Improving Generalization with Domain Convex Game

Fangrui Lv[1] Jian Liang Shuang Li[1,∗] Jinming Zhang[1] Di Liu

[1] Beijing Institute of Technology, China

[1] {fangruilv,shuangli,jinming-zhang}@bit.edu.cn {liangjianzb12,liudi010}@gmail.com

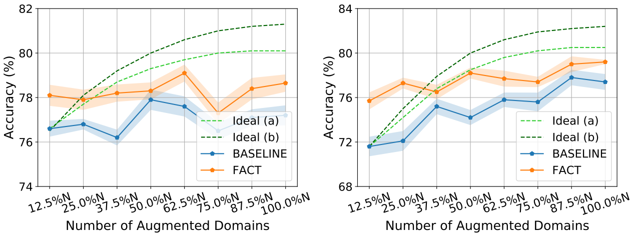
Abstract

Domain generalization (DG) tends to alleviate the poor generalization capability of deep neural networks by learning model with multiple source domains. A classical solution to DG is domain augmentation, the common belief of which is that diversifying source domains will be conducive to the out-of-distribution generalization. However, these claims are understood intuitively, rather than mathematically. Our explorations empirically reveal that the correlation between model generalization and the diversity of domains may be not strictly positive, which limits the effectiveness of domain augmentation. This work therefore aim to guarantee and further enhance the validity of this strand. To this end, we propose a new perspective on DG that recasts it as a convex game between domains. We first encourage each diversified domain to enhance model generalization by elaborately designing a regularization term based on supermodularity. Meanwhile, a sample filter is constructed to eliminate low-quality samples, thereby avoiding the impact of potentially harmful information. Our framework presents a new avenue for the formal analysis of DG, heuristic analysis and extensive experiments demonstrate the rationality and effectiveness. [1]

1. Introduction

Owning extraordinary representation learning ability, deep neural networks (DNNs) have achieved remarkable success on a variety of tasks when the training and test data are drawn from the same distribution [9, 11, 16]. Whereas for out-of-distribution data, DNNs have demonstrated poor generalization capability since the i.i.d. assumption is violated, which is common in real-world conditions [27, 28, 42]. To tackle this issue, domain generalization (DG) has become a propulsion technology, aiming to learn a robust model from multiple source domains so that can generalize well to any unseen target domains with different statistics [2, 19, 22, 30].

Among extensive solutions to improve generalization, domain augmentation [39, 46, 48, 56] has been a classical and prevalent strategy, which focuses on exposing the model with more diverse domains via some augmentation techniques. A common belief is that generalizable models would become easier to learn when the training distributions become more diverse, which has been also emphasized by a recent work [47]. Notwithstanding the promising results shown by this strand of approaches, the claims above are vague and lack of theoretical justification, formal analyses of the relation between domain diversity and model generalization are sparse. Further, the transfer of knowledge may even hurt the performance on target domains in some cases, which is referred to as negative transfer [33, 41]. Thus the relation of domain diversity and model generalization remains unclear. In light of these points, we begin by considering the question: **The stronger the domain diversity, will it certainly help to improve the model generalization capability?**



(a) Cartoon.

(b) Sketch.

Figure 1. The relation between model generalization and domain diversity with Cartoon and Sketch on PACS dataset as the unseen target domain, respectively. N is the maximum number of augmented domains. Note that the *solid lines* denote the actual results of a BASELINE method that combines DeepAll with Fourier augmentation strategy and a SOTA domain augmentation method FACT, while the *dash lines* represent the ideal relation in this work.

To explore this issue, we first quantify domain diversity as the number of augmented domains. Then we conduct a brief experiment using Fourier augmentation strategy [48] as a classical and representative instance. The results presented in Fig 1 show that with the increase of domain diversity, the model generalization (measured by the accuracy on unseen target domain) may not necessarily increase, but sometimes decreases instead, as the solid lines show. On the one hand,

∗ Corresponding author.

[1] Code is available at "https://github.com/BIT-DA/DCG".

this may be because the model does not best utilize the rich information of diversified domains; on the other hand, it may be due to the existence of low-quality samples which contain redundant or noisy information that is unprofitable to generalization [18]. This discovery indicates that there is still room for improvement of the effectiveness of domain augmentation if we enable each domain to be certainly conducive to model generalization as the dash lines in Fig 1.

In this work, we therefore aim to ensure the strictly positive correlation between model generalization and domain diversity to guarantee and further enhance the effectiveness of domain augmentation. To do this, we take inspiration from the literature of convex game that requires each player to bring profit to the coalition [4, 13, 40], which is consistent to our key insight, i.e, make each domain bring benefit to model generalization. Thus, we propose to formalize DG as a convex game between domains. First, we design a novel regularization term based on the supermodularity of convex game. This regularization encourages each diversified domain to contribute to improving model generalization, thus enables the model to better exploit the diverse information. In the meawhile, considering that there may exist samples with unprofitable or even harmful information to generalization, we further construct a sample filter based on the proposed regularization to get rid of the low-quality samples such as noisy or redundant ones, so that their deterioration to model generalization can be avoided. We provide some heuristic analyses and intuitive explanations about the mechanisms behind to demonstrate the rationality in Section 4.

Nevertheless, it is well known that the supermodularity also indicates increasing marginal contribution, which may not hold intuitively in DG, where the marginal contribution of domains is generally decreasing. To mitigate the gap between theory and practice, we impose a constraint on the naive supermodularity when construct our regularization term. We constrain the regularization to work only in case that the supermodularity is violated, i.e., when the marginal contribution of domains decreases. Thus, the limit of our regularization optimization is actually to achieve a *constant marginal contribution*, rather than an impracticable *increasing marginal contribution*. Hence, our regularization can additionally regularize the decreasing speed of the marginal contribution as slow as possible by optimizing towards the *constant marginal contribution*, just like changing the line *Ideal (a)* in Fig 1 into line *Ideal (b)*. Generally, the role of our proposed supermodularity regularization is to encourage the contribution of each domain, and further relieve the *decreasing marginal contribution* of domains to a certain extent, so as to better utilize the diversified information.

Contributions. Our contributions in this work include: (i) Exploring the relation of model generalization and source domain diversity, which reveals the limit of previous domain augmentation strand; (ii) Introducing convex game into DG to guarantee and further enhance the validity of domain augmentation. The proposed framework encourages each domain to conducive to generalization while avoiding the negative impact of low-quality samples, enabling the model to better utilize the information within diversified domains; (iii) Providing heuristic analysis and intuitive explanations about the rationality. The effectiveness and superiority are verified empirically across extensive real-world datasets.

2. Related Work

Domain Generalization researches out-of-distribution generalization with knowledge only extracted from multiple source domains. A promising direction is to diversify training domains so as to improve generalization, referring as to domain augmentation [39, 46, 48, 54, 56]. L2A-OT [54] creates pseudo-novel domains from source data by maximizing an optimal transport-based divergence measure. CrossGrad [39] generates samples from fictitious domains via gradient-based domain perturbation while AdvAug [46] achieves so via adversarially perturbing images. MixStyle [56] and FACT [48] mix style information of different instances to synthetic novel domains. Instead of enriching domain diversity, another popular solution that learning domain-invariant representations by distribution alignment via kernel-based optimization [8, 30], adversarial learning [22, 29], or using uncertainty modeling [24] demonstrate effectiveness for model generalization. Other recent DG works also explore low-rank decomposition [36], self-supervised signals [5], gradient-guided dropout [12], etc. Though our proposed framework builds on the domain augmentation group, we aim to guarantee and further enhance their efficacy beyond via a convex game perspective.

