Generalizing to Unseen Domains: A Survey on Domain Generalization

Jindong Wang, Cuiling Lan, *Member, IEEE*, Chang Liu, Yidong Ouyang, Wenjun Zeng, *Fellow, IEEE*, Tao Qin, *Senior Member, IEEE*

***Abstract*—Machine learning systems generally assume that the training and testing distributions are the same. To this end, a key requirement is to develop models that can generalize to unseen distributions. Domain generalization (DG), *i.e.*, out-of-distribution generalization, has attracted increasing interests in recent years. Domain generalization deals with a challenging setting where one or several different but related domain(s) are given, and the goal is to learn a model that can generalize to an *unseen* test domain. Great progress has been made in the area of domain generalization for years. This paper presents the first review of recent advances in this area. First, we provide a formal definition of domain generalization and discuss several related fields. We then thoroughly review the theories related to domain generalization and carefully analyze the theory behind generalization. We categorize recent algorithms into three classes: data manipulation, representation learning, and learning strategy, and present several popular algorithms in detail for each category. Third, we introduce the commonly used datasets and applications. Finally, we summarize existing literature and present some potential research topics for the future.**

***Index Terms*—Domain generalization, Domain adaptation, Transfer learning, Out-of-distribution generalization**

I. Introduction

MACHINE learning (ML) has achieved remarkable success in various areas, such as computer vision, natural language processing, and healthcare. The goal of ML is to design a model that can learn general and predictive knowledge from training data, and then apply the model to new (test) data. Traditional ML models are trained based on the *i.i.d.* assumption that training and testing data are identically and independently distributed. However, this assumption does not always hold in reality. When the probability distributions of training data and testing data are different, the performance of ML models often deteriorates due to domain distribution gaps. Collecting the data of all possible domains to train ML models is expensive and even prohibitively impossible. Therefore, enhancing the *generalization* ability of ML models is important in both industry and academic fields.

There are many generalization-related research topics such as domain adaptation, meta-learning, transfer learning, covariate shift, and so on. In recent years, *Domain generalization (DG)* has received much attention. As shown in Fig. 1, the goal of domain generalization is to learn a model from one or several different but related domains (*i.e.*, diverse training datasets) that will generalize well on *unseen* testing domains. For instance, given a training set consisting of images coming from sketches, cartoon images and paintings, domain generalization requires to train a good machine learning model that has minimum prediction error in classifying images coming from natural images or photos, which are clearly having distinguished distributions from the images in training set. Over the past years, domain generalization has made significant progress in various areas such as computer vision and natural language processing. Despite the progress, there has not been a survey in this area that comprehensively introduces and summarizes its main ideas, learning algorithms and other related problems to provide research insights for the future.

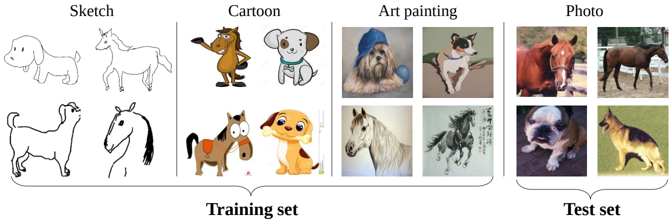

Fig. 1. Examples from the dataset PACS [1] for domain generalization. The training set is composed of images belonging to domains of sketch, cartoon, and art paintings. DG aims to learn a generalized model that performs well on the unseen target domain of photos.

In this paper, we present the first survey on domain generalization to introduce its recent advances, with special focus on its formulations, theories, algorithms, datasets, applications, and future research directions. We hope that this survey can provide a comprehensive review for interested researchers, and inspire more research in this and related areas.

This paper is a heavily extended version of our previously accepted short paper at IJCAI-21 survey track (6 pages, included in the appendix file). Compared to the short paper, this version makes the following extensions. 1) We present the theoretical analysis on domain generalization and the related domain adaptation. 2) We broaden the analysis of methods of different categories by including more related algorithms, comparisons, and discussion. 3) We extend the scope of datasets and applications. We have added more illustrative figures and tables to facilitate the understanding of the big picture and a quick look up.

- We present the theory analysis on domain generalization and the related domain adaptation.
- We substantially extend the methodology by adding new categories: generative modeling for feature disentangle-

J. Wang, C. Lan, C. Liu, W. Zeng, and T. Qin are with Microsoft Research, Beijing, China. Correspondence to: Tao Qin.
E-mail: {jindong.wang,culan,changliu,wezeng,taoqin}@microsoft.com.
Y. Ouyang is with Central University of Finance and Economics, Beijing, China.

ment, invariant risk minimization, gradient operation-based methods, and other learning strategies to comprehensively summarize these DG methods.

- For all the categories, we broaden the analysis of methods of different categories by including more related algorithms, comparisons, and discussion.
- We extend the scope of datasets and applications, and we also explore evaluation standards to domain generalization.

This paper is organized as follows. We formulate the problem of domain generalization and discuss its relationship with existing research areas in Section II. Section III presents the related theories in domain generalization. In Section IV, we describe some representative DG methods in detail. Section V presents the applications and Section VI introduces the benchmark datasets for DG. We summarize the insights from existing work and present some possible future directions in Section VII. Finally, we conclude this paper in Section VIII.

II. Background

A. Formalization of domain generalization

In this section, we introduce the notations and definitions used in this paper.

Definition 1 (Domain). *Let $\mathcal{X}$ denote a nonempty input space and $\mathcal{Y}$ an output space. A domain is composed of data that are sampled from a distribution. We denote it as $\mathcal{S} = \{(\mathbf{x}_i, y_i)\}_{i=1}^n \sim P_{XY}$, where $\mathbf{x} \in \mathcal{X} \subset \mathbb{R}^d$, $y \in \mathcal{Y} \subset \mathbb{R}$ denotes the label, and P_{XY} denotes the joint distribution of the input sample and output label. X, Y denote the corresponding random variables.*

Definition 2 (Domain generalization). *As shown in Fig. 2, in domain generalization, we are given M training (source) domains $\mathcal{S}_{train} = \{\mathcal{S}^i \mid i = 1, \cdots, M\}$ where $\mathcal{S}^i = \{(\mathbf{x}^i_j, y^i_j)\}_{j=1}^{n_i}$ denotes the i-th domain. The joint distributions between each pair of domains are different: $P^i_{XY} \neq P^j_{XY}, 1 \leq i \neq j \leq M$. The goal of domain generalization is to learn a robust and generalizable predictive function $h : \mathcal{X} \rightarrow \mathcal{Y}$ from the M training domains to achieve a minimum prediction error on an* unseen *test domain $\mathcal{S}_{test}$ (i.e., $\mathcal{S}_{test}$ cannot be accessed in training and $P^{test}_{XY} \neq P^i_{XY}$ for $i \in \{1, \cdots, M\}$):*

$$\min_h \mathbb{E}_{(\mathbf{x},y)\in\mathcal{S}_{test}}[\ell(h(\mathbf{x}), y)],$$

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

We list the frequently used notations in TABLE I.

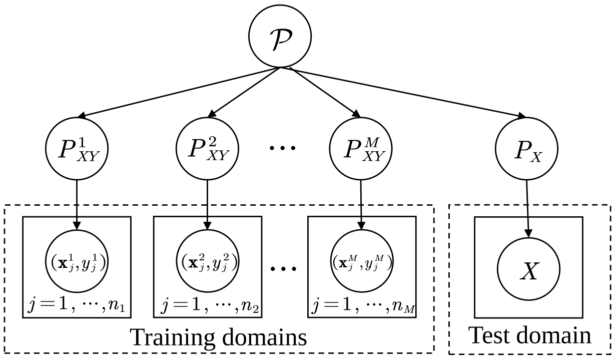

Fig. 2. Illustration of domain generalization.

TABLE I
Notations used in this paper.

Notation	Description	Notation	Description
$\mathbf{x}, y$	Instance/label	$\ell(\cdot, \cdot)$	Loss function
$\mathcal{X}, \mathcal{Y}$	Feature/label space	h	Predictive function
$\mathcal{S}$	Domain	g, f	Feature extractor/classifier
$P(\cdot)$	Distribution	ϵ	Error (risk)
$\mathbb{E}[\cdot]$	Expectation	θ	Model parameter
M	Number of source domains	n_i	Data size of source domain i

B. Related research areas

There are several research fields closely related to domain generalization, including but not limited to: transfer learning, domain adaptation, multi-task learning, multiple domain learning, meta-learning, lifelong learning, and zero-shot learning. We summarize their differences with domain generalization in TABLE II and briefly describe them in the following.

Multi-task learning [2] jointly optimizes models on several related tasks. By sharing representations between these tasks, we could enable the model to generalize better on the original task. Note that multi-task learning does not aim to enhance the generalization to a new (unseen) task. Particularly, multi-domain learning is a kind of multi-task learning, which trains on multiple related domains to learn good models for each original domain [3] instead of new test domains.

Transfer learning [4, 5, 6] trains a model on a source task and aims to enhance the performance of the model on a different but related target domain/task. Pretraining-finetuning is the commonly used strategy for transfer learning where the source and target domains have different tasks and target domain is accessed in training. In DG, the target domain cannot be accessed and the training and test tasks are often the same while they have different distributions.

Domain adaptation (DA) [7, 8] is also popular in recent years. DA aims to maximize the performance on a given target domain using existing training source domain(s). The difference between DA and DG is that DA has access to the target domain data while DG cannot see them during training. This makes DG more challenging than DA but more realistic and favorable in practical applications.

Meta-learning [9, 10, 11] aims to learn the learning algorithm itself by learning from previous experience or tasks, i.e., learning-to-learn. While the learning tasks are different in meta-learning, the learning tasks are the same in domain generalization. Meta-learning is a general learning strategy that can be used for DG [12, 13, 14, 15] by simulating the meta-train and meta-test tasks in training domains to enhance the performance of DG.

Lifelong Learning [16], or continual learning, cares about the learning ability among multiple sequential domains/tasks. It requires the model to continually learn over time by accommodating new knowledge while retaining previously learned experiences. This is also different from DG since it can access the target domain in each time step and it does not explicitly handle the different distributions across domains.

Zero-shot learning [17, 18] aims at learning models from seen classes and classify samples whose categories are unseen in training. In contrast, domain generalization in general studies the problem where training and testing data are from the same classes but with different distributions.

In addition, domain generalization is also related to distributionally robust optimization (DRO) [19], whose goal is to learn a model at worst-case distribution scenario to hope it can generalize well to the test data. DRO focuses on the optimization process that can also be leveraged in domain generalization research. Moreover, DG can also be done via data manipulation or representation learning methods, which are different from DRO methods.

III. Theory

In this section, we review some theories related to domain generalization. Since domain adaptation is closely related to domain generalization, we begin with the theory of domain adaptation.

A. Domain adaptation

For a binary classification problem, we denote the true labeling functions on the source domain as $h^{*s} : \mathcal{X} \to [0,1]$ [1] and that on the target domain as h^{*t}. Let $h : \mathcal{X} \to [0,1]$ be any classifier from a hypothesis space $\mathcal{H}$. The classification error on the source domain between two classifiers h and h' can then be measured by

$$\epsilon^s(h,h') = \mathbb{E}_{\mathbf{x}\sim P_X^s}[h(\mathbf{x}) \neq h'(\mathbf{x})] = \mathbb{E}_{\mathbf{x}\sim P_X^s}[|h(\mathbf{x}) - h'(\mathbf{x})|],$$

and similarly we can define ϵ^t when taking $\mathbf{x} \sim P_X^t$ in the expectation. We define $\epsilon^s(h) := \epsilon^s(h, h^{*s})$ and $\epsilon^t(h) := \epsilon^t(h, h^{*t})$ as the risk of classifier h on the source and target domains, respectively.

The goal of DG/DA is to minimize the target risk $\epsilon^t(h)$, but it is not accessible since we do not have any information on h^{*t}. So people seek to bound the target risk $\epsilon^t(h)$ using the tractable source risk $\epsilon^s(h)$. Ben-David et al. [20] (Thm. 1) give a bound relating the two risks:

$$\begin{aligned}\epsilon^t(h) \leq{}& \epsilon^s(h) + 2d_1(P_X^s, P_X^t) \\ &+ \min_{P_X \in \{P_X^s, P_X^t\}} \mathbb{E}_{\mathbf{x}\sim P_X}[|h^{*s}(\mathbf{x}) - h^{*t}(\mathbf{x})|],\end{aligned} \tag{1}$$

where $d_1(P_X^s, P_X^t) := \sup_{\mathcal{A}\in\mathscr{X}} |P_X^s[\mathcal{A}] - P_X^t[\mathcal{A}]|$ is the *total variation* between the two distributions, and $\mathscr{X}$ denotes the sigma-field on $\mathcal{X}$. The second term on the r.h.s measures the difference of the distributions across the two domains, and the third term represents the difference in the labeling functions.

