}^{C}$, where C is the number of classes. Let there be K source domains $\{S_1, S_2, \cdots, S_K\}$ over the common data space $\mathcal{X}$. The k-th fully labeled source domain is denoted as $\mathcal{D}_s^k = \{(\mathbf{x}_i^k, y_i)\}_{i=1}^{M_{s_k}}$, where $y_i \in \mathcal{Y}$ stands for the label, and M_{s_k} stands for the size of the k-th source domain. Denote the information required for label classification as Y, for domain classification as D and for k-th domain classification as D_k. $I(A;B)$ denote mutual information. We define a stochastic image augmentation function $\mathcal{M}(\cdot)$ that transforms $\mathbf{x} \in \mathcal{X}$ into augmented version $\widetilde{\mathbf{x}} = \mathcal{M}(\mathbf{x})$.

We define F is the feature extraction function. G^1 and G^2 are two classifiers, which take the features $\mathbf{z} \in \mathcal{Z}$ extracted from feature extraction as input. $P_F^{S_i}$ is the representation distribution. We use $\circ$ to denote the function composition operator. Thus, the hypothesis family on the source and target domains can be defined as $\mathcal{H} := \{h(\cdot) = G \circ F(\cdot) \colon \mathcal{X} \xrightarrow{F} \mathcal{Z} \xrightarrow{G} \mathcal{Y}_\triangle\}$, where $\mathcal{Y}_\triangle := \{\boldsymbol{\pi} \in \mathbb{R}^C : \|\boldsymbol{\pi}\|_1 = 1 \wedge \boldsymbol{\pi} \geq 0\}$ is the $(C-1)$-simplex. $\odot$ denotes the logic operation XNOR, where $a \odot b = 1$ when a and b are equal; otherwise, $a \odot b = 0$.

3.2 Dynamic Domain-weighted Contrastive Loss

In contrastive learning, positive pairs provide supervised information, while negative pairs provide counterexamples. As shown in Fig. 2, we demonstrate the disparity between the conventional contrastive loss $\mathcal{L}_{\text{cl}}$ and the dynamic domain-weighted contrastive loss $\mathcal{L}_{\text{ms}}$.

Firstly, a randomly selected batch of instances are used as the anchors $\{(\mathbf{x}_i^{k_i}, y_i)\}_{i=1}^{N}$. We then further select and concatenate the corresponding positive instances batch $\{\{(\mathbf{x}_i^{k_i}, y_i)\}, \{(\mathbf{x}_{i+N}^{k_{i+N}}, y_{i+N})\}\}_{i=1}^{N}$ which satisfies the requirements of $y_i = y_{i+N}$ and $k_i \neq k_{i+N}$.

For each anchor, we partition the set S of all the instances in the same batch into four disjoint subsets $S_{\text{pos-}}$, $S_{\text{pos+}}$, $S_{\text{neg-}}$ and $S_{\text{neg+}}$. Binary matrices $\mathbf{M}_{\text{pos-}}$, $\mathbf{M}_{\text{pos+}}$, $\mathbf{M}_{\text{neg-}}$, $\mathbf{M}_{\text{neg+}}$ are then designed for each pair of the union of these four disjoint subsets:

- $S_{\text{pos}} = S_{\text{pos-}} \cup S_{\text{pos+}}$ is the subset of instances that have the same label as the anchor. For each anchor $\mathbf{x}_i^{k_i}$, the number of instances corresponding to S_{pos} is $m_0(\mathbf{x}_i^{k_i}) = \sum_{j=1}^{2N} \mathbf{M}_{\text{pos}}(\mathbf{x}_i^{k_i}, \mathbf{x}_j^{k_j})$. The algebraic description of this subset is:

$$\mathbf{M}_{\text{pos}}(\mathbf{x}_i^{k_i}, \mathbf{x}_j^{k_j}) = y_i \odot y_j, \ \ \mathbf{M}_{\text{pos-}}(\mathbf{x}_i^{k_i}, \mathbf{x}_j^{k_j}) = (\neg k_i \odot k_j) \wedge (y_i \odot y_j). \quad (1)$$

- $S_{\text{neg+}}$ is the subset of instances from the same domain but with different labels from the anchor. Define $m_2(\mathbf{x}_i^{k_i}) = \sum_{j=1}^{2N} \mathbf{M}_{\text{neg+}}(\mathbf{x}_i^{k_i}, \mathbf{x}_j^{k_j})$. The algebraic description is:

$$\mathbf{M}_{\text{neg+}}(\mathbf{x}_i^{k_i}, \mathbf{x}_j^{k_j}) = (k_i \odot k_j) \wedge (\neg y_i \odot y_j). \quad (2)$$

- The other subset $S_{\text{neg-}}$ is the subset of instances from the different domains and labels from the anchor with a size of $m_1(\mathbf{x}_i^{k_i}) = 2N - m_0(\mathbf{x}_i^{k_i}) - m_2(\mathbf{x}_i^{k_i})$, can be described as:

$$\mathbf{M}_{\text{neg-}}(\mathbf{x}_i^{k_i}, \mathbf{x}_j^{k_j}) = (\neg k_i \odot k_j) \wedge (\neg y_i \odot y_j). \quad (3)$$

Inspired by the information bottleneck theorem [25], for domain generalization setting, we have:

Theorem 1 *Suppose f_{op} is a mutual information sufficient encoder[1], Y and D are independent (See 3.1 for definition). For tasks that generalize to arbitrary given unknown domain, the optimal positive pairs $(\mathbf{x}_i^{k_i, y_i}, \mathbf{x}_j^{k_j, y_j})$ selection are:*

$$(\mathbf{x}_i^*, \mathbf{x}_j^*) = \underset{\mathbf{x}_i, \mathbf{x}_j}{\operatorname{argmin}} -\kappa I(\mathbf{x}_i; \mathbf{x}_j; Y) + I(\mathbf{x}_i; \mathbf{x}_j; D) + I(\mathbf{x}_i; \mathbf{x}_j | Y, D). \quad (4)$$

[1] The encoder $(\mathbf{z}_i, \mathbf{z}_j) = f_{op}((\mathbf{x}_i, \mathbf{x}_j))$ is optimal for preserving the mutual information of positive pairs, namely $I(\mathbf{x}_i; \mathbf{x}_j) = I(\mathbf{z}_i; \mathbf{z}_j)$.