Convex Game is a highly interesting class of cooperative games introduced by [40]. A game is called convex when it satisfies the condition that the profit obtained by the cooperation of two coalitions plus the profit obtained by their intersection will not be less than the sum of profit obtained by the two respectively (a.k.a. supermodularity) [4, 13, 40]. Co-Mixup [15] formulates the optimal construction of mixup augmentation data while encouraging diversity among them by introducing supermodularity. Nevertheless, it is applied to supervised learning which aims to construct salience mixed samples. Recently, [38] rethinks the single-round minmax setting of DG and recasts it as a repeated online game between a player minimizing risk and an adversary presenting test distributions in light of online convex optimization [10]. We note that the definition of convex game exploited in our work follows [40], distinct from that in [10, 38]. To the best of our knowledge, this work is the first to introduce convex game into DG to enhance generalization capability.

Meta Learning [43] is a long-term research exploring to learn how to train a particular model through the training of a meta-model [7, 23, 37], and has drawn increasing attention from DG community [2, 6, 20, 25] recently. The main idea is to simulate domain shift during training by drawing virtual-train/test domains from the original source domains. MLDG [20] originates the episode training paradigm from [7], back-propagating the second-order gradients from an ordinary task loss on random meta-test domains split from the source domains. Subsequent meta learning-based DG methods utilize a similar strategy to meta-learn a regularizer [2], feature-critic network [25], or semantic relationships [6]. Different from the former paradigm that purely leverages the gradient of task objective, which may cause sub-optimal, we utilize the ordinary task losses to construct a supermodularity regularization with more stable optimization, aiming to encourage each training domain to contribute to model generalization.

3. Domain Convex Game

Motivated by such an observation in Section 1, we propose Domain Convex Game (DCG) framework to train models that can best utilize domain diversity, as illustrated in Fig. 2. First, we cast DG as a convex game between domains and design a novel regularization term employing the supermodularity, which encourages each domain to benefit model generalization. Further, we construct a sample filter based on the regularization to exclude bad samples that may cause negative effect on generalization. In this section, we define the problem setup and present the general form of DCG.

3.1. Preliminary

Assuming that there are P source domains of data $\mathcal{D}_s = \cup_{k=1}^{P} D_k$ with n_k labelled samples $\{(\boldsymbol{x}_i^k, y_i^k)\}_{i=1}^{n_k}$ in the k-th domain D_k, where $\boldsymbol{x}_i^k$ and $y_i^k \in \{1, 2, \cdots, C\}$ denote the samples and corresponding labels. DG aims to train a domain-agnostic model $f(\cdot, \boldsymbol{\theta})$ parametrized with $\boldsymbol{\theta}$ on source domains that can generalize well on unseen target domain(s) $\mathcal{D}_t$. As an effective solution for DG, domain augmentation aims to enrich the diversity of source domains generally by synthesizing novel domains via mixing domain-related information, hence boosting model generalization [48, 54, 56]. Our work builds on this strand, and the key insight is to ensure and further improve its efficacy by better leveraging the domain diversity. For concision, in this paper, we adopt a simple Fourier-based augmentation technique [48, 49] to prepare our diversified source domains. Note that the augmentation strategy is substitutable.

Technically, owing to the property that the phase component of Fourier spectrum preserves high-level semantics of the original signal, while the amplitude component contains low-level statistics [32, 35], we augment the source data by distorting the amplitude information while keeping the phase information unchanged. Specifically, we mix the amplitude spectrum of an instance with that of another arbitrary instance by a linear interpolation strategy to synthesize augmented instances from novel domains. We refer readers to [48, 49] for implementation details. Since each augmented sample is generated by mixing domain information of sample pairs from random source domains in a random proportion, it has statistics distinct from the others so that can be regarded as drawn from a novel augmented domain. Thus, we possess another Q augmented source domains of data $\mathcal{D}_s^{aug} = \cup_{k=1}^{Q} D_{P+k}$ with only one sample $\{(\boldsymbol{x}_i^{P+k}, y_i^{P+k})\}_{i=1}^{1}$ in the $(P+k)$-th domain D_{P+k}, where $\boldsymbol{x}_i^{P+k}$ and y_i^{P+k} denote the augmented samples and corresponding labels. Note that the number of augmented domains generated this way is equivalent to the total number of all the original samples since each original sample pair will generate a pair of augmented samples. The goal of DCG is to train a generalizable model $f(\cdot, \boldsymbol{\theta})$ for unseen target domain(s) $\mathcal{D}_t$ with the aid of all $P+Q$ diversified source domains $\mathcal{D}_s \cup \mathcal{D}_s^{aug}$.

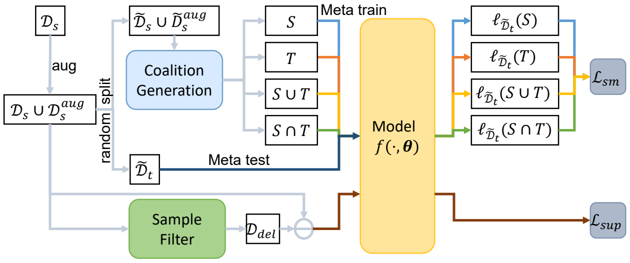

$\mathcal{D}_s$
aug
$\mathcal{D}_s \cup \mathcal{D}_s^{aug}$
random split
$\widetilde{\mathcal{D}}_s \cup \widetilde{\mathcal{D}}_s^{aug}$
Coalition Generation
$\widetilde{\mathcal{D}}_t$
Meta test
Meta train
S
T
$S \cup T$
$S \cap T$
Model $f(\cdot, \boldsymbol{\theta})$
$\ell_{\widetilde{\mathcal{D}}_t}(S)$
$\ell_{\widetilde{\mathcal{D}}_t}(T)$
$\ell_{\widetilde{\mathcal{D}}_t}(S \cup T)$
$\ell_{\widetilde{\mathcal{D}}_t}(S \cap T)$
$\mathcal{L}_{sm}$
Sample Filter
$\mathcal{D}_{del}$
$\mathcal{L}_{sup}$

Figure 2. The pipeline of DCG. We first randomly split the diversified training domains into meta-train and meta-test domains, and generate four coalitions from the former according to the definition of convex game. Then we conduct meta learning on the four coalitions respectively and construct our regularization loss utilizing the meta-test losses of them based on the supermodularity. Meanwhile, we eliminate the low-quality samples by a sample filter and calculate supervision loss on the retained samples.

