Addressing Data Scarcity in Domain Generalization for Computer Vision Applications in Image Classification

by

Kimathi Kaai

A thesis
presented to the University of Waterloo
in fulfillment of the
thesis requirement for the degree of
Master of Applied Science
in
Systems Design Engineering

Waterloo, Ontario, Canada, 2024

Author's Declaration

I hereby declare that I am the sole author of this thesis. This is a true copy of the thesis, including any required final revisions, as accepted by my examiners.

I understand that my thesis may be made electronically available to the public.

Abstract

Domain generalization (DG) for image classification is a crucial task in machine learning that focuses on transferring domain-invariant knowledge from multiple source domains to an unseen target domain. Traditional DG methods assume that classes of interest are present across multiple domains (*domain-shared*), which helps mitigate spurious correlations between domain and class. However, in real-world scenarios, data scarcity often leads to classes being present in only a single domain (*domain-linked*), resulting in poor generalization performance. This thesis introduces the domain-linked DG task and proposes a novel methodology to address this challenge.

This thesis proposes `FOND` a `F`airness-inspired c`ON`trastive learning objective for `D`omain-linked domain generalization, which leverages domain-shared classes to learn domain-invariant representations for domain-linked classes. `FOND` is designed to enhance generalization by minimizing the impact of task-irrelevant domain-specific features.

The theoretical analysis in this thesis extends existing domain adaptation error bounds to the domain-linked DG task, providing insights into the factors that influence generalization performance. Key theoretical findings include the understanding that domain-shared classes typically have more samples and learn domain-invariant features more effectively than domain-linked classes. This analysis informs the design of `FOND`, ensuring that it addresses the unique challenges of domain-linked DG.

Furthermore, experiments are performed across multiple datasets and experimental settings to evaluate the effectiveness of various current methodologies. The proposed method achieves state-of-the-art performance in domain-linked DG tasks, with minimal trade-offs in the performance of domain-shared classes. Experimental results highlight the impact of shared-class settings, total class size, and inter-domain variations on the generalizability of domain-linked classes. Visualizations of learned representations further illustrate the robustness of `FOND` in capturing domain-invariant features.

In summary, this thesis advocates future DG research for domain-linked classes by (1) theoretically and experimentally analyzing the factors impacting domain-linked class representation learning, (2) demonstrating the ineffectiveness of current state-of-the-art DG approaches, and (3) proposing an algorithm to learn generalizable representations for domain-linked classes by transferring useful representations from domain-shared ones.

Acknowledgements

First, I would like to thank my two supervisors, Asst. Prof. Sirisha Rambhatla and Prof. Alexander Wong, for granting me access to their immense knowledge and continual support. I especially thank Asst. Prof. Sirisha, for initiating my journey into academic research and for the close and deliberate guidance.

It was a privilege to be part of both the Vision and Image Processing Lab and the Critical ML Lab. It was truly a blessing to be able to work alongside other academic researchers and build meaningful relationships. I would also like to thank Jinman (Eddie) Park for being a mentor and a friend of mine throughout my MASc journey.

I would also like to thank Prof. David Clausi and Prof. Mark Crowley for being part of my thesis committee and providing me with constructive feedback.

Dedication

This is dedicated first to God, who, by His grace, opened doors for me to pursue a MASc in this field. Secondly, I dedicate this to my loving parents and brother, who have been a consistent source of encouragement and counsel throughout this journey.

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

Machine learning (ML) has seen significant success across various fields, such as computer vision [56], natural language processing [51], and healthcare [32]. The primary objective is to create models capable of learning general and predictive insights from training data, which can be applied to new data. Traditional ML models are typically trained assuming that training and testing data are identically and independently distributed *i.i.d.* [67].

However, this assumption often does not hold in real-world scenarios. There are often variations (i.e. *domain-shifts*) between training and testing data, leading to severe degradation in predictive performance [73, 88]. These domain shifts, for example, can be identified as differences in imaging equipment for medical imaging and changes in lightning and product types for defect detection tasks in intelligent manufacturing systems. Gathering data, including enough potential domain shifts to train ML models, is costly and sometimes not feasible. Consequently, improving the generalization capability of ML models is crucial in both industry and academic research.

1.1 Domain Generalization

The research field of *domain generalization* (DG) addresses these challenges by developing machine learning techniques to learn discriminative representations that can generalize to data distributions (*domains*) different from those observed during training, i.e. *out-of-distribution* (OOD) [88]. In other words, the domain generalization task relaxes the *i.i.d* assumption by allowing source domain distribution(s) at the train time to be different from the target domain (test data). Given this goal, the guiding principle in modern

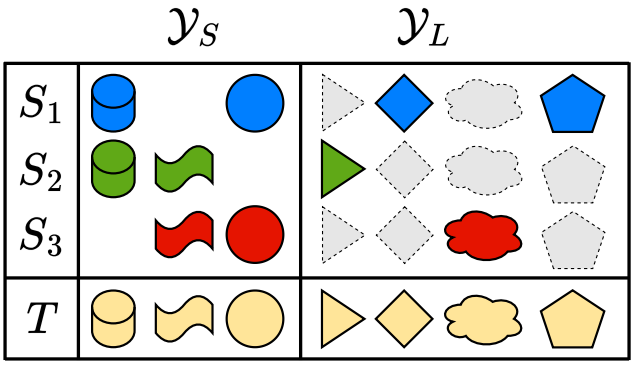


Figure 1.1: **Illustrating domain-linked ($\mathcal{Y}_L$) and domain-shared ($\mathcal{Y}_S$) classes.** This figure illustrates a shape classification task where the domains are represented by color. During training, some classes are expressed in multiple domains (e.g., circle) while others are expressed in only one domain (e.g., triangle).

DG algorithms is to learn representations that are invariant to source (training) domains and hence generalizable to unseen target (test) domains [4, 79]. As a result, recent works aim to explicitly reduce the representation discrepancy between multiple source-domains, by leveraging distribution-alignment [44, 52], adversarial networks [30, 80], domain-based feature-alignment [29, 55, 76], meta-learning, and few-shot approaches [20, 34, 49, 62, 82].

1.2 Domain-Linked Class Generalization

However, existing DG methods rely on classes being observed in multiple source domains and/or focus only on the overall accuracy. In the real world, however, classes of interest may often be observed in specific domains (*domain-linked*, $\mathcal{Y}_L$), setting them apart from those observed in multiple domains (*domain-shared*, $\mathcal{Y}_S$). For clarity, refere to Fig. 1.1 for a visual illustration of domain-linked ($\mathcal{Y}_L$) and domain-shared ($\mathcal{Y}_S$) classes. This figure illustrates a shape classification task where the domains are represented by color. During training, some classes are expressed in multiple domains (e.g., circle) while others are expressed in only one domain (e.g., triangle).

The lack of data diversity in domain-linked classes leads to generalization challenges in applications such as quality control in manufacturing and food inspection, where certain defects (classes) are often unique/linked to particular types of products (domains) [6]. Similar observations are made in healthcare [10], autonomous driving [48], and fraud detec-

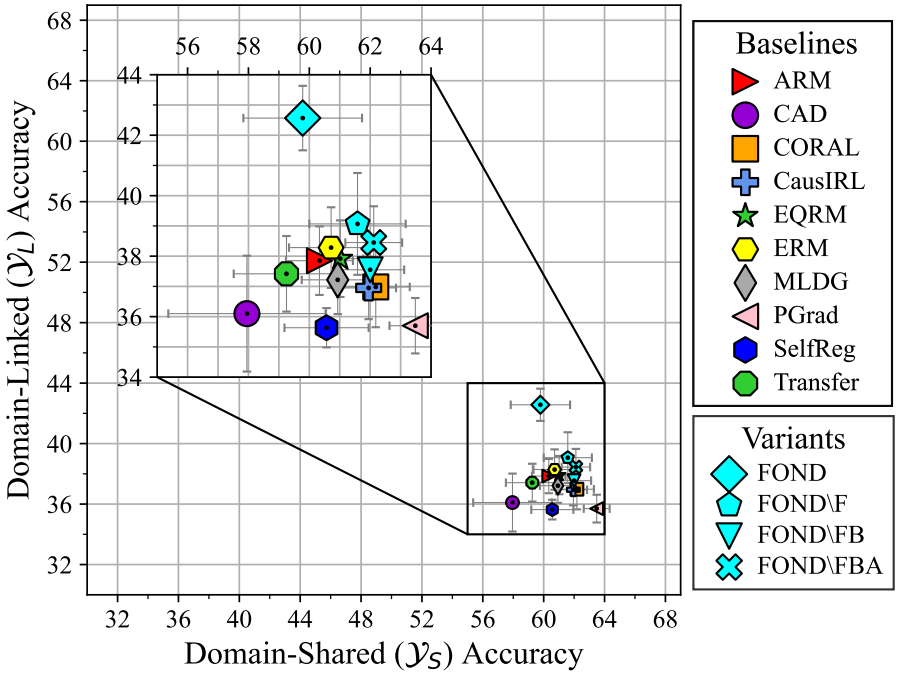


Figure 1.2: **Domain-linked ($\mathcal{Y}_L$) and domain-shared ($\mathcal{Y}_S$) performance discrepancies.** This graph displays the domain-linked and domain-shared classification accuracies of current DG methods (referred to as *Baselines*) alongside the proposed methodology, `FOND`, and its variations. These values represent averages across the evaluation datasets. Chapter 5 introduces these baselines and datasets along with a more detailed analysis.

tion [2] applications where classes of interest may *only* be tied to particular demographics, regions, etc. In this thesis, we focus on prominent DG datasets (which will be introduced later in this section) to enable comparisons with established DG methodologies and to encourage future research on application-specific settings.