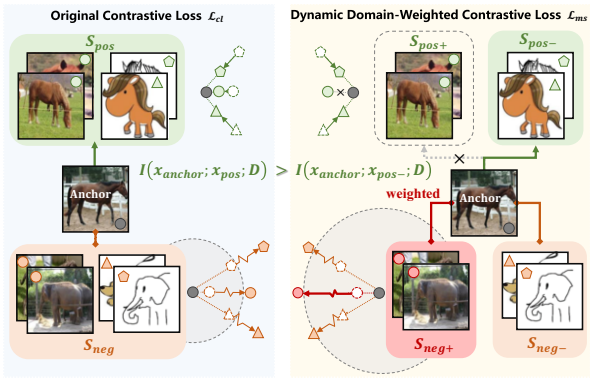

Figure 2. Comparison of the original super contrastive loss and our proposed dynamic domain-weighted contrastive loss.

Different from the general encoder, the mutual information sufficient encoder obtains the optimal value by selecting the most suitable positive instance pairs. Since Y and D are independent and $I_{(i,j)\in S_{\text{pos+}}}(\mathbf{x}_i; \mathbf{x}_j; D) > I_{(i,j)\in S_{\text{pos-}}}(\mathbf{x}_i; \mathbf{x}_j; D)$ holds almost surely. We can conclude that choosing positive pairs from $S_{\text{pos-}}$ will result in a better model performance than the ones from S_{pos} or $S_{\text{pos+}}$.

The proposed dynamic domain-weighted contrastive loss $\mathcal{L}_{\text{ms}}$ refines positive and negative pairs based on domain attributes and selects them effectively. **Positive pairs selection strategy:** Inspired by Thm. 1, only (i,j) in the subset $S_{\text{pos-}}$ are selected as the positive pairs. **Negative instances weighting strategy:** We select negative pairs from $S_{\text{neg-}}$ and $S_{\text{neg+}}$, with greater weight for $S_{\text{neg+}}$. The probability that the $S_{\text{neg+}}$ is an empty set (namely $m_2 = 0$) decreases with increasing batch size, and is almost 0 when the batch size is significantly larger than the number of domains. For mathematical form unification, we give the added weight to all negative instances in this case. Therefore, the dynamic domain-weighted matrix can be formulated as:

$$\mathbf{w}(i,j) = \begin{cases} \mathbf{M}_{\text{neg-}}(i,j) \dfrac{C_{\text{w}}}{2N - m_0(i)}, & m_2(i) = 0 \\ \mathbf{M}_{\text{neg-}}(i,j) + \mathbf{M}_{\text{neg+}}(i,j) \dfrac{C_{\text{w}} - m_1(i)}{m_2(i)}, & m_2(i) \neq 0 \end{cases} \quad (5)$$

where $C_{\text{w}} = \eta(2N-2)$, and $\eta \geq 1$ is an adjustable hyper-parameter that controls the magnification of the $S_{\text{neg+}}$ instances. The $\mathbf{w}(i,j)$ for S_{pos} is set to 0 to avoid the negative-positive-coupling effect.

Extended with the dynamic domain-weighted factor $\mathbf{w}$, the pairwise contrastive loss can be defined as:

$$\ell_{\text{ms}}(i,j) = -\log \frac{\exp(\text{sim}(\mathbf{z}_i^{k_i}, \mathbf{z}_j^{k_j})/\tau)}{\sum_{l=1}^{2N} \mathbf{w}(i,l) \exp(\text{sim}(\mathbf{z}_i^{k_i}, \mathbf{z}_l^{k_l})/\tau)}. \quad (6)$$

where $\text{sim}(\mathbf{u}, \mathbf{v}) = (\mathbf{u} \cdot \mathbf{v})/(\|\mathbf{u}\| \cdot \|\mathbf{v}\|)$ is the pairwise normalized cosine similarity, belonging to the closed interval $[-1, 1]$.

As a consequence, the dynamic domain-weighted contrastive loss function $\mathcal{L}_{\text{ms}}$ can be formalized as:

$$\mathcal{L}_{\text{ms}} = \frac{1}{2N} \Sigma_{i=1}^{N} [\ell_{\text{ms}}(i, i+N) + \ell_{\text{ms}}(i+N, i)]. \quad (7)$$

As shown in Step 1 in Fig.3, the goal of the dynamic domain-weighted contrastive loss is to reduce the representation discrepancy within subdomains of the same class and increase the distinction