3.2. Supermodularity Regularization Term

Let $M = \{1, 2, \cdots, m\}$ be a finite set of players and 2^M is the family of $2^{|M|}$ subsets of M. A cooperative game with player set M is a map $v : 2^M \to \mathbb{R}$. For coalition $S \in 2^M$, $v(S)$ is called the worth of S, and is interpreted as the total profit that S can obtain when the players in S cooperate. A game is called convex if it satisfies the *supermodularity property* [4, 13, 40], i.e., for each $S, T \in 2^M$:

$$v(S \cup T) + v(S \cap T) \geq v(S) + v(T). \qquad (1)$$

According to this definition, we can obtain:

$$v(S \cup \{i\} \cup \{j\}) - v(S \cup \{i\}) \geq v(S \cup \{j\}) - v(S), \quad (2)$$

where $S \in 2^M \setminus \{\varnothing\}$ and i, j are two players not in S. We can see that convex game requires each player to contribute to the coalition, which is consistent with our key insight, that is, each training domain is expected to benefit model generalization. More than this, convex game also possesses *increasing marginal contribution property* for players, which may not hold in DG. However, this property does not hinder our goal, but can further alleviate the *decreasing marginal contribution* for domains, as discussed in Section 1.

Thus, we first cast DG as a convex game between domains. To achieve this, at each training iteration, we randomly split the original source data $\mathcal{D}_s$ into $P-V$ meta-train domains of data $\tilde{\mathcal{D}}_s$ and V meta-test domains of data $\tilde{\mathcal{D}}_t$, where $\tilde{\mathcal{D}}_s$ and $\tilde{\mathcal{D}}_t$ share no domain. Then we pick out the augmented domains generated by data in $\tilde{\mathcal{D}}_s$, denoted as $\tilde{\mathcal{D}}_s^{aug}$, and incorporate them into the meta-train domains. This strategy to conduct meta-train/test domains is to mimic the real train-test domain shift in domain augmentation strand, which is discussed in Section 5.4. Then, since one domain may contain multiple samples, we specifically consider involving a specific convex game: *convex fuzzy game* [4] where each player (i.e., each domain) can be partitioned into multiple parts (each part represents a sample in DG). Now we have a finite set of partitioned players $\tilde{M} = \tilde{\mathcal{D}}_s \cup \tilde{\mathcal{D}}_s^{aug}$. We can obtain coalitions $S, T \in 2^{\tilde{M}}$ by randomly sampling two sets of data from meta-train data $\tilde{\mathcal{D}}_s \cup \tilde{\mathcal{D}}_s^{aug}$, respectively. And $S\cup T, S\cap T$ can be naturally constructed by the union and intersection of S and T. As for the profit $v(O), O \in \{S, T, S\cup T, S\cap T\}$, we take the generalization performance evaluated on virtual-test domains $\tilde{\mathcal{D}}_t$ after the meta-training on each coalition O as the value of profit $v(O)$.

Specifically, assuming a loss function $\ell(f(\boldsymbol{x}, \boldsymbol{\theta}), y)$ for a sample between its output and label, e.g., cross-entropy loss for classification task, we first conduct virtual training on the four coalitions $\{S, T, S\cup T, S\cap T\}$, respectively, with the optimization objective:

$$\mathcal{F}(O) := \sum_{x\in O} \ell(f(\boldsymbol{x}, \boldsymbol{\theta}), y)), O \in \{S, T, S\cup T, S\cap T\}. \quad (3)$$

Then the updated virtual parameters $\boldsymbol{\theta}'$ can be computed using one step of gradient descent:

$$\boldsymbol{\theta}' = \boldsymbol{\theta} - \alpha\nabla_{\boldsymbol{\theta}}\mathcal{F}(O), \quad (4)$$

where α is the virtual step size and is empirically set to be the same as the learning rate in our experiments. Thus, we can have the corresponding meta-test loss evaluated on the virtual-test domains $\tilde{\mathcal{D}}_t$ as below:

$$\mathcal{G}(\boldsymbol{\theta}') := \mathbb{E}_{\boldsymbol{x}\in\tilde{\mathcal{D}}_t}\ell(f(\boldsymbol{x}, \boldsymbol{\theta}'), y). \quad (5)$$

This objective simulates test on unseen domains, thus can measure the model generalization obtained by training with one coalition, i.e., $v(O) = -\mathcal{G}(\boldsymbol{\theta}')$. Hence, the supermodularity regularization can be constructed naturally utilizing the meta-test losses of the four coalitions based on Eq. (1):

$$\begin{aligned}\mathcal{L}_{sm} = \max\{0, &\mathcal{G}(\boldsymbol{\theta} - \alpha\nabla_{\boldsymbol{\theta}}\mathcal{F}(S\cup T)) + \mathcal{G}(\boldsymbol{\theta} - \alpha\nabla_{\boldsymbol{\theta}}\mathcal{F}(S\cap T)) \\ &-\mathcal{G}(\boldsymbol{\theta} - \alpha\nabla_{\boldsymbol{\theta}}\mathcal{F}(S)) - \mathcal{G}(\boldsymbol{\theta} - \alpha\nabla_{\boldsymbol{\theta}}\mathcal{F}(T))\}.\end{aligned} \quad (6)$$

Here we exploit a $max(0, \cdot)$ function combined with the pure supermodularity to construct our regularization. In this way, $\mathcal{L}_{sm} > 0$ only when the inequality in Eq. (1) is violated, i.e., the domain marginal contribution is decreasing. Thus, the limit of our regularization optimization corresponds to constant marginal contribution, not the inappropriate increasing marginal contribution. Therefore, this regularization term can not only encourage each training domain to contribute to model generalization, but also alleviate the decrease of marginal contributions to some extent, enabling the model to fully leverage the rich information in diversified domains.

3.3. Sample Filter

Through the optimization of the regularization term, the model will be trained to better utilize the rich information of diversified source domains. However, what we cannot avoid is that there may exist some low-quality samples with harmful information to model generalization. For instance, noisy samples will disturb model to learn generalizable knowledge; while redundant samples may lead to overfitting that hinder the model from learning more diverse patterns.

In this view, we further conduct a sample filter to avoid the negative impact of low-quality samples. Considering that the proposed regularization aims to penalize the decreasing marginal contribution of domains and then better utilize the diverse information, the samples that contribute more to the regularization loss (i.e., cause larger increase) are more unfavorable to our goal, hindering the improvement of model generalization. Thus, we try to measure the contribution of each input to our regularization loss and define the contribution as its score. Inspired by [1] which defines the contribution of each input to the prediction by introducing layer-wise relevance propagation, we formulate the score of each input as the elementwise product between the input and its gradient to regularization loss, i.e., Input × Gradient:

$$score = \boldsymbol{x}^T\nabla_{\boldsymbol{x}}\mathcal{L}_{sm}, x \in \tilde{\mathcal{D}}_s \cup \tilde{\mathcal{D}}_s^{aug}. \quad (7)$$

The higher the score of the sample, the greater the regularization loss will be increased caused by it, and the more it will hinder model from benefiting from diversified domains. Therefore, we pick out the samples with the top-k score, denoted as $\mathcal{D}_{del}$, and cast them away when calculating the supervision loss for diversified source domains to eliminate the negative effect of low quality samples:

$$\mathcal{L}_{sup} = \mathbb{E}_{\boldsymbol{x}\in\mathcal{D}_s\cup\mathcal{D}_s^{aug}\setminus\mathcal{D}_{del}}\ell(f(\boldsymbol{x}, \boldsymbol{\theta}), y). \quad (8)$$

Thus, we optimize the regularization loss to enable model to better utilize the rich information within diversified domains. In the meanwhile, we eliminate the low-quality samples (e.g, noisy samples, redundant samples, etc) by the sample filter to avoid their negative effects. Moreover, it is found that different types of low-quality samples are more likely to be discarded in different training stages, as discussed in Section 4. And we have explored out that low quality sample filtering is necessary for both original and augmented samples in Section 5.4.