Challenges with domain-linked class generalization arise because models tend to exhibit extreme bias towards spurious domain-specific patterns in the absence of observations across domains [39, 83]. As shown in Fig. 1.2, the performance discrepancies between domain-linked and domain-shared classes can be extremely severe. Consequently, we seek to improve the generalizability of domain-linked classes while minimizing the domain-shared performance trade-offs; this task has not been sufficiently studied.

1.3 Thesis Contributions and Overview

Motivated by the success of pre-training with classes/objectives different from downstream tasks [22], this thesis explores the question — *Can we transfer useful representations from domain-shared classes to domain-linked classes?* This thesis answers this in the affirmative, identifying the cases where it makes sense to utilize the representations learned by domain-shared classes to boost the performance of the domain-linked ones. The contributions of this thesis can be summarized as follows:

1. Identifying the severe performance discrepancy between domain-linked and domain-shared class performance in existing domain generalization methods to further stimulate research in this area.
2. Proposing `FOND`; the first method for improving domain-linked class generalization using representations learned from domain-shared classes.
3. Leveraging existing domain adaptation theory to analyze when and why domain-shared classes usually generalize better than domain-linked ones.
4. Accomplishing state-of-the-art performance for domain-linked DG while providing practical insights into the data scarcity conditions that affect their generalizability.

Chapter 2 formalizes definitions related to domain-linked class generalization and offers a review of existing literature on important theoretical motivations and methodologies for domain generalization. Furthermore, Chapter 3 provides some preliminary theoretical analysis on the factors impacting domain-linked class performance. Leveraging these insights, Chapter 4 introduces a contrastive learning and fair-learning inspired objective to disentangle spurious correlations between domain and class, denoted `FOND`, a `F`airness-inspired and c`ON`trastive `D`omain-linked learning strategy. Chapter 5 evaluates `FOND` and its variants on different types of domain-shifts, domain-shared class distributions, and total number of classes. To this end, these experiments summarize 360 trials for each of the ten selected DG baselines, including SOTA [15, 76], on three established DG benchmark datasets – PACS [33], VLCS [16], and, OfficeHome [70]. We find that `FOND` improves domain-linked class performance when observing a high enough number of domain-shared classes. Interestingly, we observe that ERM is still a strong baseline, but `FOND` and its variants achieve a remarkable overall performance improvement of +9.3 over ERM on average (26.9% improvement), with a gain of 39.2% on VLCS. We also compare the domain-shared performance with SOTA, showing that domain-linked performance can be significantly boosted with a modest domain-shared class performance trade-off. Finally, conclusions, limitations and future work are discussed in Chapter 6.

Chapter 2

Background

This chapter reviews the existing domain generalization literature relevant to this thesis, identifies gaps in current knowledge, and situates this thesis within the broader academic context to justify its necessity.

2.1 Problem Formulation

To understand the core of domain generalization in machine learning, it is essential to first define what we mean by a *domain* within the context of image analysis tasks. In simple terms, a domain represents a particular data distribution, including the input space (e.g., images, text) and the corresponding output labels.

Definition 2.1.1 (Domain)**.** *Let $\mathcal{X}$ denote a nonempty input space (e.g., images, text, etc.) and $\mathcal{Y}$ an output label space. We denote as specific domain as $S = \{(\mathbf{x}_j, y_j)\}_{j=1}^{n} \sim \mathcal{D}^S : \mathcal{X}^S \times \mathcal{Y}^S$, where $\mathbf{x} \in \mathcal{X}^S \subseteq \mathbb{R}^d$ and $y \in \mathcal{Y}^S \subset \mathbb{Z}$.*

With the concept of a domain in place, we can now delve into the notion of domain generalization (DG). DG is a task that involves learning from multiple source domains to develop representations that generalize well to unseen target domains. This is crucial in scenarios where the model needs to perform accurately even on data distributions it has never encountered during training.

Definition 2.1.2 (Domain generalization)**.** *Assume K source (training) domains $\mathcal{S} = \{S^i \mid i = 1, ..., K\}$, where $S^i = \{(\mathbf{x}_j^i, y_j^i)\}_{j=1}^{n_i}$ denotes the i-th source domain with n_i samples,*

are available during training with the joint distributions between each pair of domains are different: $\mathcal{D}^{S^i} \neq \mathcal{D}^{S^j} : 1 \leq i \neq j \leq K$. *The goal is to learn a robust and generalizable predictive function* $h : \mathcal{X} \rightarrow \mathcal{Y}$ *from the set of* K *source (training) domains* $\mathcal{S}$ *to achieve a minimum prediction error on an out-of-distribution target-domain* $T \sim \mathcal{D}^T : \mathcal{X}^T \times \mathcal{Y}^T$ *(i.e.* $\mathcal{D}^T \neq \mathcal{D}^{S^i}$ *for* $i \in \{1, ..., K\}$*).*

This thesis evaluates methods under the *closed-set* domain generalization assumption (i.e. $\mathcal{Y}^T = \bigcup_{i=1}^{K} \mathcal{Y}^{S^i}$) where no source-domain expresses all target classes (i.e., $\mathcal{Y}^{S^i} \subset \mathcal{Y}^T$ for $i \in \{1, ..., K\}$). Furthermore, during training, there exists a set of classes expressed in only one source domain, i.e. *domain-linked* classes $\mathcal{Y}_L$.

Definition 2.1.3 (Set of domain-linked classes)**.** $\mathcal{Y}_L = \{y : |\{S^i | S^i \in \mathcal{S}, y \in \mathcal{Y}^{S^i}\}| = 1\}$.

Similarly, classes expressed in multiple domains, i.e. *domain-shared* classes $\mathcal{Y}_S$ as shown below. Here, $\mathcal{Y}^T = \mathcal{Y}_L \cup \mathcal{Y}_S$ and $\mathcal{Y}_L \cap \mathcal{Y}_S = \emptyset$.

Definition 2.1.4 (Set of domain-shared classes)**.** $\mathcal{Y}_S = \{y : |\{S^i | S^i \in \mathcal{S}, y \in \mathcal{Y}^{S^i}\}| > 1\}$.

In summary, the learning objective is to identify a generalizable predictive function $h : \mathcal{X} \rightarrow \mathcal{Y}$ to achieve a minimum predictive error on an out-of-distribution (also referred to as *unseen*) test domain T under the previously outlined conditions. In this thesis, the predictive function h is modelled as a neural network $M = (F \circ G)(\mathbf{x}) = G(F(\mathbf{x}))$ composed of a feature extraction F and classification G network.

2.2 Literature Review

The main idea behind modern domain generalization algorithms is to learn input representations that are unaffected by changes in the source (training) domain, i.e. *domain invariant representations* [4, 79]. Chapter 3 will briefly introduce and explore the theoretical motivations behind domain invariant representation learning.

2.2.1 Data Manipulation

These techniques primarily focus on data augmentation and generation techniques to assist the learning of domain invariant representations. Typical augmentation techniques include affine transformations (e.g., rotations, scaling, translations, reflections) in conjunction with

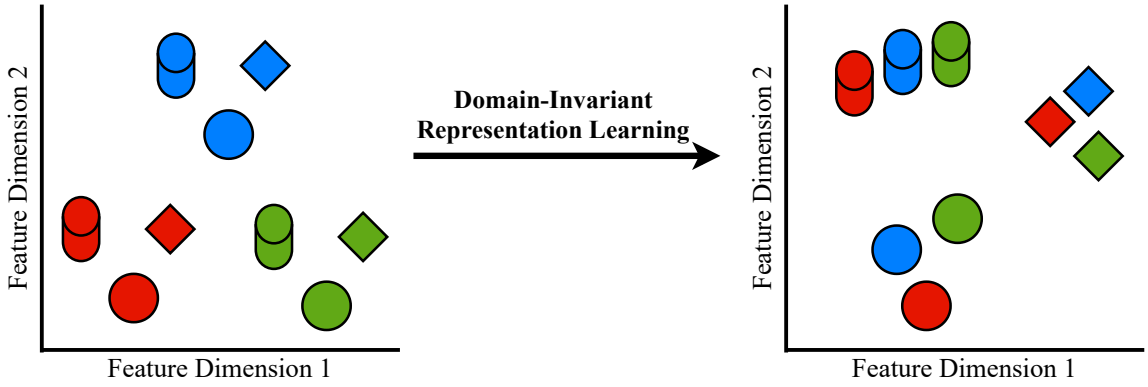


Figure 2.1: Illustrating domain-invariant (color-invariant) representation (shape) learning.

additive noise, cropping and so on [23, 61]. These basic techniques are widely employed across different machine learning applications as they enhance the generalization of a model by reducing the likelihood of over-fitting to training data. However, more complex techniques are gaining popularity, especially adversarial data augmentation. Adversarial data augmentation techniques enhance generalizability by learning to identify optimally challenging input augmentations. For example, approaches like CrossGrad [59] perturb the input along the direction of greatest domain change while changing the class label as little as possible. In a similar grain, Volpi et al. [72] and, more recently, Kim et al. [31] learn augmentations that mimic fictitious target domains to improve model generalizability. Furthermore, innovations in data generation have also positively impacted domain generalization research. Interestingly, although diffusion-based techniques [45] are growing in popularity, simple image-mixing data generation techniques such as CutMix [41], Mixup [81] and Dir-Mixup [62] remain competitive [7].