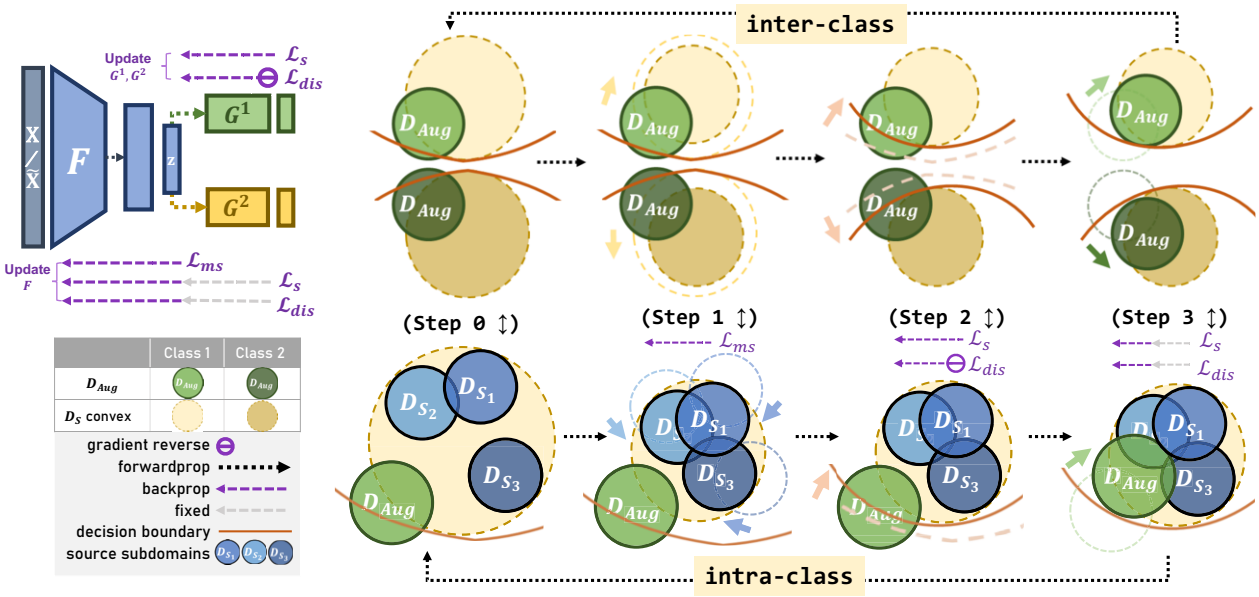

Figure 3. An overview of the proposed method. **Left**: The proposed architecture includes a feature extractor (blue) and two classifiers (green and yellow). The dashed arrows represent the gradient flows. **Right**: The top loop is the inter-class variation, and the bottom is the corresponding intra-class variation. (0) Leftmost is the initial state of a loop. (1) $\mathcal{L}_{\text{ms}}$ aligns subdomains of multi-source domains and makes the centroid distance between the subdomain convex hulls larger. (2) Train two classifiers to maximize the discrepancy on the augmented domain. (3) Reverse gradient and train the extractor to minimize the discrepancy.

between aligned subdomains. Compared to conventional contrastive loss, our proposed loss $\mathcal{L}_{\text{ms}}$, after letting go of self-domain awareness of the anchor, can obtain subdomain representations with tighter convex hulls and greater dispersion between convex hulls.

3.3 MsCtrl Architecture Design

Here, we introduce the architecture design of MsCtrl in detail (as illustrated in Fig.3).

In Step 1, $\mathcal{L}_{\text{ms}}$ is used for the subdomain alignment in the source domains. Let L_{ce} be the cross entropy loss, and $\mathcal{L}_{\text{s}}$ be the average cross entropy loss of the original instances and the augmented instances on the two classifiers, we have:

$$\mathcal{L}_{\text{s}} = \frac{1}{4}\mathbb{E}_{\mathbf{x}\sim\mathcal{X}}[L_{\text{ce}}(G^i\circ F(\mathbf{x}), y)+L_{\text{ce}}(G^i\circ F(\widetilde{\mathbf{x}}), y)]_{i=1}^{2}. \tag{8}$$

where simplexes $G^i\circ F(\cdot)\in\mathcal{Y}_{\triangle}$.

Then, we employ randomly augmented source instances to simulate the potential target domain. Since the augmented instances transform randomly, we assume that it is possible for a new augmented instance to obtain different results from the two classifiers.

$$\mathcal{L}_{\text{dis}} = \frac{1}{C}\mathbb{E}_{\mathbf{x}\sim\mathcal{X}}\left\|[p(y|G^1\circ F(\widetilde{\mathbf{x}}))-p(y|G^2\circ F(\widetilde{\mathbf{x}}))]_{y=1}^{C}\right\|_1. \tag{9}$$

To train the extractor and classifiers, we repeat the following two steps to update them throughout the training process:

- Fix the parameters of the two classifiers G^1 and G^2. Update the feature extractor F to minimize the empirical risk $\mathcal{L}_{\text{s}}$, the dynamic domain-weighted contrastive loss $\mathcal{L}_{\text{ms}}$, and the output discrepancy of two classifiers $\mathcal{L}_{\text{dis}}$:

$$\min_{F}\ \mathcal{L}_{\text{ms}} + \mathcal{L}_{\text{adv}}. \tag{10}$$

$$\text{with}\quad \mathcal{L}_{\text{adv}} = \alpha\mathcal{L}_{\text{s}} + (1-\alpha)\mathcal{L}_{\text{dis}}. \tag{11}$$

  α trade-off the source risk and the classifier adversary.

- Then, the parameters of the feature extractor are fixed and only the G^1 and G^2 parameters are trainable. The optimization objective of the two classifiers is to come up with different decision boundaries on augmented domains as far as possible, under the condition that source instances are correctly classified (Step 2 in Fig. 3):

$$\min_{G^1,G^2}\ \mathcal{L}_s - \beta\mathcal{L}_{\text{dis}}. \tag{12}$$

  where β is balance weight.

3.4 Theoretical Analysis

The key idea behind MsCtrl is to reduce the underlying target risk under the "control" of minimizing source risk and the source-source representation discrepancy. In this section, we provide a generalization risk analysis for the MsCtrl.

Proposition 1. *(Phung et al., 2021) Let $\{\mathcal{D}_s^k\}_{k=1}^K$ be a set of source domains and a mixture of source domains $\mathcal{D}^{\boldsymbol{\omega}} = \sum_{k=1}^K \boldsymbol{\omega}_{s_k}\mathcal{D}_s^k$. The target risk ϵ_t is upper bounded:*

$$\epsilon_t \le \sum_{k=1}^{K}\boldsymbol{\omega}_{s_k}\epsilon_{s_k} + \gamma + \rho + \lambda_{\mathcal{H}}. \tag{13}$$

$\lambda_{\mathcal{H}}$ is the ideal joint risk which is a natural characteristic and can be considered as a constant. $\boldsymbol{\omega}_{s_k}$ is the $(K\text{-}1)$-simplex representing the number ratio of each source domain. $\sum_{k=1}^K \boldsymbol{\omega}_{s_k}\epsilon_{s_k}$ is the source risk.

The second term $\gamma := \sum_{i=1}^K\sum_{j=1}^K Q\frac{\sqrt{\boldsymbol{\omega}_{s_j}}}{K}d(P_F^T, P_F^{S_i})$ can be regarded as the representation discrepancy between multiple source domains and the target domain, where Q is a positive constant. However, this term is impossible to be tackled directly without knowing the target domain. To this end, we have:

$$\gamma \le \sum_{i=1}^{K}\sum_{j=1}^{K} Q\frac{\sqrt{\boldsymbol{\omega}_{s_j}}}{K}(d(P_F^T, P_F^{Aug})+d(P_F^{Aug}, P_F^{S_i})). \tag{14}$$

The extrapolation relies on the feature extractor finding invariant representations between the source domains and the augmented domain. Note that although the augmented instance $\widetilde{\mathbf{x}}$ has the same label as $\mathbf{x}$, we cannot assume it can be correctly classified due to it being randomly transformed.