The overall optimization objective is:

$$\arg\min_{\boldsymbol{\theta}} \mathcal{L}_{sup} + \omega\mathcal{L}_{sm}, \tag{9}$$

where ω weights the supervision loss and the regularization term. The overall methodological flow is illustrated schematically in Fig. 2 and summarized in Appendix B.

4. Heuristic Analysis

In section 3 we cast DG as a domain convex game and present the detailed formulation of our framework which revolves around the proposed regularization term. Though this term is designed directly according to the supermodularity and has clear objective to achieve our goals, someone may still be curious about the mechanisms behind its effectiveness. So in this section, we provide some heuristic analyses and intuitive explanations to further validate the rationality.

For brevity, we take $S = \{(\boldsymbol{x}_i, y_i)\}$ and $T = \{(\boldsymbol{x}_j, y_j)\}$ as an example, where $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ are from different domains. According to Eq. (6), the optimization goal of our proposed regularization is to make the following inequality hold:

$$\begin{aligned}&\mathcal{G}(\boldsymbol{\theta} - \nabla_{\boldsymbol{\theta}}\ell(f(\boldsymbol{x}_i, \boldsymbol{\theta}), y_i) - \nabla_{\boldsymbol{\theta}}\ell(f(\boldsymbol{x}_j, \boldsymbol{\theta}), y_j)) + \mathcal{G}(\boldsymbol{\theta})\\ &- \mathcal{G}(\boldsymbol{\theta} - \nabla_{\boldsymbol{\theta}}\ell(f(\boldsymbol{x}_i, \boldsymbol{\theta}), y_i)) - \mathcal{G}(\boldsymbol{\theta} - \nabla_{\boldsymbol{\theta}}\ell(f(\boldsymbol{x}_j, \boldsymbol{\theta}), y_j)) \leq 0.\end{aligned} \tag{10}$$

We then carry out the second-order Taylor expansion on the terms in Eq. (10) and obtain:

$$\begin{aligned}&(\nabla_i + \nabla_j)^T H(\nabla_i + \nabla_j) - \nabla_i^T H\nabla_i - \nabla_j^T H\nabla_j\\ &= \nabla_i^T H\nabla_j + \nabla_j^T H\nabla_i \leq 0,\end{aligned} \tag{11}$$

∇_i, ∇_j denote $\nabla_{\boldsymbol{\theta}}\ell(f(\boldsymbol{x}_i, \boldsymbol{\theta}), y_i)$, $\nabla_{\boldsymbol{\theta}}\ell(f(\boldsymbol{x}_j, \boldsymbol{\theta}), y_j)$ respectively, $H = \frac{\partial^2\mathcal{G}(\boldsymbol{\theta})}{\partial\boldsymbol{\theta}\partial\boldsymbol{\theta}^T}$ is the Hessian matrix of $\mathcal{G}(\boldsymbol{\theta})$. We can see that all the zero- and first-order terms of the Taylor-expansion have been dissolved and only the second-order terms are left, which makes the optimization more stable.

Since Hessian matrix H is a real symmetric matrix, for the case where H is positive (negative) definite, we can perform Cholesky decomposition on $H(-H)$ as $L^T L$, where L is an upper triangular matrix with real and positive diagonal elements. Thus, Eq. (11) can be further deduced as follows:

$$\begin{aligned}&\nabla_i^T H\nabla_j + \nabla_j^T H\nabla_i\\ &= \begin{cases}(L\nabla_i)^T(L\nabla_j) + (L\nabla_j)^T(L\nabla_i) \leq 0, & H \succ 0,\\ -((L\nabla_i)^T(L\nabla_j) + (L\nabla_j)^T(L\nabla_i)) \leq 0, & H \prec 0.\end{cases}\end{aligned} \tag{12}$$

Denote $L\nabla_i, L\nabla_j$ as $\tilde{\nabla}_i, \tilde{\nabla}_j$ respectively, which can be regarded as a mapping transformation of the original gradients. Specifically, ∇_i, ∇_j are sample gradients generated in the original "training space" during the meta-training process, while $\tilde{\nabla}_i, \tilde{\nabla}_j$ are sample gradients transformed by matrix L. Since L is derivated from the regularization term calculated on meta-test data that can indicate the model generalization, we can intuitively regard the transformed $\tilde{\nabla}_i, \tilde{\nabla}_j$ as sample gradients mapped to a "generalization space". Therefore, constraining sample gradients in this mapped "generalization space" may generalize better on the real test set compared to constraining the gradients in the original "training space".

Then two main cases can be analysed respectively.

Case 1. *For Hessian matrix $H \prec 0$ (a.k.a. negative definite), Eq. (11) holds when $\tilde{\nabla}_i^T\tilde{\nabla}_j \geq 0$.*

mechanism. When $H \prec 0$, i.e., achieving local maxima, which suggests inferior model generalization, the proposed regularization would help the model improve by enforcing domain consistency on discriminability, that is, pulling the samples from different classes apart and bringing the ones from the same class closer in the "generalization space". As for sample filtering, samples that possess inconsistent gradients, e.g., noisy samples, are more prone to be discarded.

analysis. As Eq. (12) shows, our regularization aims to make the inner product of transformed sample gradients positive when $H \prec 0$, i.e., make the sample gradients consistent in the "generalization space". Assuming samples $\boldsymbol{x}_i$, $\boldsymbol{x}_j$ belong to the same class, then their transformed gradients will be inconsistent when they are apart in the "generalization space", and be consistent when they are close. In contrast, if $\boldsymbol{x}_i$, $\boldsymbol{x}_j$ are from different classes, their transformed gradients would certainly be inconsistent if the samples are close, since they share the same model while possessing different labels. Thus, the optimization of our regularization will draw the samples from the same class closer while pulling the ones from different classes apart to make the gradients consistent, which enforces domain consistency on discriminability. As for sample filtering, the samples that possess very inconsistent gradients are contrary to our goal most and are more likely to obtain larger scores, which are generally noise samples since they are often located at outliers. Therefore, the noise samples are more prone to be discarded in this case.