Training data is vital for machine learning since a model's generalizability is a function of the diversity of the training data available [67]. Consequently, domain generalization methods — including the technique proposed in this thesis — should be used in conjunction. Therefore, this thesis only seeks to improve the differentiable objectives that guide the learned input representations and does not evaluate data manipulation techniques. Furthermore, it is essential to consider the added complexity of leveraging these data generation techniques.

2.2.2 Multi-Domain Feature Alignment

Most existing DG approaches belong to this category. These techniques minimize the differences between learned features/representations from multiple source domains. The guiding principle is that image representations invariant to domain shifts should be more robust to any unseen target domain shift [79]. Fig. 2.1 visualizes the difference between class representation (i.e. shapes) distributions vary when they are domain-specific versus domain-invariant.

For example, domain invariance can be achieved by aligning the distribution of image representations between different domains. Consequently, various statistical distance metrics are borrowed to measure these distribution differences. For example, learning algorithms such as CORAL [63] and M^{3}SDA [46] minimize moments such as mean (1st-order moment) and variance (2nd-order moment) calculated over multiple training domains. Other methodologies use measures like Wasserstein distance or KL Divergences [74, 87]. Learning algorithms like DIRT [44] and MDA [24] learn how to transform feature distributions across multiple domains before minimizing their discrepancies.

However, these multi-source alignment strategies are not limited to the feature space. Methods like PGrad [76] and Fishr [52] successfully apply techniques on the optimization gradients themselves. Furthermore, recent research provides a causal perspective to learning domain invariant representations. These algorithms (e.g., CausIRL [13]) derive their insights from structural causal models to capture the underlying data generalization process. In addition, algorithms like EQRM [15] argue that alignment techniques that optimize for average performance provably lack robustness to out-of-distribution data [43]. Their research proposes a different probabilistic framework, which we also analyze in this thesis. Unsurprisingly, these approaches require observing multiple domain shifts for a given class to identify domain invariant representations. We demonstrate that this severely degrades the generalization performance of domain-linked classes.

Another popular approach for achieving domain-invariance is formulating the image representation learning as an adversarial game (i.e., *domain adversarial training*). Ganin et al. [18] popularized this approach with the Domain-adversarial neural network (DANN), which adversarial trained a generator and discriminator. Simply speaking, a discriminator is trained to distinguish representation source domains, while a generator seeks to fool the discriminator by learning domain-invariant image representations. Further adaptations of this methodology were made to gradually reduce the discrepancy between learned features from different domains [19, 35, 60, 77, 89]. However, since there is a 1:1 correlation between $\mathcal{Y}_L$ classes and their domains, adversarial domain discriminators may use class-discriminative features to predict the source domain. Therefore, the feature generated

would be penalized for capturing these task-relevant features.

2.2.3 Contrastive Learning

Contrastive learning is an important and prevalent feature alignment technique in the domain generalization literature. Popularized by Chen et al. [12], this technique makes pairwise comparisons between samples such that similar ones (i.e. positive pairs) are embedded close to each other while dissimilar samples (i.e. negative pairs) are not. Current contrastive techniques focus on multi-domain image comparisons to ignore spurious domain-specific features while learning task-relevant ones; see [88] and the references therein.

Motiian et al. [42] provided one of the earliest implementations of this strategy for the domain generalization task, inspiring future DG research with varying contrastive loss functions [25, 26, 55, 78]. A contribution worth noting is SelfReg [29], which adopted and popularized applying the contrastive objective of linearly interpolated representations from random pairs across domains. This improved the learned feature space's smoothness and generalization to unseen domains. However, we experimentally observe that these methodologies can still generate significant performance discrepancies between domain-shared and domain-linked classes. Therefore, Chapter 4 communicates how this discrepancy can be alleviated by focusing on specific types of positive and negative pairs.

2.2.4 Meta-Learning

The introduction of the meta-learning technique in ML literature greatly improved the ability to adapt large pre-trained models between different learning tasks [71]. The rationale for applying meta-learning to domain generalization is to emulate domain shifts during training, hoping the model can better handle domain shifts in unseen domains. These approaches divide the source domains into non-overlapping meta-train and meta-test episodes. The popular MAML [17] architecture inspires most DG meta-learning approaches, most notably the foundational MLDG [34] and ARM [82] methods that have become popular base DG architectures [62, 86]. However, meta-learners still suffer challenges when classes cannot be observed in both the meta-train and meta-test episodes. Consequently, approaches like Transfer [80] combine meta-learning with adversarial techniques to identify more generalizable representations.

2.2.5 Fairness

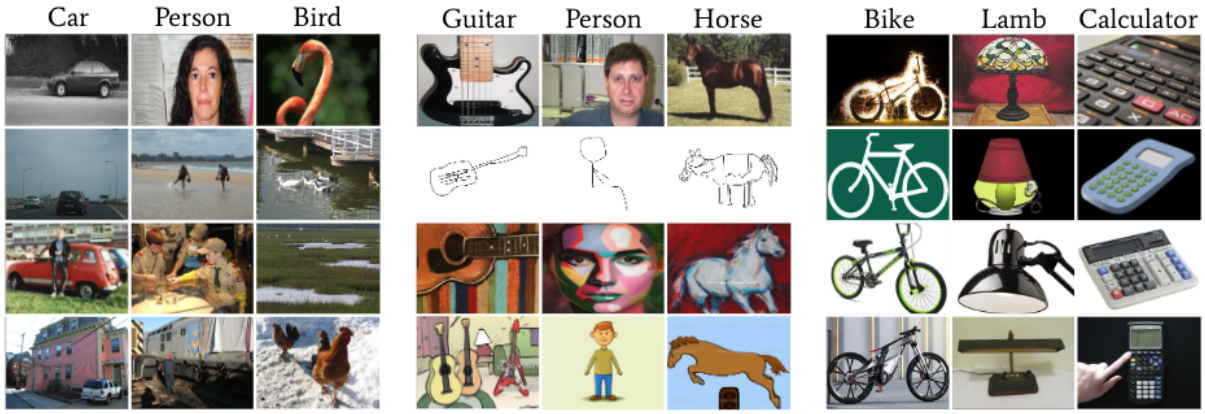
Notions of fairness in DG require equalizing appropriate statistical measures across protected attributes (e.g., race, gender) [40] to reduce performance biases. We can formulate these notions of fairness as (conditional) independence statements between random variables: prediction outcome $M(X)$, protected attribute A, and class Y [28]. For example, *demographic parity* ($M(X) \perp A$) requires the prediction outcomes to be the same across different groups; *equalized odds* ($M(X) \perp A|Y$) requires that true and false positive rates are the same across different groups; *equalized opportunity* ($M(X) \perp A|Y = y$) requires that only true positive rates are the same across different groups [47, 75]. Motivated by this research, we impose a fairness-inspired learning objective to learn generalizable representations for domain-linked classes.

2.2.6 Label Shift in Domain Generalization

The focus on domain-linked class generalization falls under the broad and recognized concept of *label-shift* in DG literature. Label shift occurs when source/target domains have different class distributions. This, however, is too broad of a categorization; therefore, published DG methodologies primarily focus on one of two types: 1) Algorithms like [13,37,80] assume distribution differences in training domains while maintaining that all classes are present in each source (training) domain; 2) Algorithms like [9, 36] focus on identifying unknown (novel) target-domain classes, i.e., *open-set generalization*. This thesis uniquely focuses on improving the performances of classes present in only one domain while maintaining that all target classes are observed during training. Furthermore, all the baselines were selected to cover a variety of DG methodologies that expect (implicitly and explicitly) differences in label distributions across domains.

2.3 Datasets

Formalized by Definition 2.1.2, the DG task requires using multiple training domains to learn a robust and generalizable predictive function to achieve a minimum prediction error on an unseen (i.e., out-of-distribution) target domain. Consequently, several image classification datasets featuring different distribution shifts are available. Among them, PACS, VLCS, and Office-Home are the three most popular image-classification datasets [73], each featuring four different domain shifts across their respective set of classes; refer to Figure 2.2 for samples. **PACS** [33] is a 9,991-image dataset consisting of four domains corresponding



(a) VLCS (b) PACS (c) OfficeHome

Figure 2.2: **Visualizing the domain shifts for each evaluation dataset.** Note how PACS and OfficeHome exhibit more obvious domain variations than VLCS.

to four different image styles: photo (P), art (A), cartoon (C) and sketch (S). The four domains hold seven object categories: dog, elephant, giraffe, guitar, horse, house and person. **VLCS** [16] is a 10,729-image dataset consisting of four domains corresponding to four different datasets: VOC2007 (V), LabelMe (L), Caltech101 (C) and SUN09 (S). Each of the four domains hold five object categories: bird, car, chair, dog and person. **OfficeHome** [70] is a 15,588-image dataset consisting of images of everyday objects organized into four domains: art-painting, clip-art, images without backgrounds and real-world photos. These domains hold 65 object categories typically found in offices and homes. Subsection 5.1.1 goes further into detail about the properties of these datasets and their importance when analyzing the generalization strength of different learning approaches for domain-linked classes

2.4 Motivation

Existing approaches presuppose all classes are expressed in multiple domains or seek to maximize average generalization accuracy. However, these methodologies demonstrate large performance discrepancies between domain-linked and domain-shared classes. Since these domain-linked classes are of interest in real-world settings, there is a need to understand the factors that impact their performance and build models that can improve their generalizability to unseen domains.