The goal of the mini-max operation on the classifiers discrepancy is to reduce the $d(P_F^{Aug}, P_F^{S_i})$. Similar to [22], we take advantage of the direct estimation of the symmetric difference divergence to relax the bound [2] of a known domain outside the source support:

$$\min_{F} \big(\max_{G^1, G^2} \mathbb{E}_{\mathbf{x}\sim\mathcal{X}}[G^1 \circ F(\widetilde{\mathbf{x}}) \neq G^2 \circ F(\widetilde{\mathbf{x}})] \big). \tag{15}$$

which is similar to Eq.(10) and Eq.(12).

And the third term $\rho := \sum_{i=1}^{K}\sum_{j=1}^{K} Q\frac{\sqrt{\omega_{s_j}}}{K} d(P_F^{s_i}, P_F^{s_j})$ is the feature representation discrepancy between every two subdomains of the same class in the latent space. This term is proportional to the average distance between elements in the convex hull of the aligned distributions. We reduce this term via minimizing the $\mathcal{L}_{\text{ms}}$ loss:

Theorem 2 *Let N be the batch size, the adjustable constant $C_w = \eta(2N-2)$. Assuming the number of instances per subdomain is similar. As $\eta, N \to \infty$, with respect to arbitrary anchor $\mathbf{x}_i \in \mathcal{X}$, the expectation of ℓ_{ms} converges to:*

$$\begin{aligned}\lim_{\eta, N\to\infty} \mathbb{E}\big[\ell_{ms}\big] = \log \mathbb{E}_{(i,l^+)\in S_{neg+}}\big[\exp\big(sim(\mathbf{z}_i, \mathbf{z}_{l^+})\big)\big] \\ -\frac{1}{\tau}\mathbb{E}_{(i,j)\in S_{pos\text{-}}}\big[sim(\mathbf{z}_i, \mathbf{z}_j)\big] + \log C_w.\end{aligned} \tag{16}$$

Thm.2 intuitively present the advantages of $\mathcal{L}_{\text{ms}}$. The first term is only related to $S_{\text{neg+}}$, suggesting that increasing C_{w} tends to ignore domain diversity to locate the essential differences between classes, allowing features to be discretized in the latent space. And the second term is only related to $S_{\text{pos-}}$, and the rationale for adopting this strategy has been elucidated in Thm.1.

4 Experimental Evaluation

4.1 Experiment Settings

4.1.1 Datasets

We conduct comprehensive experiments on three public datasets. **PACS** [16] is popular for domain generalization tasks, with very distinct differences in image styles. There are total 9,991 images in 7 classes belonging to 4 domains: Photo(P), Art painting(A), Cartoon(C) and Sketch(S). **Office-Home** [29] consists of 15,588 instances. The task of Office-Home is object recognition with 65 classes in 4 domains: Art(Ar), Clipart(Cl), Product(Pr) and Real-world(Rw). Ten classes of Office-31 [21] overlapped with Caltech dataset were selected to form a new 2,533-image dataset. **Office-Caltech 10**. It contains 4 domains: Amazon(A), Caltech(C), Webcam(W) and DSLR(D). Objects in different domains differ widely in background, scale and resolution.

4.1.2 Image Augmentation

For instances $\mathbf{x}$ that do not require augmentation, we resize these instances to 224×224. Our image augmentation setting draws on the augmentation of SimCLR [3]. In this basis, to obtain a larger extrapolation space, we use a stronger class of augmentations by improving the transformation probability, enhancing the transformation range, etc. Specifically, a 224×224 crop is taken from the center of each 227×227 resized instance. Afterwards, random left-right flip, color distortion, gaussian blur and solarization are performed.

4.1.3 Configuration

We adopt ImageNet pretrained ResNet[7] for fair comparison with existing literature. The batch size N is set to 64. The model is trained for 100 epochs. The initial learning rate for the feature extractor and task classifier layer are set to 0.0001 and 0.001, respectively. We update all parameters via the Adam optimizer with weight decay of 0.0001 and 0.0005 for the feature extractor and task classifiers, respectively. Following [34], we use a temperature τ of 0.07. Our peculiar hyper-parameter η is set to 1, β is set to 0.5, α are set to 0.8 and 0.4 for ResNet-50 and ResNet-18, respectively.

We follow the leave-one-domain-out evaluation protocol: select one domain from the dataset as the unseen target domain, and set the remaining domains as the source domains. The model is solely on the source domains without exposure to the target domain. Detailed experimental setup in PACS and Office-Home datasets is the same as DDAIG [38]. Test results of the target domain are obtained through the classification layer G^1. All results are reported with average classification accuracy over three trials.

4.1.4 Baselines

In addition to the empirical risk minimization (ERM) [28], we compare MsCtrl with 11 state-of-the-art methods: 1) Domain Adversarial Neural Network (**DANN**) [5], 2) Maximum Classifier Discrepancy (**MCD**) [22], 3) Deep Domain-adversarial Image Generation (**DDAIG**) [38], 4) Feature Stylization and Domain-aware Contrastive Learning (**FeatStyl**) [11], 5) Permuted AdaIN (**pAdaIN**) [18], 6) Style Neophile (**StyleNeo**) [12], 7) Domain-specific Optimized Normalization (**DSON**) [23], 8) Risk Extrapolation (**VREx**) [14], 9) Representation Self-challenging (**RSC**) [10], 10) Diversified Neural Averaging (**DNA**) [4], and 11) Self-supervised Contrastive Regularization (**SelfReg**) [13].

4.2 Results and Discussion

Table 1. Leave-one-domain-out results on the PACS dataset. The best performing is emphasized in bold.