Case 2. *For Hessian matrix $H \succ 0$ (a.k.a. positive definite), Eq. (11) holds when $\tilde{\nabla}_i^T\tilde{\nabla}_j \leq 0$.*

mechanism. When $H \succ 0$, i.e., achieving local optima, the proposed regularization would help the model jump out by further squeezing out the information within hard samples, that is, detecting the hard samples and then assigning them larger weights implicitly. As for sample filtering, samples that possess very consistent gradients, e.g., redundant samples, are more prone to be discarded.

analysis. As Eq. (12) shows, our regularization aims to make the inner product of the transformed sample gradients negative when $H \succ 0$, i.e., make the gradients inconsistent in the "generalization space". This objective is contrary to our main supervision loss that aims to make all the samples clustered, so it can be regarded as an adversarial optimization. Concretely, the regularization enables the model to generate and detect samples with inconsistent gradients which generally be hard samples since they are often far away from the class center. Then these hard samples would contribute more to the main supervision loss and thus can be considered as being assigned larger weights implicitly during the optimization, just like the mechanism of focal loss [26]. Thus, our regularization can help model jump out of the local optima by squeezing out more information within hard samples, avoiding the model depending on easy patterns or even overfitting on redundant ones. For sample filtering, the samples that produce very consistent gradients, which also means they are redundant ones to a certain, are more likely to be detrimental to our regularization loss and be filtered.

For the general case that H is not fully positive or negative definite, we can take SVD decomposition and regard the model as combined by positive or negative definite submatrices. Then our conclusion holds for each subspace represented by each submatrix.

5. Experiments

5.1. Dataset and Implementation Details

To evaluate our method, we conduct extensive experiments on three popular benchmarks for DG: **PACS** [21] is an object recognition benchmark that covers 9991 images of 7 categories from four different domains, i.e., Art, Cartoon, Photo and Sketch, which with large discrepancy in image styles. **Office-Home** [45] is a commonly-used benchmark including four domains (Art, Clipart, Product, RealWorld). It contains 15,500 images of 65 classes in total. **mini-DomainNet** [55] is a very large-scale domain generalization benchmark consists of about 140k images with 126 classes from four different domains (Clipart, Painting, Real, Sketch). For all benchmarks, we conduct the commonly used leave-one-domain-out experiments [19] and adopt ResNet-18/50 pre-trained on ImageNet [11] as backbone. We train the network using mini-batch SGD with batch size 16, momentum 0.9 and weight decay 5e-4. The initial learning rate is 0.001 and decayed by 0.1 at 80% of the total epochs. For hyper-parameters, we set $\omega = 0.1$ and $k = 5$ for all experiments, which are selected on validation set following standard protocol. All results are reported based on the average accuracy over three independent runs. More details and results with error bars are provided in Appendix.

5.2. Experimental Results

Results on PACS based on ReNet-18 and ResNet-50 are summarized in Table 1. It is clear that DCG achieves the best performance among all the competitors on both backbones. We notice that DCG surpasses the Fourier based augmentation method FACT by a large margin of 1.8% and 1.7% on ResNet-18 and ResNet-50, respectively, which indicate the importance of encouraging each domain to contribute to model generalization. Especially, on the harder target domains Cartoon and Sketch, our method still outperforms the SOTA. There also exist cases where DCG performs relatively poorly, this may due to the task is relatively simple (e.g. *photo*). In general, the comparisons reveal the effectiveness of DCG and further demonstrate that the convex game between domains improves model generalization.

Methods	Art	Cartoon	Photo	Sketch	Avg.
ResNet18					
DeepAll [48]	77.63	76.77	95.85	69.50	79.94
MLDG [20]	78.70	73.30	94.00	65.10	80.70
MASF [6]	80.29	77.17	94.99	71.69	81.04
L2A-OT [54]	83.30	78.20	96.20	73.60	82.80
DDAIG [53]	84.20	78.10	95.30	74.70	83.10
RSC [12]	83.43	80.31	95.99	80.85	85.15
MixStyle [56]	84.10	78.80	96.10	75.90	83.70
FACT [48]	85.37	78.38	95.15	79.15	84.51
DSU [24]	83.60	79.60	95.80	77.60	84.10
STNP [14]	84.41	79.25	94.93	**83.27**	85.47
DCG (*ours*)	**85.94**	**80.76**	**96.41**	82.08	**86.30**
ResNet50					
DeepAll [48]	84.94	76.98	97.64	76.75	84.08
RSC [12]	87.89	82.16	97.92	83.35	87.83
FACT [48]	89.63	81.77	96.75	84.46	88.15
DDG [52]	88.90	85.00	97.20	84.30	88.90
PCL [50]	90.20	83.90	**98.10**	82.60	88.70
STNP [14]	**90.35**	84.20	96.73	85.18	89.11
DCG (*ours*)	90.24	**85.12**	97.76	**86.31**	**89.84**

Table 1. Leave-one-domain-out results on PACS.

Methods	Art	Clipart	Product	Real	Avg.
DeepAll	57.88	52.72	73.50	74.80	64.72
MLDG [20]	52.88	45.72	69.90	72.68	60.30
SagNet [31]	60.20	45.38	70.42	73.38	62.34
RSC [12]	58.42	47.90	71.63	74.54	63.12
DDAIG [53]	59.20	52.30	74.60	76.00	65.50
L2A-OT [54]	60.60	50.10	74.80	**77.00**	65.60
MixStyle [56]	58.70	53.40	74.20	75.90	65.50
FACT [48]	60.34	54.85	74.48	76.55	66.56
DSU [24]	60.20	54.80	74.10	75.10	66.10
STNP [14]	59.55	55.01	73.57	75.52	65.89
DCG (*ours*)	**60.67**	**55.46**	**75.26**	76.82	**67.05**

Table 2. Leave-one-domain-out results on Office-Home.

Results on Office-Home based on ReNet-18 are presented in Table 2, where we beat all the compared baselines in terms of the average accuracy. Due to the similarity to the pre-trained dataset ImageNet, DeepAll acts as a strong base-

Methods	Clipart	Painting	Real	Sketch	Avg.
DeepAll	65.30	58.40	64.70	59.00	61.86
ERM [44]	65.50	57.10	62.30	57.10	60.50
MLDG [20]	65.70	57.00	63.70	58.10	61.12
Mixup [51]	<u>67.10</u>	59.10	64.30	59.20	62.42
MMD [17]	65.00	58.00	63.80	58.40	61.30
SagNet [31]	65.00	58.10	64.20	58.10	61.35
CORAL [34]	66.50	<u>59.50</u>	<u>66.00</u>	<u>59.50</u>	<u>62.87</u>
MTL [3]	65.30	59.00	65.60	58.50	62.10
DCG (*ours*)	**69.38**	**61.79**	**66.34**	**63.21**	**65.18**

Table 3. Leave-one-domain-out results on mini-DomainNet.