However, the total variation is a strong distance (i.e., it tends to be very large) that may loosen the bound (1), and is hard to estimate using finite samples. To address this, Ben-David et al. [20] developed another bound ([20], Thm. 2; [21], Thm. 1):

$$\epsilon^t(h) \leq \epsilon^s(h) + d_{\mathcal{H}\Delta\mathcal{H}}(P_X^s, P_X^t) + \lambda_{\mathcal{H}},$$

where the $\mathcal{H}\Delta\mathcal{H}$-divergence is defined as $d_{\mathcal{H}\Delta\mathcal{H}}(P_X^s, P_X^t) := \sup_{h,h'\in\mathcal{H}} |\epsilon^s(h,h') - \epsilon^t(h,h')|$, replacing the total variation d_1 to measure the distribution difference, and the ideal joint risk $\lambda_{\mathcal{H}} := \inf_{h\in\mathcal{H}} [\epsilon^s(h) + \epsilon^t(h)]$ measures the complexity of $\mathcal{H}$ for the prediction tasks on the two domains.

The $\mathcal{H}\Delta\mathcal{H}$-divergence has a better finite-sample guarantee, which leads to a non-asymptotic bound:

Theorem 1 (Domain adaptation error bound (non-asymptotic) [20] (Thm. 2))**.** *Let d be the Vapnik–Chervonenkis (VC) dimension [22] of $\mathcal{H}$, and $\mathcal{U}^s$ and $\mathcal{U}^t$ be unlabeled samples of size n from the two domains. Then for any $h \in \mathcal{H}$ and $\delta \in (0,1)$, the following inequality holds with probability at least $1-\delta$:*

$$\begin{aligned}\epsilon^t(h) \leq{}& \epsilon^s(h) + \hat{d}_{\mathcal{H}\Delta\mathcal{H}}(\mathcal{U}^s, \mathcal{U}^t) + \lambda_{\mathcal{H}} \\ &+ 4\sqrt{\frac{2d\log(2n) + \log(2/\delta)}{n}},\end{aligned}$$

where $\hat{d}_{\mathcal{H}\Delta\mathcal{H}}(\mathcal{U}^s, \mathcal{U}^t)$ is the estimate of $d_{\mathcal{H}\Delta\mathcal{H}}(P_X^s, P_X^t)$ on the two sets of finite data samples.

In the above bounds, the domain distribution difference $d(P_X^s, P_X^t)$ is not controllable, but one may learn a representation function $g : \mathcal{X} \to \mathcal{Z}$ that maps the original input data $\mathbf{x}$ to some representation space $\mathcal{Z}$, so that the representation distributions of the two domains become closer. This direction of DA is thus called DA based on domain-invariant representation (DA-DIR). This intuition is made formal in the following theorem:

Theorem 2 (Domain adaptation error bound on the representation space (non-asymptotic) [23])**.** *Let $g : \mathcal{X} \to \mathcal{Z}$ be a representation function towards some representation space $\mathcal{Z}$, and let $\mathcal{F}$ denote a hypothesis space with VC dimension d of classifiers on top of $\mathcal{Z}$. For unlabeled samples $\mathcal{U}^s$, $\mathcal{U}^t$ of $\mathbf{x}$ of size n from the two domains, denote the samples of representations as $\tilde{\mathcal{U}}^s := \{g(\mathbf{x}) \mid \mathbf{x} \in \mathcal{U}^s\}$ and $\tilde{\mathcal{U}}^t := \{g(\mathbf{x}) \mid \mathbf{x} \in \mathcal{U}^t\}$. Then for any $f \in \mathcal{F}$ and $\delta \in (0,1)$, the following inequality holds with probability at least $1-\delta$:*

$$\begin{aligned}\epsilon^t(f\circ g) \leq{}& \hat{\epsilon}^s(f\circ g) + \hat{d}_{\mathcal{F}}(\tilde{\mathcal{U}}^s, \tilde{\mathcal{U}}^t) + \lambda_{\mathcal{F}} \\ &+ \sqrt{\frac{4}{n}\left(d\log\frac{2n}{d} + d + \log\frac{4}{\delta}\right)},\end{aligned}$$

where $\hat{\epsilon}^s(f\circ g) := \frac{1}{n}\sum_{\mathbf{x}_j\in\mathcal{U}^s} |f(g(\mathbf{x}_j)) - h^{*s}(\mathbf{x}_j)|$ is the empirical source risk on the finite data samples.

Theorem 2 shows the error bound of domain adaptation on the representation space, where we can clearly see that the key to DA-DIR is to learn a proper g to reduce the distribution difference between the source and target domains, i.e. $\hat{d}_{\mathcal{F}}(\tilde{\mathcal{U}}^s, \tilde{\mathcal{U}}^t)$.

The theory of domain-invariant representations has since inspired many DA/DG methods, which can be seen in Section IV-B.

B. Domain generalization

1) Average risk estimation error bound: The first line of domain generalization theory considers the case where the

[1] When the output is in $(0,1)$, it means the probability of $y = 1$.

TABLE II
COMPARISON BETWEEN DOMAIN GENERALIZATION AND SOME RELATED LEARNING PARADIGMS.

Learning paradigm	Training data	Test data	Condition	Test access
Multi-task learning	$\mathcal{S}^1, \cdots, \mathcal{S}^n$	$\mathcal{S}^1, \cdots, \mathcal{S}^n$	$\mathcal{Y}^i \neq \mathcal{Y}^j, 1 \leq i \neq j \leq n$	✓
Transfer learning	$\mathcal{S}^{src}, \mathcal{S}^{tar}$	$\mathcal{S}^{tar}$	$\mathcal{Y}^{src} \neq \mathcal{Y}^{tar}$	✓
Domain adaptation	$\mathcal{S}^{src}, \mathcal{S}^{tar}$	$\mathcal{S}^{tar}$	$P(\mathcal{X}^{src}) \neq P(\mathcal{X}^{tar})$	✓
Meta-learning	$\mathcal{S}^1, \cdots, \mathcal{S}^n$	$\mathcal{S}^{n+1}$	$\mathcal{Y}^i \neq \mathcal{Y}^j, 1 \leq i \neq j \leq n+1$	✓
Lifelong learning	$\mathcal{S}^1, \cdots, \mathcal{S}^n$	$\mathcal{S}^1, \cdots, \mathcal{S}^n$	$\mathcal{S}^i$ arrives sequentially	✓
Zero-shot learning	$\mathcal{S}^1, \cdots, \mathcal{S}^n$	$\mathcal{S}^{n+1}$	$\mathcal{Y}^{n+1} \neq \mathcal{Y}^i, 1 \leq i \leq n$	×
Domain generalization	$\mathcal{S}^1, \cdots, \mathcal{S}^n$	$\mathcal{S}^{n+1}$	$P(\mathcal{S}^i) \neq P(\mathcal{S}^j), 1 \leq i \neq j \leq n+1$	×

target domain is totally unknown (not even unsupervised data), and measures the average risk over all possible target domains. Assume that all possible target distributions follow an underlying hyper-distribution $\mathcal{P}$ on $(\mathbf{x}, y)$ distributions: $P^t_{XY} \sim \mathcal{P}$, and that the source distributions also follow the same hyper-distribution: $P^1_{XY}, \cdots, P^M_{XY} \sim \mathcal{P}$. For generalization to any possible target domain, the classifier to be learned in this case also includes the domain information P_X into its input, so prediction is in the form $y = h(P_X, \mathbf{x})$ on the domain with distribution P_{XY}. For such a classifier h, its average risk over all possible target domains is then given by:

$$\mathcal{E}(h) := \mathbb{E}_{P_{XY} \sim \mathcal{P}} \mathbb{E}_{(\mathbf{x}, y) \sim P_{XY}} [\ell(h(P_X, \mathbf{x}), y)],$$

where ℓ is a loss function on $\mathcal{Y}$. Exactly evaluating the expectations is impossible, but we can estimate it using finite domains/distributions following $\mathcal{P}$, and finite $(\mathbf{x}, y)$ samples following each distribution. As we have assumed $P^1_{XY}, \cdots, P^M_{XY} \sim \mathcal{P}$, the source domains and supervised data could serve for this estimation:

$$\hat{\mathcal{E}}(h) := \frac{1}{M} \sum_{i=1}^{M} \frac{1}{n^i} \sum_{j=1}^{n^i} \ell(h(\mathcal{U}^i, \mathbf{x}^i_j), y^i_j),$$

where we use the supervised dataset $\mathcal{U}^i := \{\mathbf{x}^i_j \mid (\mathbf{x}^i_j, y^i_j) \in \mathcal{S}^i\}$ from domain i as an empirical estimation for P^i_X.

The first problem to consider is how well such an estimate approximates the target $\mathcal{E}(h)$. This can be measured by the largest difference between $\mathcal{E}(h)$ and $\hat{\mathcal{E}}(h)$ on some space of h. To our knowledge, this is first analyzed by Blanchard et al. [24], where the space of h is taken as a reproducing kernel Hilbert space (RKHS). However, different from common treatment, the classifier h here also depends on the distribution P_X, so the kernel defining the RKHS should be in the form $\bar{k}((P^1_X, \mathbf{x}_1), (P^2_X, \mathbf{x}_2))$. Blanchard et al. [24] construct such a kernel using kernels k_X, k'_X on $\mathcal{X}$ and kernel κ on the RKHS $\mathcal{H}_{k'_X}$ of kernel k'_X: $\bar{k}((P^1_X, \mathbf{x}_1), (P^2_X, \mathbf{x}_2)) := \kappa(\Psi_{k'_X}(P^1_X), \Psi_{k'_X}(P^2_X)) k_X(\mathbf{x}_1, \mathbf{x}_2)$, where $\Psi_{k'_X}(P_X) := \mathbb{E}_{\mathbf{x} \sim P_X}[k'_X(\mathbf{x}, \cdot)] \in \mathcal{H}_{k'_X}$ is the kernel embedding of distribution P_X via kernel k'. The result is given in the following theorem, which gives a bound on the largest average risk estimation error within an origin-centered closed ball $\mathcal{B}_{\mathcal{H}_{\bar{k}}}(r)$ of radius r in the RKHS $\mathcal{H}_{\bar{k}}$ of kernel $\bar{k}$, in a slightly simplified case where $n^1 = \cdots = n^M =: n$.

Theorem 3 (Average risk estimation error bound for binary classification [24])**.** *Assume that the loss function ℓ is L_ℓ-Lipschitz in its first argument and is bounded by B_ℓ. Assume also that the kernels k_X, k'_X and κ are bounded by $B^2_k, B^2_{k'} \geq 1$ and B^2_κ, respectively, and the canonical feature map $\Phi_\kappa : v \in \mathcal{H}_{k'_X} \mapsto \kappa(v, \cdot) \in \mathcal{H}_\kappa$ of κ is L_κ-Hölder of order $\alpha \in (0, 1]$ on the closed ball $\mathcal{B}_{\mathcal{H}_{k'_X}}(B_{k'})$* [2]*. Then for any $r > 0$ and $\delta \in (0, 1)$, with probability at least $1 - \delta$, it holds that:*

$$\sup_{h \in \mathcal{B}_{\mathcal{H}_{\bar{k}}}(r)} \left| \hat{\mathcal{E}}(h) - \mathcal{E}(h) \right| \leq C \Bigg(B_\ell \sqrt{-M^{-1} \log \delta} + r B_k L_\ell \Big(B_{k'} L_\kappa \big(n^{-1} \log(M/\delta) \big)^{\alpha/2} + B_\kappa / \sqrt{M} \Big) \Bigg),$$

where C is a constant.

Note that the bound becomes larger in general if (M, n) is replaced with $(1, Mn)$, so it indicates that using domain-wise datasets is better than just pooling them into one mixed dataset, so the domain information plays a role. This result is later extended by Muandet et al. [25], and Deshmukh et al. [26] give a bound for multi-class classification in a similar form.

2) Generalization risk bound: Another line of domain generalization theory consider the risk on a specific target domain, under the assumption of covariate shift (i.e., the labeling function h^* or $P_{Y|X}$ is the same over all domains). This measurement is similar to what is considered in domain adaptation theory in Section III-A, so we adopt the same definition for the source risks $\epsilon^1, \cdots, \epsilon^M$ and the target risk ϵ^t. With the covariate shift assumption, each domain is characterized by the distribution on $\mathcal{X}$. Albuquerque et al. [27] then consider approximating the target domain distribution P^t_X within the convex hull of source domain distributions: $\Lambda := \{\sum_{i=1}^{M} \pi_i P^i_X \mid \pi \in \Delta_M\}$, where Δ_M is the $(M-1)$-dimensional simplex so that each π represents a normalized mixing weights. Similar to the domain adaptation case, distribution difference is measured by the $\mathcal{H}$-divergence to include the influence of the classifier class.