Method	A	C	P	S	Avg.
ResNet-18					
ERM	75.20	78.71	89.70	70.41	78.51
DANN	80.22	75.22	93.23	65.51	78.63
MCD	77.15	81.06	94.49	74.22	81.73
DDAIG	84.20	78.10	95.30	74.70	83.08
DSON	84.67	77.65	95.87	82.23	85.11
RSC	83.43	80.31	95.99	80.85	85.15
pAdaIN	81.74	76.91	96.29	75.13	82.51
SelfReg	87.90	79.40	**96.80**	78.30	85.60
FeatStyl	85.30	81.31	95.63	81.19	85.86
StyleNeo	84.41	79.25	94.93	**83.27**	85.47
MsCtrl	**85.52**$^{\pm.8}$	**81.61**$^{\pm.7}$	95.95$^{\pm.5}$	82.50$^{\pm.5}$	**86.40**
ResNet-50					
ERM	75.63	77.69	92.16	76.97	80.61
DANN	80.08	76.69	94.43	75.06	81.63
MCD	83.01	77.43	95.69	74.19	82.58
DSON	87.04	80.62	95.99	82.90	86.64
RSC	87.89	82.16	**97.92**	83.35	87.83
pAdaIN	85.82	81.06	97.17	77.37	85.36
VREx	86.00	79.10	96.90	77.70	84.90
FeatStyl	88.48	83.83	96.59	82.92	87.96
DNA	89.80	83.40	97.70	82.60	88.38
StyleNeo	**90.35**	84.20	96.73	85.18	89.11
MsCtrl	87.68$^{\pm.6}$	**86.50**$^{\pm.6}$	96.97$^{\pm.3}$	**86.12**$^{\pm.1}$	**89.32**

The results on PACS are shown in Table 1. We compare MsCtrl with existing state-of-the-art methods on the PACS dataset. When the value of η is 1, it can be guaranteed that in arbitrary random sampling batch, $S_{\text{neg+}}$ has a higher weight than $S_{\text{neg-}}$. And the expected weight of $S_{\text{neg+}}$ is 1.7 times the expected weight of $S_{\text{neg-}}$. It can be observed that MsCtrl substantially surpasses the baselines and achieves a new state-of-the-art performance with an average accuracy of 86.40% on ResNet-18 and 89.32% on ResNet-50.

Table 2. Leave-one-domain-out results on the Office-Home dataset with ResNet-18 backbone.

Method	Ar	Cl	Pr	Rw	Avg.
ERM	55.38	50.95	71.68	72.96	62.74
DANN	57.88	48.71	71.16	72.99	62.69
MCD	55.15	53.10	71.62	70.98	62.71
DDAIG	59.20	52.30	74.60	76.00	65.53
DSON	59.37	45.70	71.84	74.68	62.90
RSC	58.42	47.90	71.63	74.54	63.12
FeatStyl	**60.24**	53.54	74.36	**76.66**	66.20
StyleNeo	59.55	**55.01**	73.57	75.52	65.89
MsCtrl	$59.41^{\pm.9}$	$54.98^{\pm.5}$	$\mathbf{74.70}^{\pm.6}$	$76.09^{\pm.2}$	**66.30**

The results on Office-Home are shown in Table 2. We compare MsCtrl against the state-of-the-art methods based on Office-Home with ResNet-18. Following [38], we randomly divide the dataset into 90% and 10% for training and validation, and only use the training split for training. The results show that MsCtrl outperforms the baselines with an average accuracy of 66.30%. According to [37], compared to PACS, the domain distribution shift in the Office-Home dataset is not so significant. Nevertheless, our model still achieves very good performance. This demonstrates that it has advantages over existing methods not only for distant target domains, but also close target domains. Since Office-Caltech 10 is relatively small, we

Table 3. Leave-one-domain-out results on the Office-Caltech 10 dataset with ResNet-18 backbone.

Method	A	C	D	W	Avg.
ERM	94.39	87.12	**100.0**	97.52	94.76
DANN	94.36	86.91	**100.0**	98.41	94.92
MCD	93.98	87.74	99.79	96.39	94.47
RSC	94.40	88.84	**100.0**	98.31	95.39
VREx	94.29	89.82	**100.0**	98.76	95.72
MsCtrl	$\mathbf{95.03}^{\pm.1}$	$\mathbf{90.21}^{\pm.1}$	$\mathbf{100.0}^{\pm.0}$	$\mathbf{99.32}^{\pm.0}$	**96.14**

use it to test the generalization ability of MsCtrl on small datasets. Table 3 presents the domain generalization results with ResNet-18 on Office-Caltech 10. We use the same experimental setup as the PACS dataset with the same backbone. As can be observed from Table 3, our method obtains the best results on all tasks and achieves an average accuracy of 96.14%. This demonstrates that MsCtrl is capable of achieving good generalization performance on small datasets.

4.3 Latent Space Visualization.

To further demonstrate the intuition on how the dynamic domain-weighted contrastive loss $\mathcal{L}_{\text{ms}}$ can obtain good feature representations in the source domains, we use t-SNE tool to visualize the representation of multiple source domain distributions in the latent space. We provide instances from all the domains. In order to highlight the role of the loss function and avoid the effect of preprocessing in the latent space, the experiments shown in the visualization do not involve pretraining on ImageNet.

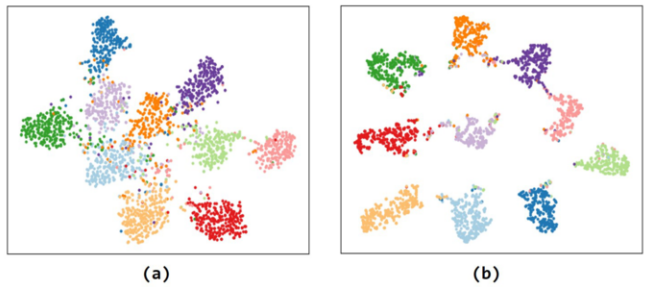

Figure 4. Visualizations of t-SNE on Office-Caltech 10 (color-coded according to class). (a) in case when use cross entropy to train the model. (b) in case when use $\mathcal{L}_{\text{ms}}$ to train the model.

As shown in Fig.4, compared to using cross entropy, employing $\mathcal{L}_{\text{ms}}$ makes the instances of the same class in different domains indistinguishable and the instances of different classes discriminative. It can be observed that the $\mathcal{L}_{\text{ms}}$ term is a good solution to overly-compressed representations of the source domains.