Method	Aug.	Reg.	Filter.	Art	Cartoon	Photo	Sketch	Avg.
Baseline	-	-	-	77.6	76.8	95.9	69.5	79.9
Model 1	✓	-	-	83.9	77.0	95.6	77.4	83.4
Model 2	✓	$\mathcal{L}_{maml}$	-	84.7	79.0	95.7	80.1	84.9
Model 3	✓	$\mathcal{L}_{sm}$	-	85.1	80.1	95.9	81.4	85.6
Model 4	✓	-	$\mathcal{F}_{maml}$	84.1	77.7	95.5	78.2	83.9
Model 5	✓	-	$\mathcal{F}_{sm}$	84.4	78.2	95.8	79.3	84.4
Model 6	✓	$\mathcal{L}_{maml}$	$\mathcal{F}_{maml}$	85.3	79.9	96.0	81.5	85.7
DCG	✓	$\mathcal{L}_{sm}$	$\mathcal{F}_{sm}$	**85.9**	**80.8**	**96.4**	**82.1**	**86.3**

Table 4. Ablation study of DCG on PACS dataset.

line on Office-Home. Many previous DG methods, such as MLDG, SagNet, and RSC, can not improve over the simple DeepAll baseline. Nevertheless, our DCG achieves a consistent improvement over DeepAll on all the held-out domains. Moreover, DCG surpasses the latest domain augmentation methods L2A-OT and FACT. The incremental advantages may be due to the relatively smaller domain shift, where the decreasing marginal contribution of domains is more severe.

Results on Mini-DomainNet based on ReNet-18 are shown in Table 3. The much larger number of categories and images makes DomainNet a much more challenging benchmark. DCG still achieves the state-of-the-art performance of 65.18%, surpassing the SOTA by a large margin of 2.31%. It indicates that the waste of diversified information in large datasets is more serious, further validating our efficacy.

5.3. Analysis

Ablation Study. In Table 4, we investigate the role of each component in DCG, including Fourier augmentation (Aug.), supermodularity regularization (Reg. ($\mathcal{L}_{sm}$)) and sample filter (Filter. ($\mathcal{F}_{sm}$)). The Baseline is trained only with the supervision loss of all the original source data. We incorporating our supermodularity regularization $\mathcal{L}_{sm}$ with the Fourier augmentation to obtain Model 3, which greatly surpasses Model 1, demonstrating the significance of encouraging each diversified domain to contribute to generalization. Besides, we aslo apply a regularization $\mathcal{L}_{maml}$ which sums the meta-test losses of all the tasks as MAML [7] to conduct Model 2, its inferiority to Model 3 indicates conducting convex game between domains is more helpful to generalization than simply applying the meta loss. Comparing Model 5 with Model 1, we can observe that the proposed sample filter is also conducive to generalization, suggesting the importance of eliminating nonprofitable information. Finally, DCG performs best in all variants, indicating that the two proposed components complement and benefit each other.

Generalization with Domain Diversity. Figure 4a, 4b show the model generalization with the increase of domain diversity. We use the classification accuracy on the held-out target domain as the metric of model generalization across domains, and the number of augmented domains to measure the domain diversity. It is clear that on both Cartoon and Sketch tasks, the model generalization capability of the baseline methods do not necessarily improve with the increase of domain diversity, but sometimes decrease instead. While in our DCG, the model generalization increases monotonically with the domain diversity on the whole and the decrease of marginal contribution of domains is alleviated. Meanwhile, in a few cases, the generalization of DCG drops a little when domain diversity increases. This is reasonable since the additional augmented domains may be low-quality or harmful to generalization. The results demonstrate that our framework indeed encourages each diversified domain to contribute to model generalization, hence guarantee and further improve the performance of domain augmentation methods.

Visualization of Filtered Samples. To visually verify that our sample filter can effectively eliminate low-quality samples, we provide the samples that obtain the top-k / bottom-k score the most times in the whole training process in Figure 3. We can see that the discarded original samples with top-k score in the first row either be noisy images that have messy background and fuzzy objects, or be images containing naive or classical information which may be redundant. While the high-quality original images in the bottom row are all vivid and rich in information. As for the augmented samples, the discarded ones are almost distinguishable while the retained high-quality ones are limpid. These comparisons demonstrate the effectiveness of our sample filter.

Sensitivity of Hyper-parameters. Figure 4c, 4d show the sensitivity of DCG to hyper-parameters ω and k. Specifically, the value of ω varies from $\{0.01, 0.05, 0.1, 0.5, 1, 0\}$, while k changes from $\{1, 3, 5, 7, 9\}$. It can be observed that DCG achieves competitive performances robustly under a wide range of hyper-parameter values, i.e., $0.05 \leq \omega \leq 0.3$ and $3 \leq k \leq 7$, in either task Cartoon or Sketch, which further verifies the stability of our method.

5.4. Discussion

How to conduct the meta-train and meta-test domains? In DCG, we considers all the diversified domains $\mathcal{D}_s \cup \mathcal{D}_s^{aug}$ into training. We first randomly split the original source domains $\mathcal{D}_s$ into meta-train and meta-test domains, next pick

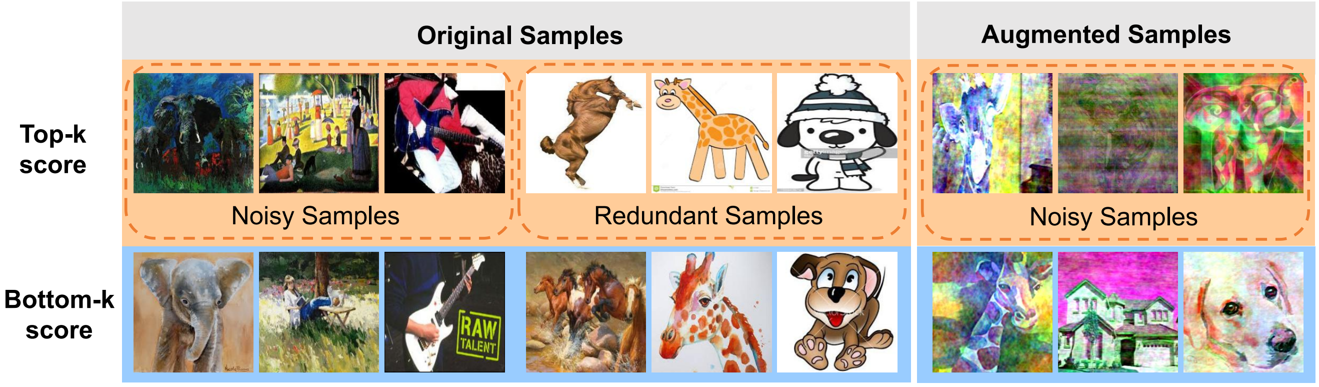
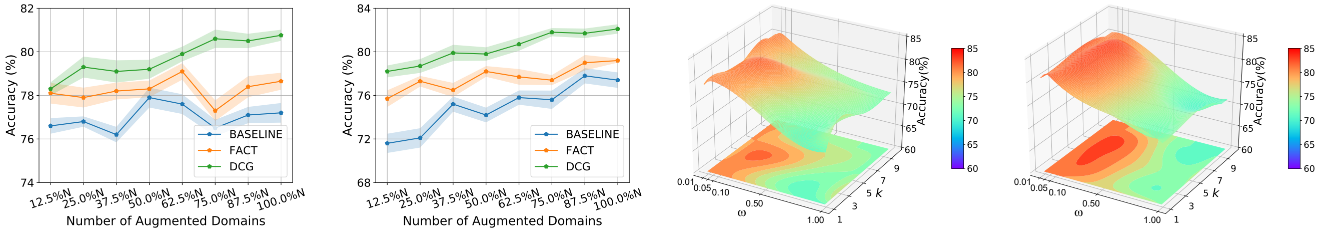


Figure 3. The visualization of samples with top-k and bottom-k score respectively with Cartoon as the unseen target domain.