Chapter 3

Theoretical Analysis

This chapter theoretically grounds the challenge of minimizing the generalization error bounds of domain-linked classes compared to domain-shared ones. Chapter 4 then applies these insights to develop the proposed algorithm; `FOND`. Since domain adaptation (DA) is closely related to domain generalization (DG), we begin our error bounds analysis there in Section 3.1. Section 3.2 communicates its extension to a multi-source setting, and Section 3.3 focuses on the key terms. Then Section 3.4 motivates the notion of domain-invariant learning. Last, with the groundwork laid by the previous sections, Section 3.5 provides the intuitions on what makes domain-linked classes generalization hard.

3.1 Reviewing Domain Adaptation Error Bounds

The main idea behind modern domain generalization algorithms is to learn consistent input representations across source (training) domains for a given class. To improve understanding of this principle, it is crucial to delve into the generalization bounds established by Ben-David et al. [3] on domain adaptation before applying it to DG. Note that unlike for domain *generalization*, domain *adaptation* assumes access to unlabeled data from the target (test) domain during the machine learning model training stage.

Given a predictor h from a hypothesis class $\mathcal{H}$, the research by Ben-David et al. [3] focuses on bounding the target domain error $\epsilon_T(h)$ using the tractable source (training) domain error $\epsilon_S(h)$, since we do not have access to the true target domain labels. Ben-David et al. [3] demonstrates that $\epsilon_T(h)$ can be bounded by four terms which are summarized by Theorem 3.1.1. First is the expected source domain error $\epsilon_S(h)$. Second is a measure of the

discrepancy between the source and target distributions. To this end, Ben-David et al. [3] proposes a new divergence metric $d_{\mathcal{H}\Delta\mathcal{H}}(\cdot,\cdot)$ which they demonstrate can be estimated from finite samples. Third, the error is bounded by the complexity (expressiveness) of the hypothesis space $\mathcal{H}$, i.e., the VC dimension d [66], and the number of *i.i.d.* training samples m. Last is the ideal joint risk $\lambda = \min_{h\in\mathcal{H}} \{\epsilon_T(h) - \epsilon_S(h)\}$ communicating the lowest possible difference a learning algorithm can attain between target and source domains.

Theorem 3.1.1 ([3] Thm. 2)**.** *Let $\mathcal{H}$ be a hypothesis space with a VC dimension of d. If $\mathcal{U}^S, \mathcal{U}^T$ are unlabeled samples of size m each drawn form $\mathcal{D}^S$ and $\mathcal{D}^T$ respectively, then for any $\delta \in (0,1)$, with probability of at least $(1-\delta)$ (over the choice of samples), for every $h \in \mathcal{H}$, the target domain error $\epsilon_T(h)$ is bounded by:*

$$\epsilon_T(h) \leq \epsilon_S(h) + \frac{1}{2}\hat{d}_{\mathcal{H}\Delta\mathcal{H}}(\mathcal{U}^S,\mathcal{U}^T) + 4\sqrt{\frac{2d\log(2m) - \log(\frac{2}{\delta})}{m}}) + \lambda \tag{3.1}$$

where $\hat{d}_{\mathcal{H}\Delta\mathcal{H}}(\mathcal{U}_S,\mathcal{U}_T)$ is the estimate of $d_{\mathcal{H}\Delta\mathcal{H}}(\mathcal{D}^S,\mathcal{D}^T)$ on the two sets of finite data samples.

The benefit of this error-bound is that it makes no assumptions about the relationships between the source and target domains. This is because, under domain adaptation (DA) settings, unlabelled target data is available to estimate the $\mathcal{H}\Delta\mathcal{H}$ divergence.

3.2 Extension to Multiple Source Domains

In the domain generalization setting, however, using multiple source domains during training is common practice. Consequently, we now seek to bound the target domain error ϵ_T of a predictor $\hat{h} \in \mathcal{H}$ that empirically minimizes a convex combination of the K source domain errors, $\epsilon_\alpha(h)$ defined as follows,

$$\epsilon_\alpha(h) = \sum_{i=1}^{K} \alpha_i \epsilon_i(h) \tag{3.2}$$

$$\hat{h} = \arg\min_{h\in\mathcal{H}} \epsilon_\alpha(h) \tag{3.3}$$

given a domain weight vector $\alpha = (\alpha_1, ..., \alpha_K)$ where $\sum_{i=1}^{K} \alpha_i = 1$.

Now, similar to Theorem 3.1.1 in Section 3.1, Ben-David et al. [3] bounds the target domain error of this predictor $\epsilon_T(\hat{h})$, in Theorem. 3.2.1, with respect to four terms: 1) The best target predictor $h_T^\star$; 2) The $\mathcal{H}\Delta\mathcal{H}$ divergence between the target and individual source domains; 3) The number of training samples m and the complexity d of the hypothesis space; 4) The ideal joint risk between the target and individual source domains.

Theorem 3.2.1 ([3] Thm. 4)**.** *Let $\mathcal{H}$ be a hypothesis space of VC dimension d. For each $i \in \{1, ..., K\}$, let S^i be a labeled sample of size $\beta_i m$ generated by drawing $\beta_i m$ points from $\mathcal{D}^{S^i}$. If $\hat{h} \in \mathcal{H}$ is the empirical minimizer of $\epsilon_\alpha(h)$ for a fixed weight vector α on these samples and $h_T^\star = \min_{h\in\mathcal{H}} \epsilon_T(h)$ is the target error minimizer, then for any $\delta \in (0, 1)$, with probability at least $1 - \delta$,*

$$\epsilon_T(\hat{h}) \leq \epsilon_T(h_T^*) + 2\sqrt{\left(\sum_{i=1}^{K} \frac{\alpha_i^2}{\beta_i}\right)\left(\frac{d\log(2m) - \log(\delta)}{2m}\right)} + \sum_{i=1}^{K} \alpha_i(2\lambda_i + \hat{d}_{\mathcal{H}\Delta\mathcal{H}}(S^i, T)) \tag{3.4}$$

where $\lambda_i = \min_{h\in\mathcal{H}}\{\epsilon_T(h) + \epsilon_i(h)\}$.

3.3 Simplifying the Error Bounds

In practice, we assume $\lambda-$closeness (i.e., $\lambda_i = 0$) because of the enormous capacities of neural networks; this assumption is often made in DG and is experimentally validated [68]. Without loss of generality, we use the Landau notation in Eq. (3.5) to simplify Eq. (3.4). Since $\epsilon_T(h_T^\star)$ is fixed, we only focus on the $d_{\mathcal{H}\Delta\mathcal{H}}$ and $O(m^{-1})$ terms to compare the target error bounds of predictors h focused on either domain-linked or shared classes.

$$\epsilon_T(\hat{h}) \leq \epsilon_T(h_T^*) + \mathcal{O}(m^{-1}) + \sum_{i=1}^{K} \alpha_i \; d_{\mathcal{H}\Delta\mathcal{H}}(S^i, T) \tag{3.5}$$

3.4 Optimizing the Fixed Source-Target Divergence

Further analysis of these error bounds is obstructed even with the simplification since the dataset's source-target domain divergence is a fixed property. Thankfully, however,

although the source-target domain divergence is fixed, one may learn a mapping function $f \in \mathcal{F}\ ;\ \mathcal{F} : \mathcal{X} \to \mathcal{Z}$ that maps the original input data $\mathbf{x} \in \mathcal{X}$ to some representation space $\mathcal{Z}$, to reduce the source-target $d_{\mathcal{H}\Delta\mathcal{H}}$ distribution divergence, i.e., $d_{\mathcal{H}\Delta\mathcal{H}}(f(S^i), f(T))$. This is the impetus of *domain-invariant representation learning* that inspires all published research in domain adaptation and generalization.