Theorem 4 (Domain generalization error bound [27])**.** *Let $\gamma := \min_{\pi \in \Delta_M} d_{\mathcal{H}}(P^t_X, \sum_{i=1}^{M} \pi_i P^i_X)$ with minimizer π^** [3] *be the distance of P^t_X from the convex hull Λ, and $P^*_X := \sum_{i=1}^{M} \pi^*_i P^i_X$ be the best approximator within Λ. Let $\rho := \sup_{P'_X, P''_X \in \Lambda} d_{\mathcal{H}}(P'_X, P''_X)$ be the diameter of Λ. Then it holds that*

$$\epsilon^t(h) \leq \sum_{i=1}^{M} \pi^*_i \epsilon^i(h) + \frac{\gamma + \rho}{2} + \lambda_{\mathcal{H}, (P^t_X, P^*_X)},$$

[2] This means that for any $u, v \in \mathcal{B}_{\mathcal{H}_{k'_X}}(\mathcal{B}_{\mathcal{H}_{k'}})$, it holds that $\|\Phi_\kappa(u) - \Phi_\kappa(v)\| \leq L_\kappa \|u - v\|^\alpha$, where the norms are of the respective RKHSs.

[3] The original presentation does not mention that the π is the minimizer, but the proof indicates so.

where $\lambda_{\mathcal{H},(P_X^t,P_X^*)}$ is the ideal joint risk across the target domain and the domain with the best approximator distribution P_X^*.

The result can be seen as the generalization of domain adaptation bounds in Section III-A when there are multiple source domains. Again similar to the domain adaptation case, this bound motivates domain generalization methods based on domain invariant representation, which simultaneously minimize the risks over all source domains corresponding to the first term of the bound, as well as the representation distribution differences among source and target domains in the hope to reduce γ and ρ on the representation space.

IV. Methodology

In this section, we introduce existing domain generalization methods in detail. As shown in Fig. 3, we categorize them into three groups, namely:

(1) **Data manipulation:** This category of methods focuses on manipulating the inputs to assist learning general representations. Along this line, there are two kinds of popular techniques: a). *Data augmentation*, which is mainly based on augmentation, randomization, and transformation of input data; b). *Data generation*, which generates diverse samples to help generalization.

(2) **Representation learning:** This category of methods is the most popular in domain generalization. There are two representative techniques: a). *Domain-invariant representation learning*, which performs kernel, adversarial training, explicitly feature alignment between domains, or invariant risk minimization to learn domain-invariant representations; b). *Feature disentanglement*, which tries to disentangle the features into domain-shared or domain-specific parts for better generalization.

(3) **Learning strategy:** This category of methods focuses on exploiting the general learning strategy to promote the generalization capability, which mainly includes three kinds of methods: a). *Ensemble learning*, which relies on the power of ensemble to learn a unified and generalized predictive function; b). *Meta-learning*, which is based on the learning-to-learn mechanism to learn general knowledge by constructing meta-learning tasks to simulate domain shift; c). *Gradient operation*, which tries to learn generalized representations by directly operating on gradients. Additionally, there are other learning strategy that can also be used for DG and we categorized them as other learning strategy.

These three categories of approaches are conceptually different. They are complementary to each other and can be combined towards higher performance. We will describe the approaches for each category in detail hereafter.

A. Data manipulation

We are always hungry for more training data in machine learning (ML). The generalization performance of a ML model often relies on the quantity and diversity of the training data. Given a limited set of training data, data manipulation is one of the cheapest and simplest way to generate samples so as to enhance the generalization capability of the model. The main objective for data manipulation-based DG is to increase the diversity of existing training data using different data manipulation methods. At the same time, the data quantity is also increased.

We formulate the general learning objective of data manipulation-based DG as:

$$\min_h \mathbb{E}_{\mathbf{x},y}[\ell(h(\mathbf{x}),y)] + \mathbb{E}_{\mathbf{x}',y}[\ell(h(\mathbf{x}'),y)],$$

where $\mathbf{x}' = \mathrm{mani}(\mathbf{x})$ denotes the manipulated data using a function $\mathrm{mani}(\cdot)$. Based on the difference on this function, we further categorize existing work into two types: *data augmentation* and *data generation*.

1) Data augmentation-based DG: **Data augmentation.** Augmentation is one of the most useful techniques for training machine learning models. Typical augmentation operations include flipping, rotation, scaling, cropping, adding noise, and so on. They have been widely used in supervised learning to enhance the generalization performance of a model by reducing overfitting [117, 34]. Without exception, they can also be adopted for DG where $\mathrm{mani}(\cdot)$ can be instantiated as these data augmentation functions.

Domain randomization. Other than typical augmentation, domain randomization is an effective technique for data augmentation. It is commonly done by generating new data that can simulate complex environments based on the limited training samples. As data becomes more complex and diverse, generalization ability can be increased. Here, the $\mathrm{mani}(\cdot)$ function is implemented as several manual transformations (commonly used in image data) such as: altering the location and texture of objects, changing the number and shape of objects, modifying the illumination and camera view, and adding different types of random noise to the data.

Tobin et al. [29] first used this method to generate more training data from the simulated environment for generalization in the real environment. Similar techniques were also used in [30, 31, 32, 28] to strengthen the generalization capability of the models. Prakash et al. [33] further took into account the structure of the scene when randomly placing objects for data generation, which enables the neural network to learn to utilize context when detecting objects.

Adversarial data augmentation. Adversarial data augmentation aims to guide the augmentation to optimize the generalization capability, by enhancing the diversity of data while assuring their reliability. Shankar et al. [35] used a Bayesian network to model dependence between label, domain and input instance, and proposed CrossGrad, a cautious data augmentation strategy that perturbs the input along the direction of greatest domain change while changing the class label as little as possible. Volpi et al. [36] proposed an iterative procedure that augments the source dataset with examples from a fictitious target domain that is “hard” under the current model, where adversarial examples are appended at each iteration to enable adaptive data augmentation. Zhou et al. [37] adversarially trained a transformation network for data augmentation instead of directly updating the inputs by gradient ascent.

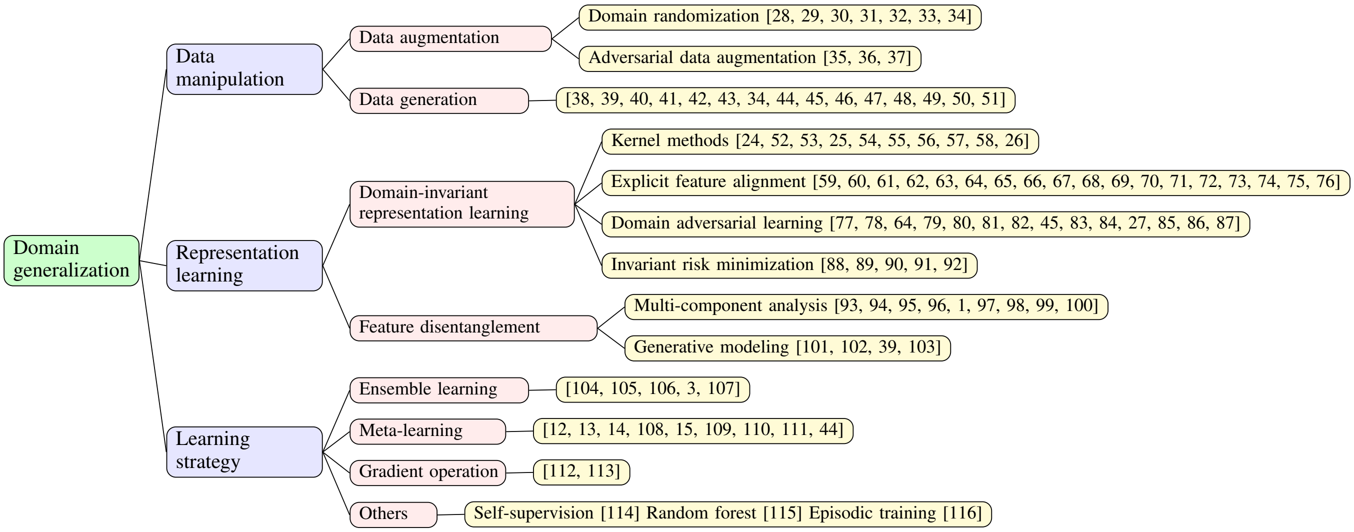
Domain generalization
Data manipulation
Data augmentation
Domain randomization [28, 29, 30, 31, 32, 33, 34]
Adversarial data augmentation [35, 36, 37]
Data generation
[38, 39, 40, 41, 42, 43, 34, 44, 45, 46, 47, 48, 49, 50, 51]
Representation learning
Domain-invariant representation learning
Kernel methods [24, 52, 53, 25, 54, 55, 56, 57, 58, 26]
Explicit feature alignment [59, 60, 61, 62, 63, 64, 65, 66, 67, 68, 69, 70, 71, 72, 73, 74, 75, 76]
Domain adversarial learning [77, 78, 64, 79, 80, 81, 82, 45, 83, 84, 27, 85, 86, 87]
Invariant risk minimization [88, 89, 90, 91, 92]
Feature disentanglement
Multi-component analysis [93, 94, 95, 96, 1, 97, 98, 99, 100]
Generative modeling [101, 102, 39, 103]
Learning strategy
Ensemble learning
[104, 105, 106, 3, 107]
Meta-learning
[12, 13, 14, 108, 15, 109, 110, 111, 44]
Gradient operation
[112, 113]
Others
Self-supervision [114] Random forest [115] Episodic training [116]

Fig. 3. Taxonomy of domain generalization methods.

2) Data generation-based DG: Data generation is also a popular technique to generate diverse and rich data to boost the generalization capability of a model. Here, the function $\mathrm{mani}(\cdot)$ can be implemented using some generative models such as Variational Auto-encoder (VAE) [118], and Generative Adversarial Networks (GAN) [119]. In addition, it can also be implemented using the Mixup [120] strategy.

Rahman et al. [38] used ComboGAN [121] to generate new data and then applied domain discrepancy measure such as MMD [122] to minimize the distribution divergence between real and generated images to help learn general representations. Qiao et al. [39] leveraged adversarial training to create "fictitious" yet "challenging" populations, where a Wasserstein Auto-Encoder (WAE) [123] was used to help generate samples that preserve the semantic and have large domain transportation. Zhou et al. [42] generated novel distributions under semantic consistency and then maximized the difference between source and the novel distributions. Somavarapu et al. [43] introduced a simple transformation based on image stylization to explore cross-source variability for better generalization, where AdaIN [124] was employed to achieve fast stylization to arbitrary styles.

In addition to the above generative models, Mixup [120] is also a popular technique for data generation. Mixup generates new data by performing linear interpolation between any two instances and between their labels with a weight sampled from a Beta distribution, which does not require to train generative models. Recently, there are several methods using Mixup for DG, by either performing Mixup in the original space [47, 48] to generate new samples; or in the feature space [49] which does not explicitly generate raw training samples. These methods achieved promising performance on popular benchmarks while remaining conceptually and computationally simple.

B. Representation learning

Representation learning has always been the focus of machine learning for decades [125] and is also one of the keys to the success of domain generalization. We decompose the prediction function h as $h = f \circ g$, where g is a representation learning function and f is the classifier function. The goal of representation learning can be formulated as:

$$\min_{f,g} \mathbb{E}_{\mathbf{x},y} \ell(f(g(\mathbf{x})), y) + \lambda \ell_{\mathrm{reg}},$$

where ℓ_{reg} denotes some regularization term and λ is the tradeoff parameter. Many methods are designed to better learn the feature extraction function g with corresponding ℓ_{reg}. In this section, we categorize the existing literature on representation learning into two main categories based on different learning principles: *domain-invariant representation learning* and *feature disentanglement*.

1) Domain-invariant representation-based DG: The work of [23] theoretically proved that if the feature representations remain invariant to different domains, the representations are general and transferable to different domains (also refer to Section III). Based on this theory, a plethora of algorithms have been proposed for domain adaptation. Similarly, for domain generalization, the goal is to reduce the representation discrepancy between multiple source domains in a specific feature space to be domain invariant so that the learned model can have a generalizable capability to the unseen domain. Along this line, there are mainly four types of methods: kernel-based methods, domain adversarial learning, explicit feature alignment, and invariant risk minimization.