From Fig.5, there is no significant difference between 16 domain-wise similarity matrices. That is, the similarity between representations is only related to the class and not affected by the domain. To be specific, the average similarity difference of different domain-wise matrices at the same position is 0.023, and the maximum similarity difference is 0.077. This is in line with our expectations for domain invariant representations.

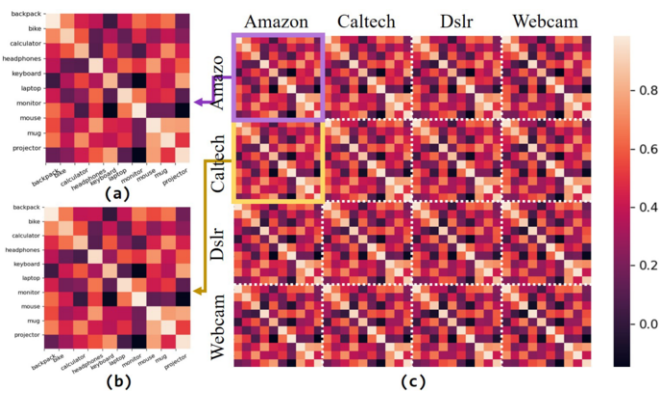

Figure 5. The average cosine similarity matrix across classes. (a) shows same domains Amazon → Amazon domain-wise similarity matrix (darker means less similarity). (b) shows different domains Amazon → Caltech domain-wise similarity matrix. And (c) is the complete matrix contains 4×4 domain-wise similarity matrices.

4.4 Ablation Studies and Futher Analysis

In this section, we study the contribution of each key design component of MsCtrl on its performance. We conduct the ablation studies on PACS with ResNet-18. Except for the item tested, other parameters are fixed to default values.

4.4.1 *Effect of Dynamic Domain-weighted Contrastive Loss*

We study the merits of our dynamic domain-weighted contrastive loss $\mathcal{L}_{\text{ms}}$ to verify the effectiveness by comparing it with the methods that use different positive and negative pairs selection schemes.

A. Positive pairs selection strategy

The results on different selection of positive pairs are shown in Table 4. The performance using $S_{\text{pos-}}$ as positive pairs goes well beyond the ones that using S_{pos} and $S_{\text{pos+}}$.

We further analyze the alignment performance of positive pairs by computing the average distance[2] between representation of positive pairs in S_{pos} and $S_{\text{pos-}}$. As shown in Table 5. Due to the inclusion of negative instances in the same domain as the anchor, $dist(F; S_{\text{pos}})$ has a lower initial average value. However, after model training, the alignment effect using S_{pos} is not obvious, whereas the alignment effect using $S_{\text{pos-}}$ is more than 27 times smaller than the initial value. This result shows that using $S_{\text{pos-}}$ as positive pairs can better align the subdomains of latent space representations than using S_{pos}, which is highly consistent with the conclusion drawn in Theorem 1.

Table 4. Leave-one-domain-out results on different selection of positive pairs on the PACS dataset with ResNet-18 backbone.

$\mathcal{L}_{\text{ms}}$	A	C	P	S	Avg.
Baseline	$75.20^{\pm.4}$	$78.71^{\pm.2}$	$89.70^{\pm.3}$	$70.41^{\pm.2}$	78.51
S_{pos}	$80.83^{\pm2.}$	$79.42^{\pm.5}$	$95.77^{\pm.1}$	$75.70^{\pm.8}$	82.93
$S_{\text{pos+}}$	$80.58^{\pm.7}$	$79.45^{\pm1.}$	$95.93^{\pm.3}$	$77.76^{\pm.8}$	83.43
$S_{\text{pos-}}$	$\mathbf{85.52}^{\pm.8}$	$\mathbf{81.61}^{\pm.7}$	$\mathbf{95.95}^{\pm.5}$	$\mathbf{82.50}^{\pm.5}$	**86.40**

Table 5. Average distance of positive pairs $dist(F; S)$ ($\times 10^{-3}$) on different selection of positive pairs on PACS with ResNet-18.

$dist(F; S)$	A	C	P	S	Avg.(↓)
$dist(F_{\text{inital}}; S_{\text{pos}})$	5.152	5.448	5.041	6.791	5.608
$dist(F_{\text{train}}; S_{\text{pos}})$	4.942	5.363	4.818	6.642	5.441
$dist(F_{\text{inital}}; S_{\text{pos-}})$	5.732	6.086	5.561	7.480	6.215
$dist(F_{\text{train}}; S_{\text{pos-}})$	**0.278**	**0.230**	**0.302**	**0.083**	**0.224**

B. Negative instances weighting strategy

Since $C_{\text{w}} = 2N - 2$, we adjust C_{w} by altering the η value. The results on different η are shown in Table 6. The performance of the model using the contrastive loss without domain-aware is inferior to the performance of the ones with the proposed dynamic domain-weighted contrastive loss. However, as η continues to increase, the performance starts to decrease. The reason for this phenomenon is due to the limited information covering negative instances in a single domain of a limited dataset. In the case of $\eta \to \infty$, negative instances from different domains of the anchor are only introduced into the calculation when $m_2(x) = 0$, which is a very low conditional probability. This greatly reduces the utilization efficiency of datasets with a limited data volume.

4.4.2 *Effect of MsCtrl Architecture*

We study the effect of MsCtrl architecture by comparing our method and alternative methods that do not utilize the augmented data, do not scale beyond the MsCtrl model architecture, and do not utilize either of these. The results are shown in Table 7.

[2] Define $dist(F; S) := -\mathbb{E}_{(i,j)\in S}[\|F(\mathbf{x}_{\text{i}}), F(\mathbf{x}_{\text{j}})\|_2^2]$.

Table 6. Ablation study results with different η on PACS with ResNet-18. $-$ represents the contrastive loss without using dynamic domain-weighted. $\frac{\mathbf{w}_{S_{\text{neg+}}}}{\mathbf{w}_{S_{\text{neg-}}}}$ represents the expected weighted ratio between $S_{\text{neg+}}$ and $S_{\text{neg-}}$.