Consequently we can reformulate predictors $h \in \mathcal{H}\ ;\ \mathcal{H} : \mathcal{X} \to \mathcal{Y}$ as a composition of an encoding function $f \in \mathcal{F}\ ;\ \mathcal{F} : \mathcal{X} \to \mathcal{Z}$ and a decoding function $g \in \mathcal{G}\ ;\ \mathcal{G} : \mathcal{Z} \to \mathcal{Y}$, i.e., $h = g \circ f$. Therefore, we now have a way to improve the target domain error bound ϵ_T by empirically minimizing a risk $\mathcal{R}$ defined as a sum of the convex combination of the K source domain errors ϵ_α, Eq. (3.2), and source-target distribution divergences $d_\alpha(h)$. We define this risk-minimizing predictor as $\hat{h}$ where $\hat{f}$ is the corresponding encoder as follows,

$$d_\alpha(h) = \sum_{i=1}^{K} \alpha_i d_{\mathcal{H}\Delta\mathcal{H}}(f(S^i), f(T)) \tag{3.6}$$

$$\mathcal{R}(h) = \epsilon_\alpha(h) + d_\alpha(h) \tag{3.7}$$

$$\hat{h} = \arg\min_{h \in \mathcal{H}} \mathcal{R}(h) \tag{3.8}$$

3.5 Comparing Domain-Linked and Domain-Shared Generalization Error Bounds

We now use the definitions and error bounds introduced in the previous sections to gain insights into the consistent experimental performance differences between domain-linked and domain-shared classes. Suppose we have a set of K source domains S^i for $i \in [1, K]$, that each contributes $\alpha_i \cdot m$ samples (noting $\sum_{i=1}^{K} \alpha_i = 1$), with m denoting the total dataset size for a particular class. The domain-linked setting could simply be represented as a single source S^i generalization task with the resulting target domain error bound,

$$\epsilon_T(\hat{h}_l) \leq \epsilon_T(h_T^*) + \mathcal{O}(\alpha_i^{-1} m^{-1}) + d_{\mathcal{H}\Delta\mathcal{H}}(\hat{f}_l(S^i), \hat{f}_l(T)) \tag{3.9}$$

where $\hat{h}_l$ is the single source empirical risk $\mathcal{R}_l$ minimizer,

$$\mathcal{R}_l(h) = \epsilon_i(h) + d_{\mathcal{H}\Delta\mathcal{H}}(f(S^i), f(T)) \tag{3.10}$$

$$\hat{h}_l = \arg\min_{h \in \mathcal{H}} \mathcal{R}_l(h) \tag{3.11}$$

with the corresponding mapping function $\hat{f}_l$. Similarly, the domain-shared setting would be represented as a K source generalization task with the target domain error bound,

$$\epsilon_T(\hat{h}_s) \leq \epsilon_T(h^*_T) + \mathcal{O}(m^{-1}) + \sum_{i=1}^{K} \alpha_i \; d_{\mathcal{H}\Delta\mathcal{H}}(\hat{f}_s(S^i), \hat{f}_s(T)) \tag{3.12}$$

where $\hat{h}_s$ is the K source empirical risk $\mathcal{R}$, Eq. (3.7).

3.5.1 Domain-Shared Classes Typically Have More Samples

The domain-linked, Eq. (3.9), and domain-shared, Eq. 3.12, error bounds communicate a dependence on the number of available samples. Given a dataset of m samples for a particular class, a class from a single source domain S^i contributes $\alpha_i \cdot m$ samples, i.e., $\mathcal{O}(\alpha_i^{-1} m^{-1})$. The inclusion of additional domains, making the class domain-shared, would have $\geq \alpha_i \cdot m$ samples, resulting in a lower error bound, i.e., $\mathcal{O}(m^{-1}) \leq \mathcal{O}(\alpha_i^{-1} m^{-1})$.

3.5.2 Domain-Shared Classes Learn Domain-Invariant Features

Last we compare the source-target $d_{\mathcal{H}\Delta\mathcal{H}}$ distribution divergence terms between the domain-linked $d_{\mathcal{H}\Delta\mathcal{H}}(f_l(S^i), f_l(T))$ and domain-shared $\sum_{i=1}^{K} \alpha_i \; d_{\mathcal{H}\Delta\mathcal{H}}(f_s(S^i), f_s(T))$ settings. We are primarily interested in comparing the ability of $\hat{f}_l$ and $\hat{f}_s$, from $\hat{h}_l$ and $\hat{h}_s$ respectively, to encode the source S^i and target T samples close together.

Unfortunately, we cannot identify empirical risk ($\mathcal{R}$ and $\mathcal{R}_l$) minimizing predictors $\hat{h}$, and corresponding decoders $\hat{f}$, since unlike for DA, the DG task does not access the target domain samples required to estimate the source-target distribution divergences. Therefore, published DG research operates on the experimentally validated assumption that an encoding function f that reduces the representation discrepancy of samples from multiple source

domains is more likely to reduce the source-target representation discrepancy [1, 80]. Consequently, we can justifiably use source-source $d_{\mathcal{H}\Delta\mathcal{H}}$ distribution divergence as a surrogate and redefine the empirical risk as,

$$\tilde{\mathcal{R}}(h) = \epsilon_\alpha(h) + \sum_{i=1}^{K} \alpha_i \sum_{j=1}^{K} d_{\mathcal{H}\Delta\mathcal{H}}(f(S^i), f(S^j)) \tag{3.13}$$

Observe that this surrogate is not helpful for the domain-linked (single source) setting since any predictor h with encoder f would minimize $d_{\mathcal{H}\Delta\mathcal{H}}(f(S^i), f(S^j))$ for $i = j$. As a result, the domain-shared encoder $\hat{f}_s$ is expected to learn more domain-invariant mappings through the joint minimization of the source-domain error ϵ_α and divergences d_α. In contrast, the domain-linked encoder f_l is only selected by minimizing the source-domain error. Therefore, it is justifiable to expect that the domain-linked source-target divergence would be greater than the weighted sum of domain-shared source-target divergences,

$$d_{\mathcal{H}\Delta\mathcal{H}}(\hat{f}_l(S^i), \hat{f}_l(T)) \geq \sum_{i=1}^{K} \alpha_i \; d_{\mathcal{H}\Delta\mathcal{H}}(\hat{f}_s(S^i), \hat{f}_s(T)) \tag{3.14}$$

These intuitions align with theoretical understandings about the importance of diversifying training data by collecting/generating more of it and augmenting what is already there [67]. Furthermore, the empirical results corroborate this theoretical intuition. Namely, increasing the number of available source domains improves the domain-invariance of learned representations, leading to more reliable target domain generalizations.

Chapter 4

Fair Contrastive Learning for Domain-Linked Class Generalization

The goal of a domain generalization learning algorithm is twofold; the learner aims to accurately predict the target class Y and also tries to be insensitive to task-irrelevant domain-specific variations (e.g., image styles in the PACS dataset), which we will denote as A. In other words, we are interested in learning a feature encoder (e.g., neural network) $F(X) = Z$ that contains as much information as possible about the target Y while at the same time filtering out domain-specific information A, i.e., *domain invariant representations* [8, 85]. We can represent this learning objective as an information bottleneck [64] expressed by Eq. (4.2) where $I(\cdot, \cdot)$ denotes the mutual information between a pair of random variables and λ a scaling factor.

$$I(Z, Y) = \sum_{z \in \mathcal{Z}} \sum_{y \in \mathcal{Y}} P_{(Z,Y)}(z, y) \log \left(\frac{P_{(Z,Y)}(z, y)}{P_{(Z)}(z) P_{(Y)}(y)} \right) \tag{4.1}$$

$$\max_{Z} \quad I(Z, Y) - \lambda I(Z, A) \tag{4.2}$$

4.1 Motivating Contrastive Learning

A naive classification loss (i.e., categorical cross-entropy) is primarily focused on maximizing the first term in Eq. (4.2), $I(Z, Y)$, namely making the learned representations Z

of an input X maximally informative of its corresponding label Y. However, since the domain generalization objective does not make the *i.i.d.* assumption, Z will likely encode irrelevant domain-specific correlations A, resulting in a model with observable poor generalization abilities [73, 88]. Consequently, pairwise (e.g., image representation to image representation) contrastive learning is an incredibly prevalent additional learning objective to minimize $I(Z, A)$. Contrastive learning regulates the learned representations Z by asserting that representations from the same class should be embedded close together while separating those from different classes. Illustrated by Figure 4.1, this improves the differentiability between classes.

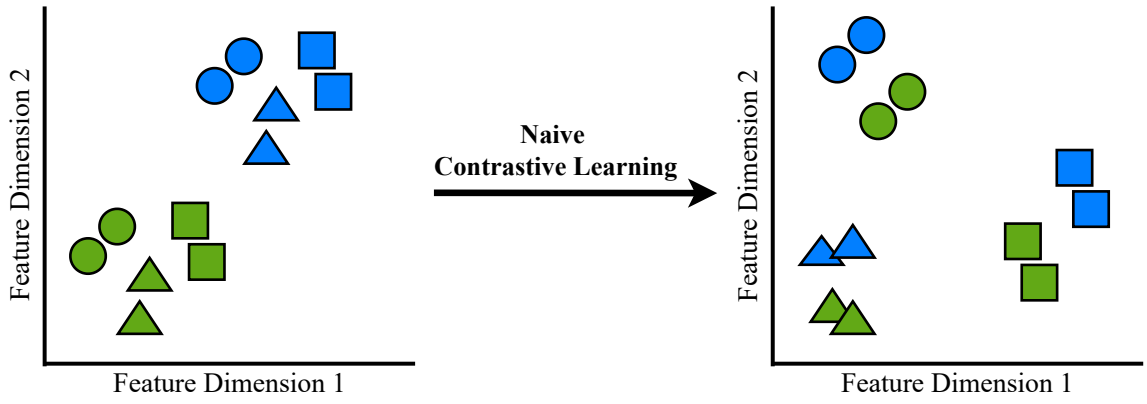


Figure 4.1: Illustrating the effects of contrastive learning on learned representations.

4.2 Method

4.2.1 Focusing on Specific Pairwise Relationships

Unfortunately, assuming that we can arrive at a representation Z that preserves all the relevant information about Y while removing all domain-specific information is unrealistic. Referencing Figure 4.1, although the boundaries between classes (shapes) are more distinguishable, they are still clustered by the domain (colours). We also observed this when plotting the learned representations of real-world datasets in Subsection 5.2.4, especially when only optimizing the classification loss (i.e. the ERM baseline).