Kernel-based methods. Kernel-based method is one of the most classical learning paradigms in machine learning. Kernel-based machine learning relies on the kernel function to transform the original data into a high-dimensional feature space without ever computing the coordinates of the data in that space, but by simply computing the inner products between the samples of all pairs in the feature space. One of the most representative kernel-based methods is Support Vector Machine (SVM) [126]. For domain generalization, there are plenty of algorithms based on kernel methods, where the representation learning function g is implemented as some feature map $\phi(\cdot)$ which is easily computed using kernel function $k(\cdot,\cdot)$ such as RBF kernel and Laplacian kernel.

Blanchard et al. [24] first used kernel method for domain generalization and extended it in [52]. They adopted the positive semi-definite kernel learning to learn a domain-invariant kernel from the training data. Grubinger et al. [53] adapted transfer component analysis (TCA) [127] to bridge the multi-domain distance to be closer for DG. Similar to the core idea of TCA, Domain-Invariant Component Analysis (DICA) [25] is one of the classic methods using kernel for DG. The goal of DICA is to find a feature transformation kernel $k(\cdot,\cdot)$ that minimizes the distribution discrepancy between all data in feature space. Gan et al. [54] adopted the similar method as DICA and further added attribute regularization. In contrast to DICA which deals with the marginal distribution, Li et al. [55] learned a feature representation which has domain-invariant class conditional distribution. Ghifary et al. [56] adopted Fisher's discriminant analysis to minimize the discrepancy of representations from the same class and the same domain, and maximize the discrepancy of representations from the different classes and different domains. They proposed Scatter Component Analysis (SCA) to learn domain-invariant representations using kernel-based method, where the SCA kernel takes into account inter- and intra-class discrepancies. Erfani et al. [57] proposed an Elliptical Summary Randomisation (ESRand) that comprises of a randomised kernel and elliptical data summarization. ESRand projected each domain into an ellipse to represent the domain information and then used some similarity metric to compute the distance. Hu et al. [58] proposed multi-domain discriminant analysis to perform class-wise kernel learning for DG, which is more fine-grained. Mahajan et al. [74] proposed to learn disentangled representations using causal matching where they proposed MatchDG to match inputs that are similar under the resultant representation to build an invariant classifier. For domain generalization, some theoretical analysis for kernel-based methods can be found from [24, 26].

Domain adversarial learning. Domain-adversarial training is widely used for learning domain-invariant features. Ganin and Lempitsky [77] and Ganin et al. [78] proposed Domain-adversarial neural network (DANN) for domain adaptation, which adversarially trains the generator and discriminator. The discriminator is trained to distinguish the domains while the generator is trained to fool the discriminator to learn domain invariant feature representations. Li et al. [64] adopted such idea for DG. Gong et al. [79] used adversarial training by gradually reducing the domain discrepancy in a manifold space. Li et al. [80] proposed a conditional invariant adversarial network (CIAN) to learn class-wise adversarial networks for DG. Similar ideas were also used in [81, 85, 86]. Jia et al. [87] used single-side adversarial learning and asymmetric triplet loss to make sure only the real faces from different domains were indistinguishable, but not for the fake ones. In addition to adversarial domain classification, Zhao et al. [82] introduced additional entropy regularization by minimizing the KL divergence between the conditional distributions of different training domains to push the network to learn domain-invariant features. Some other GAN-based methods [45, 84, 27] were also proposed with theoretically guaranteed generalization bound.

Explicit feature alignment. This line of works aligns the features across source domains to learn domain-invariant representations through explicit feature distribution alignment [64, 128, 129], or feature normalization [130, 131, 132, 62].

Motiian et al. [60] introduced a cross-domain contrastive loss for representation learning, where mapped domains are semantically aligned and yet maximally separated. Some methods explicitly minimized the feature distribution divergence by minimizing the maximum mean discrepancy (MMD) [133, 127, 134, 135], second order correlation [136, 137, 138], both mean and variance (moment matching) [129], Wasserstein distance [128], *etc*, of domains for either domain adaptation or domain generalization. Zhou et al. [128] aligned the marginal distribution of different source domains via optimal transport by minimizing the Wasserstein distance to achieve domain-invariant feature space.

Moreover, there are some works that used feature normalization techniques to enhance domain generalization capability [130, 131]. Pan et al. [130] introduced Instance Normalization (IN) layers to CNNs to improve the generalization capability of models. IN has been extensively investigated in the field of image style transfer [139, 140, 124], where the style of an image is reflected by the IN parameters, *i.e.*, mean and variance of each feature channel. Thus, IN layers [141] could be used to eliminate instance-specific style discrepancy to enhance generalization [130]. However, IN is task agnostic and may remove some discriminative information. In IBNNet, IN and Batch Normalization (BN) are utilized in parallel to preserve some discriminative information [130]. In [132], BN layers are replaced by Batch-Instance Normalization (BIN) layers, which adaptively balance BN and IN for each channel by selectively using BN and IN. Jin et al. [62, 63] proposed a Style Normalization and Restitution (SNR) module to simultaneously ensure both high generalization and discrimination capability of the networks. After the style normalization by IN, a restitution step is performed to distill task-relevant discriminative features from the residual (*i.e.*, the difference between the original feature and the style normalized feature) and add them back to the network to ensure high discrimination. The idea of restitution is extended to other alignment-based method to restorate helpful discriminative information dropped by alignment [142].

Invariant risk minimization (IRM). Arjovsky et al. [88] consider another perspective on the domain-invariance of representation for domain generalization. They do not seek to match the representation distribution of all domains, but to enforce the optimal classifier on top of the representation space to be the same across all domains. The intuition is that the ideal representation for prediction is the *cause* of y, and the causal mechanism should not be affected by other factors/mechanisms, thus is domain-invariant.

Formally, the optimization problem of IRM is:

$$\min_{\substack{g\in\mathcal{G},\\ f\in\bigcap_{i=1}^{M}\arg\min_{f'\in\mathcal{F}}\epsilon^i(f'\circ g)}} \sum_{i=1}^{M}\epsilon^i(f\circ g)$$

for some function classes $\mathcal{F}$ of g and $\mathcal{F}$ of f. The constraint for f embodies the desideratum that all domains share the same representation-level classifier, and the objective function encourages f and g to achieve a low source domain risk. However, this problem is hard to solve as it involves an inner-level optimization problem in its constraint. The authors then develop a surrogate problem to learn the feature extractor g that is much more practical:

$$\min_{g\in\mathcal{G}} \sum_{i=1}^{M} \epsilon^i(g) + \lambda \left\| \nabla_f \epsilon^i(f\circ g)\big|_{f=1} \right\|^2,$$

where a dummy representation-level classifier $f = 1$ is considered, and the gradient norm term measures the optimality of this classifier. The work also presents a generalization theory under a perhaps strong linear assumption, that for plenty enough source domains, the ground-truth invariant classifier can be identified.

Due to the new concept and easy implementation, IRM has gain notable visibility recently. There are some further theoretical analyses on the success [143] and failure cases of IRM [92], and IRM has been extended to other tasks including text classification [91] and reinforcement learning [144]. The idea to pursue the invariance of optimal representation-level classifier is also extended. Krueger et al. [145] promote this invariance by minimizing the extrapolated risk among source domains, which essentially minimizes the variance of source-domain risks. Mitrovic et al. [146] aim to learn such a representation in a self-supervised setup, where the second domain is constructed by data augmentation showing various semantic-irrelevant variations.

2) Feature disentanglement-based DG: The goal of disentangled representation learning is to learn a function that maps a sample to a feature vector that contains all the information about different factors of variation, with each dimension (or a subset of dimensions) containing information about only some factor(s). Disentanglement based domain generalization approaches in general decompose a feature representation into understandable compositions/sub-features, with one feature being domain-shared/invariant feature and the other domain-specific feature. The optimization objective of disentanglement-based DG can be summarized as:

$$\min_{g_c,g_s,f} \mathbb{E}_{\mathbf{x},y}\ell(f(g_c(\mathbf{x})),y)+\lambda\ell_{\mathrm{reg}}+\mu\ell_{\mathrm{recon}}([g_c(\mathbf{x}),g_s(\mathbf{x})],\mathbf{x}),$$

where g_c and g_s denote the domain-shared and domain-specific feature representations, respectively. λ, μ are tradeoff parameters. The loss ℓ_{reg} is a regularization term that explicitly encourages the separation of the domain shared and specific features and ℓ_{recon} denotes a reconstruction loss that prevents the loss of information. Note that $[g_c(\mathbf{x}), g_s(\mathbf{x})]$ denotes the combination/integration of two kinds of features (which is not limited to concatenation operation).

Based on the choice of network structures and implementation mechanisms, the disentanglement-based DG methods can mainly be categorized into two types: *multi-component analysis* and *generative modeling*.

Multi-component analysis. In multi-component analysis, the domain-shared and domain-specific features are in general extracted using the domain-shared and domain-specific network parameters. The method of UndoBias [94] started from a SVM model to maximize interval classification on all training data for domain generalization. They represented the parameters of the i-th domain as $\mathbf{w}_i = \mathbf{w}_0 + \Delta_i$, where $\mathbf{w}_0$ denotes the domain-shared parameters and Δ_i denotes the domain-specific parameters. Some other methods extented the idea of UndoBias from different aspects. Niu et al. [95] proposed to use multi-view learning for domain generalization. They proposed Multi-view DG (MVDG) to learn the combination of exemplar SVMs under different views for robust generalization. Ding and Fu [96] designed domain-specific networks for each domain and one shared domain-invariant network for all domains to learn disentangled representations, where low-rank reconstruction is adopted to align two types of networks in structured low-rank fashion. Li et al. [1] extended the idea of UndoBias into the neural network context and developed a low-rank parameterized CNN model for end-to-end training. Zunino et al. [97] learned disentangled representations through manually comparing the attention heat maps for certain areas from different domains. There are also other works that adopt multi-component analysis for disentanglement [98, 93, 99, 100].

Generative modeling. Generative models can be used for disentanglement from the perspective of data generation process. This kind of methods tries to formulate the generative mechanism of the samples from the domain-level, sample-level and label-level. Some works further disentangle the input into class-irrelevant features, which contains the information related to specific instance. The Domain-invariant variational autoencoder (DIVA) [101] disentangled the features into domain information, category information, and other information, which is learned in the VAE framework. Peng et al. [102] disentangled the fine-grained domain information and category information that are learned in VAEs. Qiao et al. [39] also used VAE for disentanglement, where they proposed a Unified Feature Disentanglement Network (UFDN) that treated both data domains and image attributes of interest as latent factors to be disentangled. Liu et al. [103] proposed a unified solution for both domain adaptation and domain generalization based on a causal view of data-generating process, and develop an identification guarantee for the semantic representation and generalization error bounds.

C. Learning strategy

In addition to data manipulation and representation learning, DG was also studied in general machine learning paradigms, which is divided into four categories: *ensemble learning-based DG*, *meta-learning-based DG*, *gradient operation-based DG*, and *others*.

1) Ensemble learning-based DG: Ensemble learning usually combines multiple models, such as classifiers or experts, to enhance the power of models. For domain generalization, ensemble learning exploits the relationship between multiple source domains by using specific network architecture designs and training strategies to improve generalization. They assume that any sample can be regarded as an integrated sample of the multiple source domains, so the overall prediction result can be seen as the superposition of the multiple domain networks.

Mancini et al. [104] proposed to use learnable weights for aggregating the predictions from different source specific classifiers, where a domain predictor is used to predict the probability that a sample belongs to each domain (weights). Segù et al. [61] maintained domain-dependent batch normalization (BN) statistics and BN parameters for different source domains while all the other parameters were shared. In inference, the final prediction was a linear combination of the domain-dependent models with the combination weights inferred by measuring the distances between the instance normalization statistics of the test sample and the accumulated population statistics of each domain. The work of [105] proposed domain-specific layers of different source domains and learning the linear aggregation of these layers to represent a test sample. Zhou et al. [3] proposed Domain Adaptive Ensemble Learning (DAEL), where a DAEL model is composed of a CNN feature extractor shared across domains and multiple domain-specific classifier heads. Each classifier is an expert to its own domain and a non-expert to others. DAEL aims to learn these experts collaboratively, by teaching the non-experts with the expert so as to encourage the ensemble learn how to handle data from unseen domains.