η	$\frac{\mathbf{w}_{S_{\text{neg+}}}}{\mathbf{w}_{S_{\text{neg-}}}}$	A	C	P	S	Avg.
Baseline	$-$	$75.20^{\pm.4}$	$78.71^{\pm.2}$	$89.70^{\pm.3}$	$70.41^{\pm.2}$	78.51
$-$	1	$83.89^{\pm1.}$	$80.74^{\pm.4}$	$95.61^{\pm.8}$	$80.58^{\pm.8}$	85.20
$\eta = 1$	1.7	$85.52^{\pm.8}$	$81.61^{\pm.7}$	$95.95^{\pm.5}$	$82.50^{\pm.5}$	86.40
$\eta = 2$	6.3	$85.72^{\pm.9}$	$81.74^{\pm.5}$	$96.27^{\pm.5}$	$81.49^{\pm1.}$	86.30
$\eta = 5$	20.3	$85.89^{\pm1.}$	$80.89^{\pm.2}$	$96.07^{\pm.3}$	$81.08^{\pm1.}$	85.98
$\eta = 10$	43.7	$85.90^{\pm.2}$	$80.82^{\pm.8}$	$96.19^{\pm.2}$	$80.99^{\pm.3}$	85.97

On the one hand, when $\widetilde{\mathbf{x}}$ is replaced with $\mathbf{x}$ in all formulae in the model, the average classification accuracy is 82.93%. In this way, we investigate the extrapolation ability of the model when there is no augmented domain to be used as a potential guide.

On the other hand, we trained the model using the same source domain data and augmented data, but without using the MsCtrl adversarial training architecture. Through adversarial training, the randomly varying augmented instances can be better gathered into the original subdomain and guide the extrapolation process. The average accuracy of the model without using the MsCtrl adversarial training architecture is 84.79%. We found that MsCtrl adversarial training architecture can help the model achieve better generalization performance when there is augmented data.

Table 7. Ablation study results on MsCtrl adversarial training architecture on the PACS dataset with ResNet-18 backbone.

$\widetilde{\mathbf{x}}$	$\text{MsCtrl}_{\text{Arch}}$	A	C	P	S	Avg.
–	–	$75.20^{\pm.4}$	$78.71^{\pm.2}$	$89.70^{\pm.3}$	$70.41^{\pm.2}$	78.51
–	✓	$83.07^{\pm.4}$	$79.36^{\pm.3}$	$95.20^{\pm.1}$	$74.08^{\pm3.}$	82.93
✓	–	$83.14^{\pm1.}$	$81.17^{\pm.5}$	$94.89^{\pm.4}$	$79.96^{\pm.5}$	84.79
✓	✓	$85.52^{\pm.6}$	$81.61^{\pm.5}$	$95.95^{\pm.4}$	$82.50^{\pm.4}$	86.40

5 Conclusions

In this paper, we proposed MsCtrl, a novel method for domain generalization. The key idea is to fast minimize the discrepancy between the source domain and the known potential target domain (i.e. the augmented domain) in an adversarial manner, under the "control" of the source risk and the representation discrepancy between the source subdomains remain sufficiently low. To ensure strict source subdomain alignment, we proposed a novel dynamic domain-weighted contrastive loss to remove the domain information noise, which significantly reduces the convex hull diameter of the aligned source subdomains. Extensive experiments demonstrated the effectiveness of MsCtrl on domain generalization tasks as it surpasses state-of-the-art methods in most experimental settings.

Acknowledgements

This work was supported by the National Key Research & Development Program of China (No. 2020YFC2007104), Natural Science Foundation of China (No. 61972383), Beijing Municipal Science & Technology Commission (No. Z221100002722009), the Strategic Priority Research Program of Chinese Academy of Sciences (No. XDA28040500), Youth Innovation Promotion Association CAS (No. 2021101).

References

[1] Pranjal Awasthi, Nishanth Dikkala, and Pritish Kamath, 'Do more negative samples necessarily hurt in contrastive learning?', in *Proceedings of the International Conference on Machine Learning*, pp. 1101–1116. PMLR, (2022).

[2] Shai Ben-David, John Blitzer, Koby Crammer, Alex Kulesza, Fernando Pereira, and Jennifer Wortman Vaughan, 'A theory of learning from different domains', *Machine learning*, **79**(1), 151–175, (2010).

[3] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton, 'A simple framework for contrastive learning of visual representations', in *International conference on machine learning*, pp. 1597–1607. PMLR, (2020).

[4] Xu Chu, Yujie Jin, Wenwu Zhu, Yasha Wang, Xin Wang, Shanghang Zhang, and Hong Mei, 'Dna: Domain generalization with diversified neural averaging', in *International Conference on Machine Learning*, pp. 4010–4034. PMLR, (2022).

[5] Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, Mario Marchand, and Victor Lempitsky, 'Domain-adversarial training of neural networks', *The journal of machine learning research*, **17**(1), 2096–2030, (2016).

[6] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio, 'Generative adversarial nets', *Advances in Neural Information Processing Systems*, **27**, (2014).

[7] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun, 'Deep residual learning for image recognition', in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, (2016).

[8] Judy Hoffman, Mehryar Mohri, and Ningshan Zhang, 'Algorithms and theory for multiple-source adaptation', *Advances in Neural Information Processing Systems*, **31**, (2018).

[9] Jiaxing Huang, Dayan Guan, Aoran Xiao, and Shijian Lu, 'Fsdr: Frequency space domain randomization for domain generalization', in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 6891–6902, (2021).

[10] Zeyi Huang, Haohan Wang, Eric P Xing, and Dong Huang, 'Self-challenging improves cross-domain generalization', in *European Conference on Computer Vision*, pp. 124–140. Springer, (2020).

[11] Seogkyu Jeon, Kibeom Hong, Pilhyeon Lee, Jewook Lee, and Hyeran Byun, 'Feature stylization and domain-aware contrastive learning for domain generalization', in *Proceedings of the ACM International Conference on Multimedia*, pp. 22–31, (2021).

[12] Juwon Kang, Sohyun Lee, Namyup Kim, and Suha Kwak, 'Style neophile: Constantly seeking novel styles for domain generalization', in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 7130–7140, (2022).