In alignment with the analysis provided in Chapter 3, observing domain variances allows models to learn representations that reduce the average distribution divergence between

learned source and target domain data representations. Since algorithms do not observe enough domain variances in $\mathcal{Y}_L$ data comparison to $\mathcal{Y}_S$ data, their classification accuracies are much worse; refer to Figure 1.2. Therefore, unlike current contrastive learning approaches, we must treat $\mathcal{Y}_L$ and $\mathcal{Y}_S$ classes differently.

Maximizing the mutual information between positive (same-class) inter-domain samples guides domain-invariant learning [11, 53]. For example, in Figure 1.1, an algorithm observing pairwise relationships between samples from domain-shared $\mathcal{Y}_S$ classes may observe that encoding edge features increases mutual information while color reduces it. Furthermore, due to the success of *hard negative mining* in representation learning literature [38, 54, 84], we hypothesize that negative (different-class) intra-domain comparisons are more informative than negative inter-domain comparisons for reducing spurious domain and class correlations. For example, in Figure 1.1, an algorithm may achieve color invariance by minimizing mutual information between samples from different classes (shapes) but from the same domain (color). Additionally, you can refer to Figure 4.2 for a visual demonstration of the difference between inter-domain positive and intra-domain negative pairs using the PACS evaluation dataset.

Therefore, we define a feature extractor $F : \mathcal{X} \to \mathcal{H}$ to take an input samples $\mathbf{x} \in \mathcal{X}$ and generate representation vectors, $\mathbf{h} \in \mathcal{H} \subseteq \mathbb{R}^{d_F}$. We regularize the representation vectors by applying a contrastive objective to the output of a projection network $Q : \mathcal{H} \to \mathcal{Z}$ that generates normalized, lower-dimensional representations $\mathbf{z} \in \mathcal{Z} \subseteq \mathbb{R}^{d_Q}$. The goal of the contrastive objective defined by Eq. (4.3) is to maximize the cosine similarity of projected representations $\mathbf{z}$ between samples from the same class and minimize those that are not.

$$\mathcal{L}_{xdom} = \sum_{i \in I} \frac{-1}{|P(i)|} \sum_{p \in P(i)} \log \frac{\alpha(\mathbf{z}_i, \mathbf{z}_p) \cdot \exp(\mathbf{z}_i \cdot \mathbf{z}_p / \tau)}{\sum_{j \in I \setminus \{i\}} \beta(\mathbf{z}_i, \mathbf{z}_j) \cdot \exp(\mathbf{z}_i \cdot \mathbf{z}_j / \tau)} \tag{4.3}$$

$$\alpha(\mathbf{z}_i, \mathbf{z}_p) = \begin{cases} a, & \text{if } S(\mathbf{z}_i) \neq S(\mathbf{z}_p), \text{ where } a \geq 1 \\ 1, & \text{otherwise} \end{cases} \tag{4.4}$$

$$\beta(\mathbf{z}_i, \mathbf{z}_j) = \begin{cases} b, & \text{if } S(\mathbf{z}_i) = S(\mathbf{z}_j), y_i \neq y_j, \text{ where } b \geq 1 \\ 1, & \text{otherwise} \end{cases} \tag{4.5}$$

Let $i \in I \equiv \{1...N\}$ be the index of a sample (denoted as the *anchor*) where N denotes the batch size. $P(i) = \{p \in I \setminus \{i\} : y_p = y_i\}$ is the set of indices of all positives in the batch. The α function increases the cosine similarity weight of the anchor $\mathbf{z}_i$ and positive

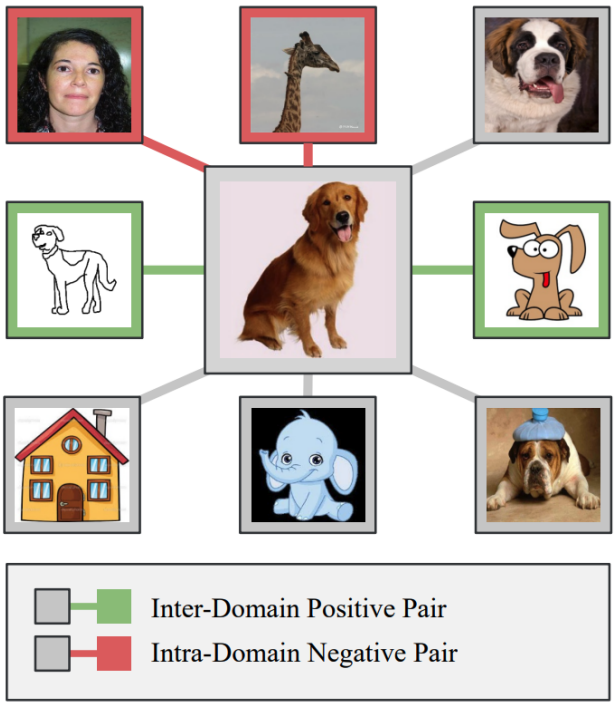


Figure 4.2: **Illustrating contrastive pair types using the PACS dataset.** Given a picture-style (domain) image of a dog (class), the red connections denote intra-domain negative (different class) pairs. In contrast, the green connections denote inter-domain positive (same class) pairs. Image classes in the PACS dataset are split between four domains: picture, art, cartoon and sketches; refer to Section 2.3 for more information on the evaluation datasets.

$\mathbf{z}_p$ sample if they are inter-domain (i.e., $S(\mathbf{z}_i) \neq S(\mathbf{z}_p)$) pairs; refer to Eq. (4.4). Note that $S(\mathbf{z}_i)$ denotes the domain $S \in \mathcal{S}$ that $\mathbf{z}_i$ belongs to. Additionally, the β function increases the cosine-similarity weight of the anchor $\mathbf{z}_i$ and $\mathbf{z}_j$ if they are negative, intra-domain (i.e., $S(\mathbf{z}_i) = S(\mathbf{z}_j)$) pairs; refer to Eq. (4.5).

4.2.2 Transferring Domain-Invariant Representations

Increasing the weight of positive inter-domain similarity metrics through α in Eq. (4.3) biases the model towards domain-shared $\mathcal{Y}_S$ generalization since these metrics are not present between domain-linked $\mathcal{Y}_L$ class samples. Therefore, we draw inspiration from notions of *fair* learning in DG literature to learn generalizable features for $\mathcal{Y}_L$ classes.

Notions of fair learning in DG literature aim to make predictive outcome $\hat{Y}$ of a model $M(X)$ independent of protected attributes A (e.g., gender, race, age) present in a dataset [40,47,75]. Directly applying this notion of fair learning by defining the protected attribute A as whether a sample belongs to a domain-shared or domain-linked class would make the prediction outcome $M(X) = \hat{Y}$ entirely dependent on A.

To overcome these challenges, we constrain the *error rates* across domain-shared and domain-linked classes. The objective is to enforce that the domain-invariant representations learned from domain-shared $\mathcal{Y}_S$ classes *also* improve the generalizability of the disadvantaged domain-linked $\mathcal{Y}_L$ classes. The violation of this objective is measured by Eq. (4.6). We observe in Chapter 5 that this leads to significant improvements in $\mathcal{Y}_L$ classes, given the model observes a sufficient number of $\mathcal{Y}_S$ classes; this is denoted as the *High* shared-class experimental setting defined in Subsection 5.1.2. Since the task is image classification, the corresponding task loss, $\mathcal{L}_{task}$, is categorical cross-entropy.

$$\mathcal{L}_{disc} = |\mathcal{L}_{task}^{\mathcal{Y}_L} - \mathcal{L}_{task}^{\mathcal{Y}_S}| \tag{4.6}$$

Algorithm 1 `FOND` Training Algorithm

Require: Source datasets $\mathcal{S}$; Feature extractor F; Projection network Q; Classification network G; Positive pair weight α; Negative pair weight β; Loss regularizers λ_{xdom} and λ_{disc}

while Not Converged **do**

$\mathcal{B} = \{(\mathbf{x}^1, \mathbf{y}^1), (\mathbf{x}^2, \mathbf{y}^2), ..., (\mathbf{x}^K, \mathbf{y}^K)\}$ ▷ Sample a batch from K source domains $\mathcal{S}$

$\mathcal{B}_F = \{(\mathbf{h}^1, \mathbf{y}^1), (\mathbf{h}^2, \mathbf{y}^2), ..., (\mathbf{h}^K, \mathbf{y}^K)\} \leftarrow F(\mathcal{B})$ ▷ Generate the feature vectors

$\mathcal{B}_Q = \{(\mathbf{z}^1, \mathbf{y}^1), (\mathbf{z}^2, \mathbf{y}^2), ..., (\mathbf{z}^K, \mathbf{y}^K)\} \leftarrow Q(\mathcal{B}_F)$ ▷ Generate the feature projections

$\mathcal{L}_{xdom} \leftarrow \{\mathcal{B}_Q, \alpha, \beta\}$ ▷ Calculate the contrastive loss; Eq. (4.3)

$\mathcal{B}_C \leftarrow G(\mathcal{B}_F)$ ▷ Generate the classification logits

$\mathcal{B}_C^{(L)}, \mathcal{B}_C^{(S)} \leftarrow \{\mathcal{B}_C\}$ ▷ Separate logits based on ground-truth label group, i.e. $\mathcal{Y}_L$ or $\mathcal{Y}_S$

$\mathcal{L}_{fair} \leftarrow \{\mathcal{B}_C^{(L)}, \mathcal{B}_C^{(S)}\}$ ▷ Calculate the $\mathcal{Y}_L$ vs $\mathcal{Y}_S$ discrepancy loss; Eq. (4.6)