2) Meta-learning-based DG: The key idea of meta-learning is to learn a general model from multiple tasks by either optimization-based methods [147], metric-based learning [148], or model-based methods [149]. The idea of meta-learning has been exploited for domain generalization. They divide the data form multi-source domains into meta-train and meta-test sets to simulate domain shift. Denote θ the model parameters to be learned, meta-learning can be formulated as:

$$\begin{aligned}\theta^* &= \mathrm{Learn}(\mathcal{S}_{mte};\phi^*)\\ &= \mathrm{Learn}(\mathcal{S}_{mte};\mathrm{MetaLearn}(\mathcal{S}_{mtrn})),\end{aligned}$$

where $\phi^* = \mathrm{MetaLearn}(\mathcal{S}_{mtrn})$ denotes the meta-learned parameters from the meta-train set $\mathcal{S}_{mtrn}$ which is then used to learn the model parameters θ^* on the meta-test set $\mathcal{S}_{mte}$. The two functions $\mathrm{Learn}(\cdot)$ and $\mathrm{MetaLearn}(\cdot)$ are to be designed and implemented by different meta-learning algorithms, which corresponds to a bi-level optimization problem. The gradient update can be formulated as:

$$\theta = \theta - \alpha\frac{\partial(\ell(\mathcal{S}_{mte};\theta) + \beta\ell(\mathcal{S}_{mtrn};\phi))}{\partial\theta},$$

where η, β are learning rates for outer and inner loops, respectively.

Finn et al. [147] proposed Model-agnostic meta-learning (MAML). Inspired by MAML, Li et al. [12] proposed MLDG (meta-learning for domain generalization) to use the meta-learning strategy for DG. MLDG splits the data from the source domains into meta-train and meta-test to simulate the domain shift situation to learn general representations. Balaji et al. [13] proposed to learn a meta regularizer (MetaReg) for the classifier. [14] proposed feature-critic training for the feature extractor by designing a meta optimizer. Dou et al. [108] used the similar idea of MLDG and additionally introduced two complementary losses to explicitly regularize the semantic structure of feature space. Du et al. [15] proposed an extended version of information bottleneck named Meta Variational Information Bottleneck (MetaVIB). They regularize the Kullback–Leibler (KL) divergence between distributions of latent encoding of the samples that have the same category from different domains and learn to generate weights by using stochastic neural networks. Recently, some works also adopted meta-learning for semi-supervised DG or discriminative DG [109, 110, 111, 44].

3) Gradient operation-based DG: Other than meta-learning and ensemble learning, several recent works consider using gradient information to force the network learn generalized representations. Huang et al. [112] proposed a self-challenging training algorithm that aims to learn general representations by manipulating gradients. They iteratively discarded the dominant features activated on the training data, and forces the network to activate remaining features that correlate with labels. In this way, network can be forced to learn from more bad cases which will improve generalization ability. Shi et al. [113] proposed a gradient-matching scheme, where their assumption is that the gradient direction of two domains should be the same to enhance common representation learning. To this end, they proposed to maximize the gradient inner product (GIP) to align the gradient direction across domains. With this operation, the network can find weights such that the input-output correspondence is as close as possible across domains. GIP can be formulated as:

$$\mathcal{L} = \mathcal{L}_{\text{cls}}\ (\mathcal{S}_{train};\theta) - \lambda\frac{2}{M(M-1)}\sum_{i,j}^{i\neq j} G_i \cdot G_j,$$

where G_i, G_j are gradient for two domains that can be calulated as $G = \mathbb{E}\frac{\partial\ell(x,y;\theta)}{\partial\theta}$.

4) Other learning strategy: There are some other learning strategy for domain generalization. For instance, self-supervised learning is a recently popular learning paradigm that builds self-supervised tasks from the large-scale unlabeled data [150]. Inspired by this, Carlucci et al. [114] introduced a self-supervision task of solving jigsaw puzzles to learn generalized representations.

Li et al. [116] learned feature extractors and classifiers alternatively. First, they fix the classifier to learn the worst-case feature extractor; then, they fix the feature extractor to learn the worst-case classifier. Through such episodic training, eventually the feature extractor and classifier can learn from these worst cases to perform better for out-of-distribution tasks. Ryu et al. [115] used random forest to improve the generalization ability of convolutional neural networks (CNN). They sampled the triplets based on the probability mass function of the split results given by random forest, which is used for updating the CNN parameters by triplet loss. In the future, there will be more works for DG that uses other learning strategy.

V. Applications

In this section, we discuss the popular tasks/applications for domain generalization.

TABLE III
EIGHT POPULAR IMAGE DATASETS FOR DOMAIN GENERALIZATION.

Dataset	#Domain	#Class	#Sample	Description	Reference
Office-Caltech	4	10	2,533	Caltech, Amazon, Webcam, DSLR	[151]
Office-31	3	31	4,110	Amazon, Webcam, DSLR	[151]
PACS	4	7	9,991	Art, Cartoon, Photos, Sketches	[1]
VLCS	4	5	10,729	Caltech101, LabelMe, SUN09, VOC2007	[152]
Office-Home	4	65	15,588	Art, Clipart, Product, Real	[153]
Terra Incognita	4	10	24,788	Wild animal images taken at locations L100, L38, L43, L46	[154]
Rotated MNIST	6	10	70,000	Digits rotated from $0°$ to $90°$ with an interval of $15°$	[59]
DomainNet	6	345	586,575	Clipart, Infograph, Painting, Quickdraw, Real, Sketch	[155]

High generalization ability is desired in various vision tasks. Many works investigate DG on classification. Some works also study DG for semantic segmentation [79], action recognition [99, 116], face anti-spoofing [81], person Re-ID [47, 142], street view recognition [39], video understanding [95]. Medical analysis is one of the important application areas for DG due to its nature of data scarcity and existence of domain gaps, with the tasks of tissue segmentation [108], Parkinson's disease recognition [25], activity recognition [57], tissue segmentation [108], chest X-ray recognition [74, 72], and EEG-based seizure detection [69].

Apart from those areas, DG is also useful in reinforcement learning of robot control [49, 12] to generalize to an unseen environment. Some work used DG to recognize speech utterance [35, 100], fault diagnosis [72, 68, 67], physics [70], brain-computer interface [66].

In natural language processing, it is also common that training data comes from different domains with different distributions and DG techniques are helpful. Some work used domain generalization for sentiment classification on the Amazon Review dataset [86, 13]. Others used DG for semantic parsing [111], web page classification [45]. Moreover, DG techniques favor broad prospects in some applications, such as financial analysis, stock prediction, weather prediction, and logistics.

VI. DATASETS

In this section, we introduce the widely used benchmark datasets for domain generalization. TABLE III provides an overview of several popular datasets. For image classification, the datasets include Rotated MNIST [59], Office-Caltech [151], Office-31 [151], PACS [1], VLCS [152], Office-Home [153], Terra Incognita [154], and DomainNet [155]. In each dataset, there are usually M domains (refer to TABLE III), where images from all domains share the same categories.

The Graft-versus-host disease (GvHD) dataset [156] is also popular and is used to test several methods [25, 24] for the flow cytometry problem. This dataset was collected from 30 patients with sample sizes ranging from 1,000 to 10,000. This is a time series classification dataset.

Some works [79, 28, 63] applied domain generalization to semantic segmentation, where CityScape [157] and GTA5 [158] datasets were adopted as benchmark datasets. Some works applied domain generalization to object detection, using the datasets of Cityscapes [157], GTA5 [158], Synthia [159] for investigation [63]. Some other works used public datasets or RandPerson [160] for person re-identification [131, 161, 62].

Some works [12, 49] used the OpenAI Gym [162] as the testbed to evaluate the performance of algorithms in reinforcement learning problems such as Cart-Pole and mountain car.

In addition to these widely used datasets, there are also other datasets used in existing literature. The Parkinson's telemonitoring dataset [163] is popular for predicting the clinician's motor and total UPDRS scoring of Parkinson's disease symptoms from voice measures. Some methods [25, 52, 53] used the data from several people as the training domains to learn models that generalize to unseen subjects. Recently, a large dataset called WILDS [164] that aims to benchmark algorithms against distribution shift in the wide has been released.

To test domain generalization algorithms on a test domain, researchers often adopt the "leave-one-domain-out" strategy to choose one domain as the test domain and treat others as the training domains. It is common to split each training domain into training and validation sets, and take all validation sets for model selection and hyperparameter tuning. A full benchmark results of several algorithms on these image classification datasets can be found in [71].

It is worth noting that the datasets of domain generalization have some overlaps with domain adaptation. For instance, Office-31, Office-Caltech, Office-Home and DomainNet are also widely used benchmarks for domain adaptation. Therefore, most domain adaptation datasets can be used for domain generalization benchmark in addition to those we discussed here. For example, Amazon Review dataset [165] is widely used in domain adaptation. It has four different domains on product review (DVDs, Kitchen appliance, Electronics and Books), which can also be used for domain generalization.

VII. DISCUSSION

In this section, we summarize the existing methods and then present several challenges that warrant attention in the future research. Moreover, as a reference, we list all literature in domain generalization in TABLE IV.

A. Summary of existing literature

The quantity and diversity of training data are of great importance to a model's generalization ability. Many methods

TABLE IV
SUMMARY OF EXISTING LITERATURE FOR DOMAIN GENERALIZATION.

Literature	Category	Application
[31]	Augmentation	Car detection
[83]	Augmentation, meta-learning	Image classification
[34]	Augmentation	Image classification
[30]	Augmentation	Reinforcement learning
[33]	Augmentation	Reinforcement learning
CrossGrad [35]	Augmentation, adversarial	Font/Image/speech recognition
[29]	Augmentation	Reinforcement learning
[32]	Augmentation	Image classification
[36]	Augmentation	Image classification
DomainMix [47]	Augmentation	Person Re-ID
Mixup [48]	Augmentation	Image classification
[28]	Augmentation	Image classification, semantic segmentation
MixStyle [49]	Augmentation	Image classification, Person Re-ID, RL
ADAGE [46]	Generation	Image classification
[71]	Generation	Image classification, hand pose recognition
DDG [38]	Generation	Image classification
[43]	Generation	Image classification
UNVP [41]	Generation	Image classification
DeGIA [50]	Generation	Image classification
DDAIG [37]	Generation	Image classification
L2A-OT [42]	Generation	Image classification
M-ADA [39]	Generation, disentangle	Image classification, semantic segmentation
Model-based DG [51]	Generation	Image classification
[24]	Kernel	Flow cytometry
MTL [52]	Kernel	Parkinson's disease, satellite image, flow cytometry
[26]	Kernel	Satellite image
ESRand [57]	Kernel	Activity recognition
KDICA [54]	Kernel	Image classification
SCA [56]	Kernel	Image classification
TCA [53]	Kernel	Landmine detection, Parkinson's disease
MDA [58]	Kernel	Image classification
CIDDG [55]	Kernel	Image classification
DICA [25]	Kernel	Flow Cytometry, Parkinson's disease
[69]	Explicit feature align	EEG seizure
[70]	Explicit feature align	Image classification, physics
MTAE [59]	Explicit feature align	Image classification
[66]	Explicit feature align	Brain computer interface
SNR [62, 63]	Explicit feature align	Person Re-ID, Image classification, semantic segmentation, object detection
FAR [142]	Explicit feature align	Image classification
[72]	Explicit feature align	Fault diagnosis
DSDGN [68]	Explicit feature align	Fault diagnosis
[73]	Explicit feature align	Water object detection
MatchDG [74]	Explicit feature align	Image classification
[65]	Explicit feature align	Image classification
CCSA [60]	Explicit feature align	Image classification
BNE [61]	Explicit feature align	Image classification
DSON [75]	Explicit feature align	Image classification
DFDG [76]	Explicit feature align	Image classification
DDGFD [67]	Explicit feature align	Fault diagnosis
[97]	Explicit feature align	Image classification
[27]	Adversarial	Image classification
DANN [78]	Adversarial	Image classification
DLOW [79]	Adversarial	Semantic segmentation
SSDG [87]	Adversarial	Face anti-spoofing
CIAN [80]	Adversarial	Image classification
MMD-AAE [64]	Adversarial	Image classification, action recognition
CAADG [85]	Adversarial	Image classification
MADDG [81]	Adversarial	Face anti-spoofing
DANNCE [84]	Adversarial	Image classification
[86]	Adversarial	Twitter sentence classification
[82]	Adversarial	Image classification
IRM [88]	Invariant risk minimization	Image classification
Domain2Vec [98]	Disentangle, multi-component	Image classification
[96]	Disentangle, multi-component	Image classification
UndoBias [94]	Disentangle, multi-component	Image classification
[1]	Disentangle, multi-component	Image classification
LRE [99]	Disentangle, multi-component	Image classification, action recognition
MVDG [95]	Disentangle, multi-component	Image/video classification
CSD [100]	Disentangle, multi-component	Image/speech utterance classification
LRE-SVM [93]	Disentangle, multi-component	Image classification
DIVA [101]	Disentangle, generative model	Medical imaging
CSG [103]	Disentangle, generative model	Image classification
UFDN [40]	Disentangle, generative model	Image classification
DAL [102]	Disentangle, generative model	Image classification
D-SAM [105]	Ensemble	Image classification
[106]	Ensemble	Human detection
[104]	Ensemble	Image classification
DAEL [3]	Ensemble	Image classification
[107]	Ensemble	Place categorization
MetaReg [13]	Meta-learning	Image/sentiment classification
DADG [109]	Meta-learning, adversarial	Image classification
MASF [108]	Meta-learning	Image classification, tissue Segmentation
MetaVIB [15]	Meta-learning, feature align.	Image classification
[45]	Meta-learning	Web page classification
MLDG [12]	Meta-learning	Image classification, reinforcement learning
Epi-FCR [116]	Meta-learning	Image classification, action recognition
FC [14]	Meta-learning	Image classification
DGSML [110]	Meta-learning, feature align	Image classification
[111]	Meta-learning	Semantic parsing
M3L [44]	Meta-learning	Person Re-ID
Jigen [114]	Self-supervision	Image classification
RSC [112]	Self-challenging	Image classification
CCFN [115]	Random forest	Image classification

aim to enrich the training data using the aforementioned data manipulation methods to achieve good performance. However, one issue of the data manipulation methods is that there is a lack of theoretical guarantee of the unbound risk of generalization. Therefore, it is important to develop theories for the manipulation-based methods which could further guide the data generation designs without violating ethical standards.