[13] Daehee Kim, Youngjun Yoo, Seunghyun Park, Jinkyu Kim, and Jaekoo Lee, 'Selfreg: Self-supervised contrastive regularization for domain generalization', in *Proceedings of the IEEE International Conference on Computer Vision*, pp. 9619–9628, (2021).

[14] David Krueger, Ethan Caballero, Joern-Henrik Jacobsen, Amy Zhang, Jonathan Binas, Dinghuai Zhang, Remi Le Priol, and Aaron Courville, 'Out-of-distribution generalization via risk extrapolation (rex)', in *International Conference on Machine Learning*, pp. 5815–5826. PMLR, (2021).

[15] Bo Li, Yifei Shen, Yezhen Wang, Wenzhen Zhu, Dongsheng Li, Kurt Keutzer, and Han Zhao, 'Invariant information bottleneck for domain generalization', in *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pp. 7399–7407, (2022).

[16] Da Li, Yongxin Yang, Yi-Zhe Song, and Timothy M Hospedales, 'Deeper, broader and artier domain generalization', in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 5542–5550, (2017).

[17] Haoliang Li, Sinno Jialin Pan, Shiqi Wang, and Alex C. Kot, 'Domain generalization with adversarial feature learning', in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 5400–5409, (2018).

[18] Oren Nuriel, Sagie Benaim, and Lior Wolf, 'Permuted adain: Reducing the bias towards global statistics in image classification', in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 9482–9491, (2021).

[19] Trung Phung, Trung Le, Tung-Long Vuong, Toan Tran, Anh Tran, Hung Bui, and Dinh Phung, 'On learning domain-invariant representations for transfer learning with multiple sources', *Advances in Neural Information Processing Systems*, **34**, 27720–27733, (2021).

[20] Joshua David Robinson, Ching-Yao Chuang, Suvrit Sra, and Stefanie Jegelka, 'Contrastive learning with hard negative samples', in *Proceedings of the International Conference on Learning Representations*, (2020).

[21] Kate Saenko, Brian Kulis, Mario Fritz, and Trevor Darrell, 'Adapting visual category models to new domains', in *European conference on computer vision*, pp. 213–226. Springer, (2010).

[22] Kuniaki Saito, Kohei Watanabe, Yoshitaka Ushiku, and Tatsuya Harada, 'Maximum classifier discrepancy for unsupervised domain adaptation', in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 3723–3732, (2018).

[23] Seonguk Seo, Yumin Suh, Dongwan Kim, Geeho Kim, Jongwoo Han, and Bohyung Han, 'Learning to optimize domain specific normalization for domain generalization', in *European Conference on Computer Vision*, pp. 68–83. Springer, (2020).

[24] Yonglong Tian, Chen Sun, Ben Poole, Dilip Krishnan, Cordelia Schmid, and Phillip Isola, 'What makes for good views for contrastive learning?', *Advances in Neural Information Processing Systems*, **33**, 6827–6839, (2020).

[25] Naftali Tishby, Fernando C Pereira, and William Bialek, 'The information bottleneck method', *arXiv preprint physics/0004057*, (2000).

[26] Y-H Tsai, Y Wu, R Salakhutdinov, and L-P Morency, 'Self-supervised learning from a multi-view perspective', in *Proceedings of the International Conference on Learning Representations*, (2021).

[27] Eric Tzeng, Judy Hoffman, Kate Saenko, and Trevor Darrell, 'Adversarial discriminative domain adaptation', in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 7167–7176, (2017).

[28] Vladimir N Vapnik, 'An overview of statistical learning theory', *IEEE transactions on neural networks*, **10**(5), 988–999, (1999).

[29] Hemanth Venkateswara, Jose Eusebio, Shayok Chakraborty, and Sethuraman Panchanathan, 'Deep hashing network for unsupervised domain adaptation', in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 5385–5394, (2017).

[30] Haoqing Wang, Xun Guo, Zhi-Hong Deng, and Yan Lu, 'Rethinking minimal sufficient representation in contrastive learning', in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 16041–16050, (2022).

[31] Jindong Wang, Cuiling Lan, Chang Liu, Yidong Ouyang, Tao Qin, Wang Lu, Yiqiang Chen, Wenjun Zeng, and Philip Yu, 'Generalizing to unseen domains: A survey on domain generalization', *IEEE Transactions on Knowledge and Data Engineering*, 1–1, (2022).

[32] Zengmao Wang, Chaoyang Zhou, Bo Du, and Fengxiang He, 'Self-paced supervision for multi-source domain adaptation', in *Proceedings of the Thirty-First International Joint Conference on Artificial Intelligence*, (2022).

[33] Zeya Wang, Baoyu Jing, Yang Ni, Nanqing Dong, Pengtao Xie, and Eric Xing, 'Adversarial domain adaptation being aware of class relationships', (2020).

[34] Zhirong Wu, Yuanjun Xiong, Stella X. Yu, and Dahua Lin, 'Unsupervised feature learning via non-parametric instance discrimination', in *Proceedings of the IEEE conference on computer vision and pattern recognition*, (June 2018).

[35] Zhirong Wu, Yuanjun Xiong, Stella X Yu, and Dahua Lin, 'Unsupervised ure learning via non-parametric instance discrimination', in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 3733–3742, (2018).

[36] Qinwei Xu, Ruipeng Zhang, Ya Zhang, Yanfeng Wang, and Qi Tian, 'A fourier-based framework for domain generalization', in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 14383–14392, (2021).

[37] Xingxuan Zhang, Linjun Zhou, Renzhe Xu, Peng Cui, Zheyan Shen, and Haoxin Liu, 'Nico++: Towards better benchmarking for domain generalization', *arXiv preprint arXiv:2204.08040*, (2022).

[38] Kaiyang Zhou, Yongxin Yang, Timothy Hospedales, and Tao Xiang, 'Deep domain-adversarial image generation for domain generalisation', in *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pp. 13025–13032, (2020).

[39] Yongchun Zhu, Fuzhen Zhuang, Jindong Wang, Guolin Ke, Jingwu Chen, Jiang Bian, Hui Xiong, and Qing He, 'Deep subdomain adaptation network for image classification', *IEEE transactions on neural networks and learning systems*, **32**(4), 1713–1722, (2020).