$\mathcal{L}_{task} \leftarrow \{\mathcal{B}_C\}$ ▷ Calculate the classification loss

$\mathcal{L}_{\texttt{FOND}} = \mathcal{L}_{task} + \lambda_{xdom} \cdot \mathcal{L}_{xdom} + \lambda_{disc} \cdot \mathcal{L}_{disc}$ ▷ Calculate the overall loss; Eq. (4.7)

end while

return F, G

4.2.3 Overall Learning Objective

We refer to the combination of these learning objectives as `FOND`, i.e., a fairness-inspired contrastive learning objective for domain-linked class generalization; we summarize the

overall learning objective as follows:

$$\mathcal{L}_{\texttt{FOND}} = \mathcal{L}_{task} + \lambda_{xdom} \cdot \mathcal{L}_{xdom} + \lambda_{disc} \cdot \mathcal{L}_{disc}. \tag{4.7}$$

To reiterate, we learn domain-invariant representations by focusing on specific pairwise sample relationships through the contrastive loss $\mathcal{L}_{xdom}$ defined by Eq. (4.3). We then require these representations to improve domain-linked class generalizability by imposing a fairness-inspired performance discrepancy loss between $\mathcal{Y}_L$ and $\mathcal{Y}_S$ classes through the $\mathcal{L}_{disc}$ loss defined by Eq. (4.6). Furthermore, Algorithm 1 provides the steps performed for each mini-batch update, Section 4.4 outlines the model architecture and Subsection 5.1.4 outlines additional implementation details with code available at https://github.com/criticalml-uw/fond.

4.3 Variations

Based on the values assigned for a and b in the α (Eq. (4.4)) and β (Eq. (4.5)) functions,FOND can be represented in multiple ways. This thesis focuses on the variants presented in Table. 4.1 with evaluations provided in Table. 5.2 and Section 5.2.5.

Table 4.1: Variations of the FOND algorithm.

	Components			
Variant	$\mathcal{L}_{xdom}$	α	β	$\mathcal{L}_{disc}$
FOND\FBA	✓	-	-	-
FOND\FB	✓	✓	-	-
FOND\F	✓	✓	✓	-
FOND	✓	✓	✓	✓

4.4 Model Architecture

Referring to Figure 4.3 we now explain each module of FOND.

The *Feature Extraction Network*, $F(.)$, takes a training input sample $\mathbf{x} \in \mathcal{S}$ and generates a representation vector, $\mathbf{h} = F(\mathbf{x}) \in \mathbb{R}^{d_F}$ where $d_F = 512$. We used the ResNet-18

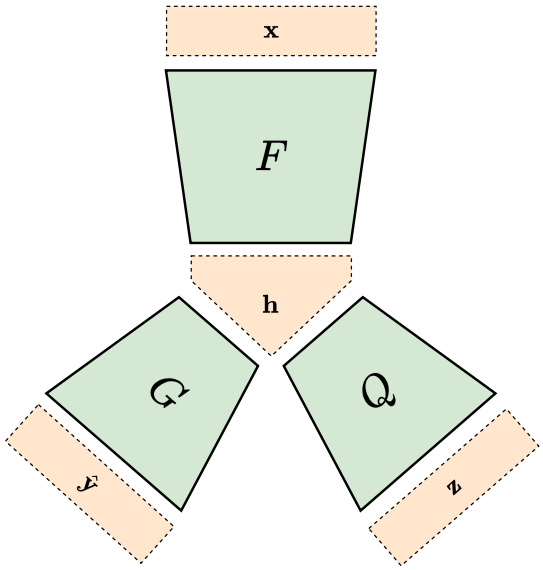


Figure 4.3: **Visualizing the FOND model architecture.** For an input, $\mathbf{x}$, the feature extractor $F(.)$ generates a representation vector, $\mathbf{h}$. The network $G(.)$ performs the downstream classification task, and $Q(.)$ generates the low-dimension representation $\mathbf{z}$ used for the contrastive objective, Eq. (4.3).

architecture [23] pre-trained on ImageNet [14] common to most image recognition DG methods. Additionally, the hidden representations are normalized, $||\mathbf{h}|| = 1$, since this has experimentally been shown to improve performance [57].

The *Projection Network*, $Q(.)$, takes the representation vector $\mathbf{h}$ and non-linearly projects it to a lower-dimensional vector $\mathbf{z} = P(\mathbf{h}) \in \mathbb{R}^{d_Q}$ where $d_Q = 256$. Additionally, the projection vector is normalized $||\mathbf{z}|| = 1$. These projections are used for FOND's contrastive learning objective Eq. (4.3). Previous works [12, 27] validate separating the representations $\mathbf{h}$ used for downstream tasks and those for contrastive objectives $\mathbf{z}$. The network is implemented by a 2-layer MLP with a ReLU activation function between the layers.

The *Classification Network*, $G(.)$, performs the image classification downstream task with the representations generated by $F(.)$, i.e., $\mathbf{h} \in \mathbb{R}^{d_F}$. The network's output is a vector of dimension $|\mathcal{Y}_T|$ denoting the softmax label probabilities of the input $\mathbf{x}$. The network is implemented by a 2-layer MLP with a ReLU activation function between the layers.

Chapter 5

Experiments

The domain-linked and domain-shared generalization strength of FOND, its variants and baselines are primarily evaluated by varying three important factors. First, we vary the amount of domain-linked/shared classes present in a dataset by creating *High* and *Low* shared-class settings for each dataset as detailed in Subsection 5.1.2. Second, we perform our evaluations across different types of inter-domain shifts. This was primarily examined by comparing the VLCS dataset with PACS since they are similar in size. While PACS has larger and more distinguishable domain shifts, VLCS has more subtle real-world shifts. Last, we observe how different sizes of target classes impact generalization. This was examined by comparing PACS and OfficeHome since they are of similar style-based domain variances and sizes, but OfficeHome has ∼10x more classes and around ∼7x less samples per class. Subsection 5.1.1 outlines more details on these datasets.

For rigorous, fair and reproducible evaluation, we mirrored the DomainBed [21] test-bed to be consistent with DG literature. We report class-averaged classification accuracy and standard errors for $\mathcal{Y}_L$ classes. Reported values arise from three Monte-Carlo runs for three datasets across four domains, five random hyper-parameter selections, and the *Low* and *High*-shared settings. Overall, the extensive experiments summarize the results of 360 experiments for each of the 11 methodologies and 3 FOND variants (i.e. 5040 trials).

5.1 Setup

5.1.1 Datasets

To evaluate $\mathcal{Y}_L$ performance with respect to inter-domain variations and the number of classes, we required keeping consistent a) dataset sizes and b) number of domains. Therefore, introduced in Section 2.3, we chose DG literature gold standard datasets [88]: PACS, VLCS, and OfficeHome, as shown in Figure 2.2. While PACS [33] (Figure 2.2b) and OfficeHome [70] (Figure 2.2c) datasets share similar style-based domain-variations, there is a ∼10x difference in class size. Furthermore, although VLCS [16] (Figure 2.2a) and PACS have similar class sizes, VLCS expresses nuanced real-world domain variations. Table 5.1 summarizes the differences in the evaluation datasets.

5.1.2 Defining Shared-Class Distribution Settings

We define two shared-class distribution settings – *Low* and *High* – denoting the relative number of shared classes $|\mathcal{Y}_S|$ with respect to the total $|\mathcal{Y}_T|$. Table 5.1 communicates the different properties of the datasets and the number of $\mathcal{Y}_L$ versus $\mathcal{Y}_S$ classes for the *Low* and *High* shared-class distribution experimental settings. In the *Low* setting $\sim 1/3$ of the classes are domain-shared; $\sim 2/3$ in the *High* setting. $\mathcal{Y}_L$ and $\mathcal{Y}_S$ classes were randomly selected and assigned round-robin to each source-domain. Domain-linked classes only exist in one training (source) domain. Domain-shared classes do not exist in all training domains; they are randomly placed in only two of the three training domains.

Table 5.1: **Evaluation dataset properties and class distribution settings.** This table communicates the defining characteristics of each evaluation dataset introduced in Subsection 5.1.1 and the number of domain-linked/shared classes present in the *High* and *Low* class distribution settings introduced in Subsection 5.1.2.

	Properties				Settings ($\lvert\mathcal{Y}_S\rvert : \lvert\mathcal{Y}_L\rvert$)	
Datasets	**# Domains**	**# Classes**	**# Samples**	**Domain Shift Type**	**Low-shared**	**High-shared**
PACS	4	7	9,991	Style-Based	3:4	5:2
VLCS	4	5	10,729	Real-World	2:3	4:1
OfficeHome	4	65	15,588	Style-Based	25:40	50:15

5.1.3 Baselines

Baselines were selected to a) cover a variety of foundational DG methodologies that are b) well represented in DG literature and have c) been benchmarked against DomainBed [21]. Therefore, we evaluate: naive empirical risk minimization, `ERM`; popular distribution-alignment, `CORAL` [63]; contrastive mixing, `RSC` [25] predecessor, `SelfReg` [29]; contrastive CLIP-based [50] method, `CAD` [55]; popular meta-learning baselines, `ARM` [82] and `MLDG` [34]; adversarial meta-learning network, `Transfer` [80]; causal representations, `CausIRL` [13]; gradient optimization, `PGrad` [76]; probabilistic framework, `EQRM` [15].