Compared to data manipulation, representation learning has theoretical support in general [23, 24, 27]. Kernel-based methods are widely used in traditional methods while deep learning-based methods play a leading role in recent years. While domain adversarial training often achieves better performance in domain adaptation, in DG, we did not see significant results improvements from these adversarial methods. We think this is probably because the task is relatively easy. For the explicit distribution matching, more and more works tend to match the joint distributions rather than just match the marginal [24, 64] or conditional [55] distributions. Thus, it is more feasible to perform dynamic distribution matching [134, 135]. Both disentanglement and IRM methods have good motivations for generalization, while more efficient training strategy can be developed.

For learning strategy, there is a trend that many works used meta-learning for DG, where it requires to design better optimization strategies to utilize the rich information of different domains. In addition to deep networks, there are also some work [115] that used random forest for DG, and we hope more diverse methods will come.

B. Future research challenges

We summarize some future research challenges for domain generalization.

1) Continuous domain generalization: For many real applications, a system consumes streaming data with non-stationary statistics. In this case, it is of great importance to perform continuous domain generalization that efficiently updates DG models to overcome catastrophic forgetting and adapts to new data. While there are some domain adaptation methods focusing on continuous learning [166], there is only very few investigation on continuous DG [167] whenever this is favorable in real scenarios.

2) Domain generalization to novel categories: The existing DG algorithms usually assume the label space for different domains are the same. A more practical and general setting is to support the generalization on new categories, *i.e.*, both domain and task generalization. This is conceptually similar to the goal of meta-learning and zero-shot learning. Some work [65, 168] proposed zero-shot DG and we expect more work to come in this area.

3) Interpretable domain generalization: Disentanglement-based DG methods decompose a feature into domain-invariant/shared and domain-specific parts, which provide some interpretation to DG. For other categories of methods, there is still a lack of deep understanding of the semantics or characteristics of learned features in DG models. Causality [103] may be one promising tool to understand domain generalization networks and provide interpretations.

4) Large-scale pre-training/self-learning and DG: In recent years, we have witnessed the rapid development of large-scale pre-training/self-learning, such as BERT [169], GPT-3 [170], and Wav2vec [171]. Pre-training on large-scale dataset and then finetuning the model to downstream tasks could improve its performance, where pre-training is beneficial to learn general representations. Therefore, how to design useful and efficient DG methods to help large-scale pre-training/self-learning is worth investigating.

5) Performance evaluation for DG: The recent work [71] pointed out that on several datasets, the performance of some DG methods is almost the same as the baseline method (i.e., empirical risk minimization). We do not take it as the full evidence that DG is not useful in real applications. Instead, we argue that this might be due to the inappropriate evaluation schemes in use today, or the domain gaps not being so large. In more realistic situations such as person ReID where there are obvious domain gaps [63], the improvement of DG is dramatic. Therefore, we stay positive about the value of DG and hope researchers can also find more suitable settings and datasets for the study.

VIII. Conclusion

Generalization has always been an important research topic in machine learning research. In this paper, we review the domain generalization areas by providing in-depth analysis of theories, existing methods, datasets and applications. Then, we thoroughly analyze the methods. Based on our analysis, we provide several potential research challenges that could be the directions of future research. We hope that this survey can provide useful insights to interested researchers and inspire more progress in the future.

References

[1] D. Li, Y. Yang, Y.-Z. Song, and T. M. Hospedales, "Deeper, broader and artier domain generalization," in *ICCV*, 2017, pp. 5542–5550.

[2] R. Caruana, "Multitask learning," *Machine learning*, vol. 28, no. 1, pp. 41–75, 1997.

[3] K. Zhou, Y. Yang, Y. Qiao, and T. Xiang, "Domain adaptive ensemble learning," *arXiv preprint arXiv:2003.07325*, 2020.

[4] S. Pan and Q. Yang, "A survey on transfer learning," *IEEE TKDE*, vol. 22, pp. 1345–1359, 2010.

[5] K. Weiss, T. M. Khoshgoftaar, and D. Wang, "A survey of transfer learning," *Journal of Big data*, vol. 3, no. 1, pp. 1–40, 2016.

[6] F. Zhuang, Z. Qi, K. Duan, D. Xi, Y. Zhu, H. Zhu, H. Xiong, and Q. He, "A comprehensive survey on transfer learning," *Proceedings of the IEEE*, vol. 109, no. 1, pp. 43–76, 2020.

[7] M. Wang and W. Deng, "Deep visual domain adaptation: A survey," *Neurocomputing*, vol. 312, pp. 135–153, 2018.

[8] V. M. Patel, R. Gopalan, R. Li, and R. Chellappa, "Visual domain adaptation: A survey of recent advances," *IEEE signal processing magazine*, vol. 32, no. 3, pp. 53–69, 2015.

[9] R. Vilalta and Y. Drissi, "A perspective view and survey of meta-learning," *Artif. Intell. Rev.*, vol. 18, no. 2, pp. 77–95, 2002.

[10] T. Hospedales, A. Antoniou, P. Micaelli, and A. Storkey, "Meta-learning in neural networks: A survey," *arXiv preprint arXiv:2004.05439*, 2020.

[11] J. Vanschoren, "Meta-learning: A survey," *arXiv preprint arXiv:1810.03548*, 2018.

[12] D. Li, Y. Yang, Y.-Z. Song, and T. M. Hospedales, "Learning to generalize: Meta-learning for domain generalization," in *AAAI*, 2018.

[13] Y. Balaji, S. Sankaranarayanan, and R. Chellappa, "Metareg: Towards domain generalization using meta-regularization," in *NeurIPS*, 2018, pp. 998–1008.

[14] Y. Li, Y. Yang, W. Zhou, and T. M. Hospedales, "Feature-critic networks for heterogeneous domain generalization," in *ICML*, 2019.

[15] Y. Du, J. Xu, H. Xiong, Q. Qiu, X. Zhen, C. G. M. Snoek, and L. Shao, "Learning to learn with variational information bottleneck for domain generalization," in *ECCV*, 2020.

[16] M. Biesialska, K. Biesialska, and M. R. Costa-jussà, "Continual lifelong learning in natural language processing: A survey," *arXiv preprint arXiv:2012.09823*, 2020.

[17] W. Wang, V. W. Zheng, H. Yu, and C. Miao, "A survey of zero-shot learning: Settings, methods, and applications," *ACM Transactions on Intelligent Systems and Technology (TIST)*, vol. 10, no. 2, pp. 1–37, 2019.

[18] Z. JI, H. WANG, Y. YU, and Y. PANG, "A decadal survey of zero-shot image classification," *SCIENTIA SINICA Informationis*, vol. 49, no. 10, pp. 1299–1320, 2019.

[19] H. Rahimian and S. Mehrotra, "Distributionally robust optimization: A review," *arXiv preprint arXiv:1908.05659*, 2019.

[20] S. Ben-David, J. Blitzer, K. Crammer, A. Kulesza, F. Pereira, and J. W. Vaughan, "A theory of learning from different domains," *Machine learning*, vol. 79, no. 1-2, pp. 151–175, 2010.

[21] F. D. Johansson, D. Sontag, and R. Ranganath, "Support and invertibility in domain-invariant representations," in *AISTAS*, 2019, pp. 527–536.

[22] V. Vapnik, E. Levin, and Y. L. Cun, "Measuring the vc-dimension of a learning machine," *Neural computation*, vol. 6, no. 5, pp. 851–876, 1994.

[23] S. Ben-David, J. Blitzer, K. Crammer, F. Pereira *et al.*, "Analysis of representations for domain adaptation," in *NIPS*, vol. 19, 2007.

[24] G. Blanchard, G. Lee, and C. Scott, "Generalizing from several related classification tasks to a new unlabeled sample," in *NeurIPS*, 2011, pp. 2178–2186.

[25] K. Muandet, D. Balduzzi, and B. Schölkopf, "Domain generalization via invariant feature representation," in *ICML*, 2013, pp. 10–18.

[26] A. A. Deshmukh, Y. Lei, S. Sharma, U. Dogan, J. W. Cutler, and C. Scott, "A generalization error bound for multi-class domain generalization," *arXiv:1905.10392*, 2019.

[27] I. Albuquerque, J. Monteiro, T. H. Falk, and I. Mitliagkas, "Adversarial target-invariant representation learning for domain generalization," *arXiv preprint arXiv:1911.00804*, 2019.

[28] X. Yue, Y. Zhang, S. Zhao, A. Sangiovanni-Vincentelli, K. Keutzer, and B. Gong, "Domain randomization and pyramid consistency: Simulation-to-real generalization without accessing target domain data," in *ICCV*, 2019, pp. 2100–2110.

[29] J. Tobin, R. Fong, A. Ray, J. Schneider, W. Zaremba, and P. Abbeel, "Domain randomization for transferring deep neural networks from simulation to the real world," in *IROS*, 2017, pp. 23–30.

[30] X. B. Peng, M. Andrychowicz, W. Zaremba, and P. Abbeel, "Sim-to-real transfer of robotic control with dynamics randomization," in *ICRA*. IEEE, 2018, pp. 1–8.

[31] R. Khirodkar, D. Yoo, and K. Kitani, "Domain randomization for scene-specific car detection and pose estimation," in *WACV*. IEEE, 2019, pp. 1932–1940.

[32] J. Tremblay, A. Prakash, D. Acuna, M. Brophy, V. Jampani, C. Anil, T. To, E. Cameracci, S. Boochoon, and S. Birchfield, "Training deep networks with synthetic data: Bridging the reality gap by domain randomization," in *CVPR Workshop*, 2018.

[33] A. Prakash, S. Boochoon, M. Brophy, D. Acuna, E. Cameracci, G. State, O. Shapira, and S. Birchfield, "Structured domain randomization: Bridging the reality gap by context-aware synthetic data," in *ICRA*. IEEE, 2019, pp. 7249–7255.

[34] N. H. Nazari and A. Kovashka, "Domain generalization using shape representation," in *ECCV*, 2020, pp. 666–670.

[35] S. Shankar, V. Piratla, S. Chakrabarti, S. Chaudhuri, P. Jyothi, and S. Sarawagi, "Generalizing across domains via cross-gradient training," in *ICLR*, 2018.

[36] R. Volpi, H. Namkoong, O. Sener, J. C. Duchi, V. Murino, and S. Savarese, "Generalizing to unseen domains via adversarial data augmentation," in *NeurIPS*, 2018, pp. 5334–5344.

[37] K. Zhou, Y. Yang, T. Hospedales, and T. Xiang, "Deep domain-adversarial image generation for domain generalisation," in *AAAI*, 2020.

[38] M. M. Rahman, C. Fookes, M. Baktashmotlagh, and S. Sridharan, "Multi-component image translation for deep domain generalization," in *WACV*. IEEE, 2019, pp. 579–588.

[39] F. Qiao, L. Zhao, and X. Peng, "Learning to learn single domain generalization," in *CVPR*, 2020, pp. 12 556–12 565.

[40] A. H. Liu, Y.-C. Liu, Y.-Y. Yeh, and Y.-C. F. Wang, "A unified feature disentangler for multi-domain image translation and manipulation," in *NeurIPS*, 2018, pp. 2590–2599.