(a) Cartoon.

(b) Sketch.

(c) Cartoon.

(d) Sketch.

Figure 4. (a)(b): relation between model generalization and domain diversity; (c)(d): sensitivity to hyper-parameters ω and k; with Cartoon and Sketch on PACS dataset as the unseen target domain.

Methods	Art	Cartoon	Photo	Sketch	Avg.
Random_meta_split	85.6	80.2	96.0	81.8	85.9
Filter_only_on_aug	85.4	80.6	96.7	81.8	86.1
Filter_only_on_ori	85.2	80.0	96.5	82.3	86.0
DCG	**85.9**	**80.8**	**96.4**	**82.1**	**86.3**

Table 5. Leave-one-domain-out results on PACS.

out the domains in $\mathcal{D}_s^{aug}$ that are augmented by the current meta-train domains and then merge them into together. Thus, there is no domain augmented by the meta-test domains in the meta-train domains, and vice versa. However, why don't we also randomly split $\mathcal{D}_s^{aug}$ into two parts, since each diversified domain can be regarded as a novel domain? We conduct experiments of this variant and the results in Table 5 shows inferior performance to DCG. This may be because the synthetic novel domains still contain part of the domain-related information of the original ones. In this view, the strategy to conduct meta-train/test domains in Section 3.2 can guarantee the meta-test domains are completely unseen, which better simulates the domain shift between diversified source domains and the held-out unseen target domain.

Is low-quality sample filtering necessary for both original and augmented samples? We conduct experiments that apply the proposed sample filter only on the original samples or augmented samples and the results are shown in Table 5. It can be seen that both variants suffer from a performance drop, which indicates that there exist low-quality samples among both original and augmented samples. Limiting the filtering range will make some low-quality samples be retained to participate in the training process, which may damage the model generalization. Besides, the performance of only filtering the original samples is slightly lower than that of only filtering the augmented ones, which should be due to the augmented samples being less natural.

6. Conclusion & Limitation

This work explores the relation of model generalization and domain diversity, aiming to guarantee and further enhance the efficacy of domain augmentation strand. We then propose a framework to enable each diversified domain contribute to generalization by casting DG as a convex game between domains. Heuristic analysis and comprehensive experiments demonstrate our rationality and effectiveness. Note that we mainly focus on the mixup-based domain augmentation techniques for clarity, while the extension of DCG to other GAN-based techniques needs to be further explored. Besides, it also remains an open problem to design a more efficient strategy to avoid the decrease in training efficiency caused by meta-learning. Nevertheless, we believe our work can inspire the future work of enriching domain diversity with improved generalization capability.

Acknowledgements. This paper was supported by National Key R&D Program of China (No. 2021YFB3301503), and also supported by the National Natural Science Foundation of China under Grant No. 61902028.

References

[1] Sebastian Bach, Alexander Binder, Grégoire Montavon, Frederick Klauschen, Klaus-Robert Müller, and Wojciech Samek. On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation. *PLoS ONE*, 10(7), 2015. 4

[2] Yogesh Balaji, Swami Sankaranarayanan, and Rama Chellappa. Metareg: Towards domain generalization using meta-regularization. In *NeurIPS*, pages 1006–1016, 2018. 1, 3

[3] Gilles Blanchard, Aniket Anand Deshmukh, Ürün Dogan, Gyemin Lee, and Clayton Scott. Domain generalization by marginal transfer learning. *J. Mach. Learn. Res.*, pages 2:1–2:55, 2021. 7

[4] Rodica Brânzei, Dinko Dimitrov, and Stef Tijs. Convex fuzzy games and participation monotonic allocation schemes. *Fuzzy sets and systems*, 139(2):267–281, 2003. 2, 3, 4

[5] Fabio Maria Carlucci, Antonio D'Innocente, Silvia Bucci, Barbara Caputo, and Tatiana Tommasi. Domain generalization by solving jigsaw puzzles. In *CVPR*, pages 2229–2238, 2019. 2

[6] Qi Dou, Daniel Coelho de Castro, Konstantinos Kamnitsas, and Ben Glocker. Domain generalization via model-agnostic learning of semantic features. In *NeurIPS*, pages 6447–6458, 2019. 3, 6

[7] Chelsea Finn, Pieter Abbeel, and Sergey Levine. Model-agnostic meta-learning for fast adaptation of deep networks. In *ICML*, volume 70, pages 1126–1135, 2017. 2, 3, 7

[8] Muhammad Ghifary, David Balduzzi, W. Bastiaan Kleijn, and Mengjie Zhang. Scatter component analysis: A unified framework for domain adaptation and domain generalization. *TPAMI*, 39(7):1414–1430, 2017. 2

[9] Ian J. Goodfellow, Yoshua Bengio, and Aaron C. Courville. *Deep Learning*. Adaptive computation and machine learning. MIT Press, 2016. 1

[10] Elad Hazan. Introduction to online convex optimization. *Found. Trends Optim.*, 2(3-4):157–325, 2016. 2

[11] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, pages 770–778, 2016. 1, 6

[12] Zeyi Huang, Haohan Wang, Eric P. Xing, and Dong Huang. Self-challenging improves cross-domain generalization. In *ECCV*, pages 124–140, 2020. 2, 6

[13] Tatsuro Ichiishi. Super-modularity: applications to convex games and to the greedy algorithm for lp. *Journal of Economic Theory*, 25(2):283–286, 1981. 2, 3

[14] Juwon Kang, Sohyun Lee, Namyup Kim, and Suha Kwak. Style neophile: Constantly seeking novel styles for domain generalization. In *CVPR*, pages 7130–7140, June 2022. 6

[15] Jang-Hyun Kim, Wonho Choo, Hosan Jeong, and Hyun Oh Song. Co-mixup: Saliency guided joint mixup with supermodular diversity. In *ICLR*, 2021. 2

[16] Yann LeCun, Yoshua Bengio, and Geoffrey E. Hinton. Deep learning. *Nat.*, 521(7553):436–444, 2015. 1

[17] Chen-Yu Lee, Tanmay Batra, Mohammad Haris Baig, and Daniel Ulbricht. Sliced wasserstein discrepancy for unsupervised domain adaptation. In *CVPR*, pages 10285–10295, 2019. 7

[18] Kuang-Huei Lee, Xiaodong He, Lei Zhang, and Linjun Yang. Cleannet: Transfer learning for scalable image classifier training with label noise. In *CVPR*, pages 5447–5456, 2018. 2

[19] Da Li, Yongxin Yang, Yi-Zhe Song, and Timothy M. Hospedales. Deeper, broader and artier domain generalization. In *ICCV*, pages 5543–5551, 2017. 1, 6

[20] Da Li, Yongxin Yang, Yi-Zhe Song, and Timothy M. Hospedales. Learning to generalize: Meta-learning for domain generalization. In *AAAI*, pages 3490–3497, 2018. 3, 6, 7

[21] Da Li, Yongxin Yang, Yi-Zhe Song, and Timothy M Hospedales. Deeper, broader and artier domain generalization. In *ICCV*, pages 5542–5550, 2017. 6