[41] T.-D. Truong, C. N. Duong, K. Luu, and M.-T. Tran, "Recognition in unseen domains: Domain generalization via universal non-volume preserving models," *arXiv preprint:1905.13040*, 2019.

[42] K. Zhou, Y. Yang, T. M. Hospedales, and T. Xiang, "Learning to generate novel domains for domain generalization," *ArXiv*, vol. abs/2007.03304, 2020.

[43] N. Somavarapu, C.-Y. Ma, and Z. Kira, "Frustratingly simple domain generalization via image stylization," *arXiv:2006.11207*, 2020.

[44] Y. Zhao, Z. Zhong, F. Yang, Z. Luo, Y. Lin, S. Li, and N. Sebe, "Learning to generalize unseen domains via memory-based multi-source meta-learning for person re-identification," *arXiv preprint arXiv:2012.00417*, 2020.

[45] V. K. Garg, A. Kalai, K. Ligett, and Z. S. Wu, "Learn to expect the unexpected: Probably approximately correct domain generalization," *arXiv preprint arXiv:2002.05660*, 2020.

[46] F. M. Carlucci, P. Russo, T. Tommasi, and B. Caputo, "Hallucinating agnostic images to generalize across domains," in *CVPR Workshop*, 2019, pp. 0–0.

[47] W. Wang, S. Liao, F. Zhao, C. Kang, and L. Shao, "Domainmix: Learning generalizable person re-identification without human annotations," *arXiv:2011.11953*, 2020.

[48] Y. Wang, H. Li, and A. C. Kot, "Heterogeneous domain generalization via domain mixup," in *ICASSP*, 2020, pp. 3622–3626.

[49] K. Zhou, Y. Yang, Y. Qiao, and T. Xiang, "Domain generalization with mixstyle," in *ICLR*, 2021.

[50] T.-D. Truong, K. Luu, C.-N. Duong, N. Le, and M.-T. Tran, "Image alignment in unseen domains via domain deep generalization," *arXiv preprint arXiv:1905.12028*, 2019.

[51] A. Robey, G. J. Pappas, and H. Hassani, "Model-based domain generalization," *arXiv preprint arXiv:2102.11436*, 2021.

[52] G. Blanchard, A. A. Deshmukh, U. Dogan, G. Lee, and C. Scott, "Domain generalization by marginal transfer learning," *arXiv preprint arXiv:1711.07910*, 2017.

[53] T. Grubinger, A. Birlutiu, H. Schöner, T. Natschläger, and T. Heskes, "Domain generalization based on transfer component analysis," in *International Work-Conference on Artificial Neural Networks*. Springer, 2015, pp. 325–334.

[54] C. Gan, T. Yang, and B. Gong, "Learning attributes equals multi-source domain generalization," in *CVPR*, 2016, pp. 87–97.

[55] Y. Li, M. Gong, X. Tian, T. Liu, and D. Tao, "Domain generalization via conditional invariant representations," in *AAAI*, 2018.

[56] M. Ghifary, D. Balduzzi, W. B. Kleijn, and M. Zhang, "Scatter component analysis: A unified framework for domain adaptation and domain generalization," *IEEE TPAMI*, vol. 39, no. 7, pp. 1414–1430, 2016.

[57] S. Erfani, M. Baktashmotlagh, M. Moshtaghi, V. Nguyen, C. Leckie, J. Bailey, and R. Kotagiri, "Robust domain generalisation by enforcing distribution invariance," in *AAAI*, 2016, pp. 1455–1461.

[58] S. Hu, K. Zhang, Z. Chen, and L. Chan, "Domain generalization via multidomain discriminant analysis," in *UAI*, vol. 35, 2019.

[59] M. Ghifary, W. Kleijn, M. Zhang, and D. Balduzzi, "Domain generalization for object recognition with multi-task autoencoders," *ICCV*, pp. 2551–2559, 2015.

[60] S. Motiian, M. Piccirilli, D. A. Adjeroh, and G. Doretto, "Unified deep supervised domain adaptation and generalization," in *ICCV*, 2017, pp. 5715–5725.

[61] M. Segù, A. Tonioni, and F. Tombari, "Batch normalization embeddings for deep domain generalization," *arXiv preprint arXiv:2011.12672*, 2020.

[62] X. Jin, C. Lan, W. Zeng, Z. Chen, and L. Zhang, "Style normalization and restitution for generalizable person re-identification," in *CVPR*, 2020, pp. 3143–3152.

[63] X. Jin, C. Lan, W. Zeng, and Z. Chen, "Style normalization and restitution for domain generalization and adaptation," *arXiv preprint arXiv:2101.00588*, 2021.

[64] H. Li, S. J. Pan, S. Wang, and A. Kot, "Domain generalization with adversarial feature learning," in *CVPR*, 2018, pp. 5400–5409.

[65] U. Maniyar, A. A. Deshmukh, U. Dogan, V. N. Balasubramanian *et al.*, "Zero shot domain generalization," *arXiv preprint arXiv:2008.07443*, 2020.

[66] D.-K. Han and J.-H. Jeong, "Domain generalization for session-independent brain-computer interface," *arXiv preprint arXiv:2012.03533*, 2020.

[67] H. Zheng, Y. Yang, J. Yin, Y. Li, R. Wang, and M. Xu, "Deep domain generalization combining a priori diagnosis knowledge toward cross-domain fault diagnosis of rolling bearing," *IEEE Trans. on Instrumentation and Measurement*, vol. 70, pp. 1–11, 2020.

[68] Y. Liao, R. Huang, J. Li, Z. Chen, and W. Li, "Deep semisupervised

domain generalization network for rotary machinery fault diagnosis under variable speed," *IEEE Transactions on Instrumentation and Measurement*, vol. 69, no. 10, pp. 8064–8075, 2020.

[69] K. Ayodele, W. Ikezogwo, M. Komolafe, and P. Ogunbona, "Supervised domain generalization for integration of disparate scalp eeg datasets for automatic epileptic seizure detection," *Computers in Biology and Medicine*, vol. 120, p. 103757, 2020.

[70] E. Chen, T. S. Mathai, V. Sarode, H. Choset, and J. Galeotti, "A study of domain generalization on ultrasound-based multi-class segmentation of arteries, veins, ligaments, and nerves using transfer learning," *arXiv preprint arXiv:2011.07019*, 2020.

[71] I. Gulrajani and D. Lopez-Paz, "In search of lost domain generalization," in *ICLR*, 2021.

[72] X. Li, W. Zhang, H. Ma, Z. Luo, and X. Li, "Domain generalization in rotating machinery fault diagnostics using deep neural networks," *Neurocomputing*, vol. 403, pp. 409–420, 2020.

[73] H. Liu, P. Song, and R. Ding, "Towards domain generalization in underwater object detection," in *ICIP*. IEEE, 2020, pp. 1971–1975.

[74] D. Mahajan, S. Tople, and A. Sharma, "Domain generalization using causal matching," *ArXiv*, vol. abs/2006.07500, 2020.

[75] S. Seo, Y. Suh, D. Kim, J. Han, and B. Han, "Learning to optimize domain specific normalization for domain generalization," in *ECCV*, 2020.

[76] W. Zhang, M. Ragab, and R. Sagarna, "Robust domain-free domain generalization with class-aware alignment," in *ICASSP*, 2021.

[77] Y. Ganin and V. Lempitsky, "Unsupervised domain adaptation by backpropagation," in *ICML*, 2015, pp. 1180–1189.

[78] Y. Ganin, E. Ustinova, H. Ajakan, P. Germain, H. Larochelle, F. Laviolette, M. Marchand, and V. Lempitsky, "Domain-adversarial training of neural networks," *J. Mach. Learn. Res.*, vol. 17, pp. 59:1–59:35, 2016.

[79] R. Gong, W. Li, Y. Chen, and L. V. Gool, "Dlow: Domain flow for adaptation and generalization," in *CVPR*, 2019, pp. 2477–2486.

[80] Y. Li, X. Tian, M. Gong, Y. Liu, T. Liu, K. Zhang, and D. Tao, "Deep domain generalization via conditional invariant adversarial networks," in *ECCV*, 2018, pp. 624–639.

[81] R. Shao, X. Lan, J. Li, and P. C. Yuen, "Multi-adversarial discriminative deep domain generalization for face presentation attack detection," in *CVPR*, 2019, pp. 10 023–10 031.

[82] S. Zhao, M. Gong, T. Liu, H. Fu, and D. Tao, "Domain generalization via entropy regularization," in *NeurIPS*, vol. 33, 2020.

[83] T. Matsuura and T. Harada, "Domain generalization using a mixture of multiple latent domains," in *AAAI*, 2020.

[84] A. Sicilia, X. Zhao, and S. J. Hwang, "Domain adversarial neural networks for domain generalization: When it works and how to improve," *arXiv preprint arXiv:2102.03924*, 2021.

[85] M. M. Rahman, C. Fookes, M. Baktashmotlagh, and S. Sridharan, "Correlation-aware adversarial domain adaptation and generalization," *Pattern Recognition*, vol. 100, p. 107124, 2020.

[86] Z. Wang, Q. Wang, C. Lv, X. Cao, and G. Fu, "Unseen target stance detection with adversarial domain generalization," in *IJCNN*, 2020, pp. 1–8.

[87] Y. Jia, J. Zhang, S. Shan, and X. Chen, "Single-side domain generalization for face anti-spoofing," in *CVPR*, 2020, pp. 8484–8493.

[88] M. Arjovsky, L. Bottou, I. Gulrajani, and D. Lopez-Paz, "Invariant risk minimization," *arXiv preprint arXiv:1907.02893*, 2019.

[89] R. Guo, P. Zhang, H. Liu, and E. Kiciman, "Out-of-distribution prediction with invariant risk minimization: The limitation and an effective fix," *arXiv preprint arXiv:2101.07732*, 2021.

[90] K. Ahuja, J. Wang, A. Dhurandhar, K. Shanmugam, and K. R. Varshney, "Empirical or invariant risk minimization? a sample complexity perspective," *arXiv preprint arXiv:2010.16412*, 2020.

[91] Y. J. Choe, J. Ham, and K. Park, "An empirical study of invariant risk minimization," *arXiv preprint arXiv:2004.05007*, 2020.

[92] E. Rosenfeld, P. Ravikumar, and A. Risteski, "The risks of invariant risk minimization," *arXiv preprint arXiv:2010.05761*, 2020.

[93] Z. Xu, W. Li, L. Niu, and D. Xu, "Exploiting low-rank structure from latent domains for domain generalization," in *ECCV*. Springer, 2014, pp. 628–643.

[94] A. Khosla, T. Zhou, T. Malisiewicz, A. A. Efros, and A. Torralba, "Undoing the damage of dataset bias," in *ECCV*. Springer, 2012, pp. 158–171.

[95] L. Niu, W. Li, and D. Xu, "Multi-view domain generalization for visual recognition," in *ICCV*, 2015, pp. 4193–4201.

[96] Z. Ding and Y. Fu, "Deep domain generalization with structured low-rank constraint," *IEEE TIP*, vol. 27, no. 1, pp. 304–313, 2017.

[97] A. Zunino, S. A. Bargal, R. Volpi, M. Sameki, J. Zhang, S. Sclaroff, V. Murino, and K. Saenko, "Explainable deep classification models for domain generalization," *arXiv preprint arXiv:2003.06498*, 2020.

[98] A. A. Deshmukh, A. Bansal, and A. Rastogi, "Domain2vec: Deep domain generalization," *arXiv preprint arXiv:1807.02919*, 2018.

[99] W. Li, Z. Xu, D. Xu, D. Dai, and L. Van Gool, "Domain generalization and adaptation using low rank exemplar svms," *TPAMI*, vol. 40, no. 5, pp. 1114–1127, 2017.

[100] V. Piratla, P. Netrapalli, and S. Sarawagi, "Efficient domain generalization via common-specific low-rank decomposition," in *ICML*, 2020, pp. 7728–7738.

[101] M. Ilse, J. M. Tomczak, C. Louizos, and M. Welling, "Diva: Domain invariant variational autoencoders," in *Proceedings of the Third Conference on Medical Imaging with Deep Learning*, 2020.

[102] X. Peng, Z. Huang, X. Sun, and K. Saenko, "Domain agnostic learning with disentangled representations," in *ICML*, 2019.

[103] C. Liu, X. Sun, J. Wang, T. Li, T. Qin, W. Chen, and T.-Y. Liu, "Learning causal semantic representation for out-of-distribution prediction," *arXiv preprint arXiv:2011.01681*, 2020.

[104] M. Mancini, S. R. Bulò, B. Caputo, and E. Ricci, "Best sources forward: domain generalization through source-specific nets," in *ICIP*, 2018, pp. 1353–1357.

[105] A. D'Innocente and B. Caputo, "Domain generalization with domain-specific aggregation modules," in *German Conference on Pattern Recognition*. Springer, 2018, pp. 187–198.

[106] W. He, H. Zheng, and J. Lai, "Domain attention model for domain generalization in object detection," in *PRCV*, 2018, pp. 27–39.

[107] M. Mancini, S. R. Bulo, B. Caputo, and E. Ricci, "Robust place categorization with deep domain generalization," *IEEE Robotics and Automation Letters*, vol. 3, no. 3, pp. 2093–2100, 2018.

[108] Q. Dou, D. C. de Castro, K. Kamnitsas, and B. Glocker, "Domain generalization via model-agnostic learning of semantic features," in *NeurIPS*, 2019.

[109] K. Chen, D. Zhuang, and J. M. Chang, "Discriminative adversarial domain generalization with meta-learning based cross-domain validation," *arXiv preprint arXiv:2011.00444*, 2020.

[110] H. Sharifi-Noghabi, H. Asghari, N. Mehrasa, and M. Ester, "Domain generalization via semi-supervised meta learning," *arXiv preprint arXiv:2009.12658*, 2020.

[111] B. Wang, M. Lapata, and I. Titov, "Meta-learning for domain generalization in semantic parsing," *arXiv:2010.11988*, 2020.

[112] Z. Huang, H. Wang, E. P. Xing, and D. Huang, "Self-challenging improves cross-domain generalization," in *ECCV*, vol. 2, 2020.

[113] Y. Shi, J. Seely, P. H. Torr, N. Siddharth, A. Hannun, N. Usunier, and G. Synnaeve, "Gradient matching for domain generalization," *arXiv preprint arXiv:2104.09937*, 2021.

[114] F. M. Carlucci, A. D'Innocente, S. Bucci, B. Caputo, and T. Tommasi, "Domain generalization by solving jigsaw puzzles," in *CVPR*, 2019, pp. 2229–2238.

[115] J. Ryu, G. Kwon, M.-H. Yang, and J. Lim, "Generalized convolutional forest networks for domain generalization and visual recognition," in *ICLR*, 2019.

[116] D. Li, J. Zhang, Y. Yang, C. Liu, Y.-Z. Song, and T. M. Hospedales, "Episodic training for domain generalization," in *CVPR*, 2019, pp. 1446–1455.

[117] C. Shorten and T. M. Khoshgoftaar, "A survey on image data augmentation for deep learning," *Journal of Big Data*, vol. 6, no. 1, pp. 1–48, 2019.

[118] D. P. Kingma and M. Welling, "Auto-encoding variational bayes," *arXiv:1312.6114*, 2013.

[119] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial networks," in *NIPS*, 2014.

[120] H. Zhang, M. Cisse, Y. N. Dauphin, and D. Lopez-Paz, "mixup: Beyond empirical risk minimization," in *ICLR*, 2018.

[121] A. Anoosheh, E. Agustsson, R. Timofte, and L. Van Gool, "Combogan: Unrestrained scalability for image domain translation," in *CVPR Workshop*, 2018, pp. 783–790.

[122] A. Gretton, K. M. Borgwardt, M. J. Rasch, B. Schölkopf, and A. Smola, "A kernel two-sample test," *The Journal of Machine Learning Research*, vol. 13, no. 1, pp. 723–773, 2012.

[123] I. Tolstikhin, O. Bousquet, S. Gelly, and B. Schoelkopf, "Wasserstein auto-encoders," *arXiv preprint arXiv:1711.01558*, 2017.

[124] X. Huang and S. Belongie, "Arbitrary style transfer in real-time with adaptive instance normalization," in *ICCV*, 2017, pp. 1501–1510.

[125] Y. Bengio, A. Courville, and P. Vincent, "Representation learning: A review and new perspectives," *IEEE TPAMI*, vol. 35, no. 8, pp. 1798–1828, 2013.

[126] C. Cortes and V. Vapnik, "Support-vector networks," *Machine learning*,

vol. 20, no. 3, pp. 273–297, 1995.

[127] S. J. Pan, I. Tsang, J. T. Kwok, and Q. Yang, "Domain adaptation via transfer component analysis," *IEEE TNN*, vol. 22, pp. 199–210, 2011.

[128] F. Zhou, Z. Jiang, C. Shui, B. Wang, and B. Chaib-draa, "Domain generalization with optimal transport and metric learning," *ArXiv*, vol. abs/2007.10573, 2020.

[129] X. Peng, Q. Bai, X. Xia, Z. Huang, K. Saenko, and B. Wang, "Moment matching for multi-source domain adaptation," in *ICCV*, 2019, pp. 1406–1415.

[130] X. Pan, P. Luo, J. Shi, and X. Tang, "Two at once: Enhancing learning and generalization capacities via ibn-net," in *ECCV*, 2018, pp. 464–479.

[131] J. Jia, Q. Ruan, and T. M. Hospedales, "Frustratingly easy person re-identification: Generalizing person re-id in practice," 2019.

[132] H. Nam and H.-E. Kim, "Batch-instance normalization for adaptively style-invariant neural networks," in *NeurIPS*, 2018, pp. 2558–2567.

[133] E. Tzeng, J. Hoffman, N. Zhang, K. Saenko, and T. Darrell, "Deep domain confusion: Maximizing for domain invariance," *arXiv preprint arXiv:1412.3474*, 2014.

[134] J. Wang, W. Feng, Y. Chen, H. Yu, M. Huang, and P. S. Yu, "Visual domain adaptation with manifold embedded distribution alignment," in *ACMMM*, 2018, pp. 402–410.

[135] J. Wang, Y. Chen, W. Feng, H. Yu, M. Huang, and Q. Yang, "Transfer learning with dynamic distribution adaptation," *ACM TIST*, vol. 11, no. 1, pp. 1–25, 2020.

[136] B. Sun, J. Feng, and K. Saenko, "Return of frustratingly easy domain adaptation," in *AAAI*, 2016.

[137] B. Sun and K. Saenko, "Deep coral: Correlation alignment for deep domain adaptation," in *ECCV*, 2016, pp. 443–450.

[138] X. Peng and K. Saenko, "Synthetic to real adaptation with generative correlation alignment networks," in *WACV*. IEEE, 2018, pp. 1982–1991.

[139] D. Ulyanov, A. Vedaldi, and V. Lempitsky, "Improved texture networks: Maximizing quality and diversity in feed-forward stylization and texture synthesis," in *CVPR*, 2017, pp. 6924–6932.

[140] V. Dumoulin, J. Shlens, and M. Kudlur, "A learned representation for artistic style," *ICLR*, 2017.

[141] D. Ulyanov, A. Vedaldi, and V. Lempitsky, "Instance normalization: The missing ingredient for fast stylization," *arXiv preprint arXiv:1607.08022*, 2016.

[142] X. Jin, C. Lan, W. Zeng, and Z. Chen, "Feature alignment and restoration for domain generalization and adaptation," in *NeurIPS*, 2020.

[143] K. Ahuja, K. Shanmugam, K. Varshney, and A. Dhurandhar, "Invariant risk minimization games," in *ICML*, 2020, pp. 145–155.

[144] A. Sonar, V. Pacelli, and A. Majumdar, "Invariant policy optimization: Towards stronger generalization in reinforcement learning," *arXiv preprint arXiv:2006.01096*, 2020.

[145] D. Krueger, E. Caballero, J. Jacobsen, A. Zhang, J. Binas, R. L. Priol, and A. C. Courville, "Out-of-distribution generalization via risk extrapolation (rex)," *ArXiv*, vol. abs/2003.00688, 2020.

[146] J. Mitrovic, B. McWilliams, J. C. Walker, L. H. Buesing, and C. Blundell, "Representation learning via invariant causal mechanisms," in *International Conference on Learning Representations*, 2021. [Online]. Available: https://openreview.net/forum?id=9p2ekP904Rs

[147] C. Finn, P. Abbeel, and S. Levine, "Model-agnostic meta-learning for fast adaptation of deep networks," in *ICML*, 2017.

[148] J. Snell, K. Swersky, and R. S. Zemel, "Prototypical networks for few-shot learning," in *NeurIPS*, 2017.

[149] A. Santoro, S. Bartunov, M. Botvinick, D. Wierstra, and T. Lillicrap, "Meta-learning with memory-augmented neural networks," in *ICML*, 2016, pp. 1842–1850.

[150] L. Jing and Y. Tian, "Self-supervised visual feature learning with deep neural networks: A survey," *IEEE TPAMI*, 2020.

[151] K. Saenko, B. Kulis, M. Fritz, and T. Darrell, "Adapting visual category models to new domains," in *ECCV*. Springer, 2010, pp. 213–226.

[152] C. Fang, Y. Xu, and D. N. Rockmore, "Unbiased metric learning: On the utilization of multiple datasets and web images for softening bias," in *ICCV*, 2013, pp. 1657–1664.

[153] H. Venkateswara, J. Eusebio, S. Chakraborty, and S. Panchanathan, "Deep hashing network for unsupervised domain adaptation," *CVPR*, pp. 5385–5394, 2017.

[154] S. Beery, G. V. Horn, and P. Perona, "Recognition in terra incognita," in *ECCV*, 2018.

[155] X. Peng, Q. Bai, X. Xia, Z. Huang, K. Saenko, and B. Wang, "Moment matching for multi-source domain adaptation," *ICCV*, pp. 1406–1415, 2019.

[156] R. R. Brinkman, M. Gasparetto, S.-J. J. Lee, A. J. Ribickas, J. Perkins, W. Janssen, R. Smiley, and C. Smith, "High-content flow cytometry and temporal data analysis for defining a cellular signature of graft-versus-host disease," *Biology of Blood and Marrow Transplantation*, vol. 13, no. 6, pp. 691–700, 2007.

[157] M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, and B. Schiele, "The cityscapes dataset for semantic urban scene understanding," in *CVPR*, 2016, pp. 3213–3223.

[158] S. R. Richter, V. Vineet, S. Roth, and V. Koltun, "Playing for data: Ground truth from computer games," in *ECCV*. Springer, 2016, pp. 102–118.

[159] G. Ros, L. Sellart, J. Materzynska, D. Vazquez, and A. M. Lopez, "The synthia dataset: A large collection of synthetic images for semantic segmentation of urban scenes," in *CVPR*, 2016, pp. 3234–3243.

[160] Y. Wang, S. Liao, and L. Shao, "Surpassing real-world source training data: Random 3d characters for generalizable person re-identification," in *ACMMM*, 2020, pp. 3422–3430.

[161] J. Song, Y. Yang, Y.-Z. Song, T. Xiang, and T. M. Hospedales, "Generalizable person re-identification by domain-invariant mapping network," in *CVPR*, 2019.

[162] G. Brockman, V. Cheung, L. Pettersson, J. Schneider, J. Schulman, J. Tang, and W. Zaremba, "Openai gym," *arXiv preprint arXiv:1606.01540*, 2016.

[163] M. Little, P. McSharry, E. Hunter, J. Spielman, and L. Ramig, "Suitability of dysphonia measurements for telemonitoring of parkinson's disease," *Nature Precedings*, pp. 1–1, 2008.

[164] P. W. Koh, S. Sagawa, H. Marklund, S. M. Xie, M. Zhang, A. Balsubramani, W. Hu, M. Yasunaga, R. L. Phillips, S. Beery *et al.*, "Wilds: A benchmark of in-the-wild distribution shifts," *arXiv preprint arXiv:2012.07421*, 2020.

[165] J. Blitzer, R. McDonald, and F. Pereira, "Domain adaptation with structural correspondence learning," in *EMNLP*, 2006, pp. 120–128.

[166] H. Wang, H. He, and D. Katabi, "Continuously indexed domain adaptation," *arXiv preprint arXiv:2007.01807*, 2020.

[167] D. Li, Y. Yang, Y.-Z. Song, and T. Hospedales, "Sequential learning for domain generalization," in *ECCV*, 2020, pp. 603–619.

[168] M. Mancini, Z. Akata, E. Ricci, and B. Caputo, "Towards recognizing unseen categories in unseen domains," in *ECCV*, 2020.

[169] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," *arXiv preprint arXiv:1810.04805*, 2018.

[170] T. B. Brown, B. Mann, N. Ryder, M. Subbiah, J. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell *et al.*, "Language models are few-shot learners," in *NeurIPS*, 2020.

[171] A. Baevski, H. Zhou, A. Mohamed, and M. Auli, "wav2vec 2.0: A framework for self-supervised learning of speech representations," *arXiv preprint arXiv:2006.11477*, 2020.