[22] Haoliang Li, Sinno Jialin Pan, Shiqi Wang, and Alex C. Kot. Domain generalization with adversarial feature learning. In *CVPR*, pages 5400–5409, 2018. 1, 2

[23] Ke Li and Jitendra Malik. Learning to optimize. In *ICLR*, 2017. 2

[24] Xiaotong Li, Yongxing Dai, Yixiao Ge, Jun Liu, Ying Shan, and Lingyu Duan. Uncertainty modeling for out-of-distribution generalization. In *ICLR*, 2022. 2, 6

[25] Yiying Li, Yongxin Yang, Wei Zhou, and Timothy M. Hospedales. Feature-critic networks for heterogeneous domain generalization. In *ICML*, pages 3915–3924, 2019. 3

[26] Tsung-Yi Lin, Priya Goyal, Ross B. Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. In *ICCV*, pages 2999–3007. IEEE Computer Society, 2017. 6

[27] Mingsheng Long, Han Zhu, Jianmin Wang, and Michael I. Jordan. Unsupervised domain adaptation with residual transfer networks. In *NeurIPS*, pages 136–144, 2016. 1

[28] Xinhong Ma, Tianzhu Zhang, and Changsheng Xu. Deep multi-modality adversarial networks for unsupervised domain adaptation. *IEEE Trans. Multim.*, 21(9):2419–2431, 2019. 1

[29] Saeid Motiian, Marco Piccirilli, Donald A. Adjeroh, and Gianfranco Doretto. Unified deep supervised domain adaptation and generalization. In *ICCV*, pages 5716–5726, 2017. 2

[30] Krikamol Muandet, David Balduzzi, and Bernhard Schölkopf. Domain generalization via invariant feature representation. In *ICML*, pages 10–18, 2013. 1, 2

[31] Hyeonseob Nam, HyunJae Lee, Jongchan Park, Wonjun Yoon, and Donggeun Yoo. Reducing domain gap via style-agnostic networks. *CoRR*, abs/1910.11645, 2019. 6, 7

[32] A. V. Oppenheim and J. S. Lim. The importance of phase in signals. *Proc IEEE*, 69(5):529–541, 1981. 3

[33] Sinno Jialin Pan and Qiang Yang. A survey on transfer learning. *IEEE Trans. Knowl. Data Eng.*, 22(10):1345–1359, 2010. 1

[34] Xingchao Peng, Qinxun Bai, Xide Xia, Zijun Huang, Kate Saenko, and Bo Wang. Moment matching for multi-source domain adaptation. In *ICCV*, pages 1406–1415, 2019. 7

[35] L. N. Piotrowski and F. W. Campbell. A demonstration of the visual importance and flexibility of spatial-frequency amplitude and phase. *Perception*, 11(3):337–46, 1982. 3

[36] Vihari Piratla, Praneeth Netrapalli, and Sunita Sarawagi. Efficient domain generalization via common-specific low-rank decomposition. In *ICML*, pages 7728–7738, 2020. 2

[37] Sachin Ravi and Hugo Larochelle. Optimization as a model for few-shot learning. In *ICLR*, 2017. 2

[38] Elan Rosenfeld, Pradeep Ravikumar, and Andrej Risteski. An online learning approach to interpolation and extrapolation in domain generalization. *CoRR*, abs/2102.13128, 2021. 2

[39] Shiv Shankar, Vihari Piratla, Soumen Chakrabarti, Siddhartha Chaudhuri, Preethi Jyothi, and Sunita Sarawagi. Generalizing across domains via cross-gradient training. In *ICLR*, 2018. 1, 2

[40] Lloyd S Shapley. Cores of convex games. *International journal of game theory*, 1(1):11–26, 1971. 2, 3

[41] Ben Tan, Yu Zhang, Sinno Jialin Pan, and Qiang Yang. Distant domain transfer learning. In *AAAI*, pages 2604–2610, 2017. 1

[42] Rohan Taori, Achal Dave, Vaishaal Shankar, Nicholas Carlini, Benjamin Recht, and Ludwig Schmidt. Measuring robustness to natural distribution shifts in image classification. In *NeurIPS*, 2020. 1

[43] Sebastian Thrun and Lorien Pratt. *Learning to learn*. Springer Science & Business Media, 2012. 2

[44] Vladimir Vapnik. An overview of statistical learning theory. *IEEE Trans. Neural Networks*, 10(5):988–999, 1999. 7

[45] Hemanth Venkateswara, Jose Eusebio, Shayok Chakraborty, and Sethuraman Panchanathan. Deep hashing network for unsupervised domain adaptation. In *CVPR*, pages 5018–5027, 2017. 6

[46] Riccardo Volpi, Hongseok Namkoong, Ozan Sener, John C. Duchi, Vittorio Murino, and Silvio Savarese. Generalizing to unseen domains via adversarial data augmentation. In *NeurIPS*, pages 5339–5349, 2018. 1, 2

[47] Keyulu Xu, Mozhi Zhang, Jingling Li, Simon Shaolei Du, and Stefanie Jegelka. How neural networks extrapolate: From feedforward to graph neural networks. In *ICLR*, 2021. 1

[48] Qinwei Xu, Ruipeng Zhang, Ya Zhang, Yanfeng Wang, and Qi Tian. A fourier-based framework for domain generalization. In *CVPR*, pages 14383–14392, 2021. 1, 2, 3, 6

[49] Yanchao Yang and Stefano Soatto. FDA: fourier domain adaptation for semantic segmentation. In *CVPR*, pages 4084–4094, 2020. 3

[50] Xufeng Yao, Yang Bai, Xinyun Zhang, Yuechen Zhang, Qi Sun, Ran Chen, Ruiyu Li, and Bei Yu. PCL: proxy-based contrastive learning for domain generalization. In *CVPR*, pages 7087–7097, 2022. 6

[51] Hongyi Zhang, Moustapha Cissé, Yann N. Dauphin, and David Lopez-Paz. mixup: Beyond empirical risk minimization. In *ICLR*, 2018. 7

[52] Hanlin Zhang, Yi-Fan Zhang, Weiyang Liu, Adrian Weller, Bernhard Schölkopf, and Eric P. Xing. Towards principled disentanglement for domain generalization. In *CVPR*, pages 8014–8024, 2022. 6

[53] Kaiyang Zhou, Yongxin Yang, Timothy M. Hospedales, and Tao Xiang. Deep domain-adversarial image generation for domain generalisation. In *AAAI*, pages 13025–13032, 2020. 6

[54] Kaiyang Zhou, Yongxin Yang, Timothy M. Hospedales, and Tao Xiang. Learning to generate novel domains for domain generalization. In *ECCV*, pages 561–578, 2020. 2, 3, 6

[55] Kaiyang Zhou, Yongxin Yang, Yu Qiao, and Tao Xiang. Domain adaptive ensemble learning. *IEEE Transactions on Image Processing*, 30:8008–8018, 2021. 6

[56] Kaiyang Zhou, Yongxin Yang, Yu Qiao, and Tao Xiang. Domain generalization with mixstyle. In *ICLR*, 2021. 1, 2, 3, 6