5.1.4 Implementation

For consistency, all algorithms have a fine-tuned ResNet-18 backbone [23] pre-trained on ImageNet [14]. Specifically, we replace the final (softmax) layer, insert a dropout layer and then fine-tune the entire network. Since minibatches from different domains follow different distributions, batch normalization degrades domain generalization algorithms [58]. Therefore, we freeze all batch normalization layers before fine-tuning them. The training data augmentations include random size crops and aspect ratios, resizing to 224 $\times$ 224 pixels, random horizontal flips, random color jitter, and random gray scaling. The experiments ran on different GPUs: NVIDIA RTXA6000 and NVIDIA GeForce RTX2080.

5.1.5 Hyper-Parameter Search

We perform five random search attempts for each algorithm over a joint distribution of all their hyper-parameters. This is repeated for each of the five sets of hyper-parameters, and the set maximizing the average domain-linked $\mathcal{Y}_L$ accuracy is selected. This search is performed across three different seeds where all hyper-parameters are optimized anew for each algorithm, dataset and partial-overlap setting. The hyper-parameter search space for each algorithm is provided in the attached code.

5.1.6 Model Selection

Given K domains, we train K models, sharing the same hyper-parameters θ. Each model holds a different domain. During the training of each model, 80% of the training domain data is used for training, and the other 20% is set aside as part of the validation set for model selection. We evaluate each model on 100% of the held-out (target) domain

data and average the $\mathcal{Y}_L$ accuracy of these K models over their held-out domains. This gives us an estimate of the quality of a given set of hyper-parameters. This strategy was chosen because it aligns with the goal of maximizing expected performance under out-of-distribution domain variations without picking the model using the out-of-distribution data. The $\mathcal{Y}_L$ accuracy performance across held-out domains and final averages for each dataset, algorithm and shared-class setting are displayed in Table 5.2.

5.2 Results

5.2.1 Impact of Shared-Class Settings and Total Class Size

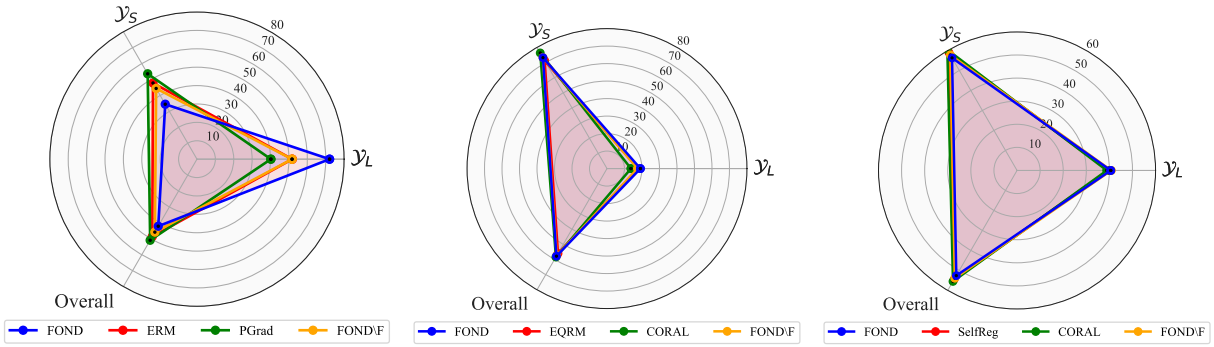
We report the performance under the *Low* and *High* shared class settings in Figure 5.2 and Table 5.2. `FOND` relies on observing domain-shared classes, therefore it only demonstrates top-4 performance in the *Low* setting. In the *High* setting, we observe that `FOND` consistently outperforms all baselines as shown in Table 5.2. Strikingly, `FOND` results in a 39% performance improvement over the best baseline (ERM) for VLCS. Figure 5.2 (top) shows some interesting trends. While VLCS and OfficeHome's performance seems to improve from *Low* to *High*, this is not true for PACS. Does this mean that the shared setting does not help? To probe this further, in Figure 5.2 (bottom), we track only those classes which were domain-linked in both *Low* and *High* settings to see how an increase in domain-shared classes impacts the performance of domain-linked classes. We see that, in fact, for both VLCS and PACS `FOND` improves the performance for domain-linked classes! However, another interesting trend presents itself for OfficeHome. First, we observe that `FOND` outperforms all other baselines in the *High* setting. But when we track classes present in both the *Low* and *High* settings, their performance decreases; refer to Figure 5.2f (bottom). We note that for small class-size datasets (PACS & VLCS), the transition from *Low* to *High* entails making only 2 domain-linked classes become domain-shared. However, for OfficeHome, 25 classes become domain-shared, resulting in algorithms prioritizing the larger domain-shared corpus.

5.2.2 Impact of Inter-Domain Variations

To analyze the impact of domain differences, we turn to dataset characteristics. As shown in Figure 2.2, VLCS domains are real-world, while in PACS and OfficeHome, domain differences are more obvious. We find while `FOND` outperforms all baselines for each of the

Table 5.2: $\mathcal{Y}_L$ **class accuracies under the *Low* and *High* settings.** FOND accomplishes state-of-the-art (SOTA) performance for domain-linked classes by transferring domain-invariant representations from domain-shared ones. In alignment, we observe FONDs large performance improvements when transitioning from the *Low* to *High* setting. The **bolded** and underlined values denote the first and second-best accuracy scores. Additionally, *(-red)* and *(+blue)* values indicate performance differences relative to the naive ERM baseline.

		Datasets			
Setting	**Algorithm**	**VLCS**	**PACS**	**OfficeHome**	**Average**
Low	ERM	50.7 ± 1.0	36.5 ± 0.5	38.5 ± 0.4	41.9 *(0.0)*
	CORAL [63]	45.5 ± 1.6	33.3 ± 0.8	40.7 ± 0.2	39.8 *(-2.1)*
	MLDG [34]	**50.8 ± 2.0**	**38.0 ± 0.1**	38.1 ± 0.1	**42.3** *(+0.4)*
	ARM [82]	47.7 ± 0.9	36.8 ± 1.3	39.0 ± 0.1	41.2 *(-0.7)*
	SelfReg [29]	46.6 ± 1.2	32.4 ± 0.4	40.0 ± 0.3	39.6 *(-2.3)*
	CAD [55]	45.5 ± 1.5	33.0 ± 0.9	36.9 ± 1.2	38.5 *(-3.4)*
	Transfer [80]	48.3 ± 0.6	36.4 ± 1.8	38.1 ± 0.3	40.9 *(-1.0)*
	CausIRL [13]	45.8 ± 0.9	33.6 ± 0.9	40.9 ± 0.2	40.1 *(-1.8)*
	EQRM [15]	49.1 ± 1.1	37.4 ± 0.7	39.9 ± 0.2	42.1 *(+0.2)*
	PGrad [76]	49.0 ± 0.7	34.4 ± 0.5	39.0 ± 0.3	40.8 *(-1.1)*
	FOND	48.0 ± 0.4	35.3 ± 1.2	40.3 ± 0.3	41.2 *(-0.7)*
	FOND\F	48.5 ± 1.0	35.3 ± 0.5	40.6 ± 0.6	41.5 *(-0.4)*
	FOND\FB	50.0 ± 0.2	33.2 ± 0.5	41.0 ± 0.5	41.4 *(-0.5)*
	FOND\FBA	46.6 ± 1.0	35.4 ± 1.2	**41.0 ± 0.4**	41.0 *(-0.9)*
High	ERM	51.8 ± 3.3	14.7 ± 2.2	37.5 ± 0.6	34.6 *(0.0)*
	CORAL [63]	49.8 ± 4.2	13.7 ± 1.0	38.9 ± 0.2	34.1 *(-0.5)*
	MLDG [34]	45.2 ± 3.4	13.8 ± 0.5	37.4 ± 0.7	32.1 *(-2.5)*
	ARM [82]	49.0 ± 1.4	16.2 ± 2.9	38.4 ± 0.2	34.5 *(-0.1)*
	SelfReg [29]	41.9 ± 0.2	13.4 ± 1.2	39.5 ± 0.6	31.6 *(-3.0)*
	CAD [55]	51.7 ± 5.8	13.1 ± 0.7	36.4 ± 1.4	33.7 *(-0.9)*
	Transfer [80]	48.9 ± 3.0	16.0 ± 1.6	36.8 ± 0.2	33.9 *(-0.7)*
	CausIRL [13]	48.9 ± 2.5	13.3 ± 1.5	39.2 ± 0.2	33.8 *(-0.8)*
	EQRM [15]	45.4 ± 3.5	17.9 ± 2.0	37.8 ± 0.1	33.7 *(-0.9)*
	PGrad [76]	40.2 ± 1.8	12.6 ± 1.4	39.0 ± 0.8	30.6 *(-4.0)*
	FOND	**72.1 ± 3.5**	**19.1 ± 0.6**	40.6 ± 0.4	**43.9** *(+9.3)*
	FOND\F	51.7 ± 6.0	17.5 ± 1.4	40.8 ± 0.6	36.7 *(+2.1)*
	FOND\FB	44.0 ± 2.3	15.4 ± 0.6	**41.7 ± 0.7**	33.7 *(-0.9)*
	FOND\FBA	51.3 ± 2.8	17.3 ± 1.3	39.1 ± 0.5	35.9 *(+1.3)*



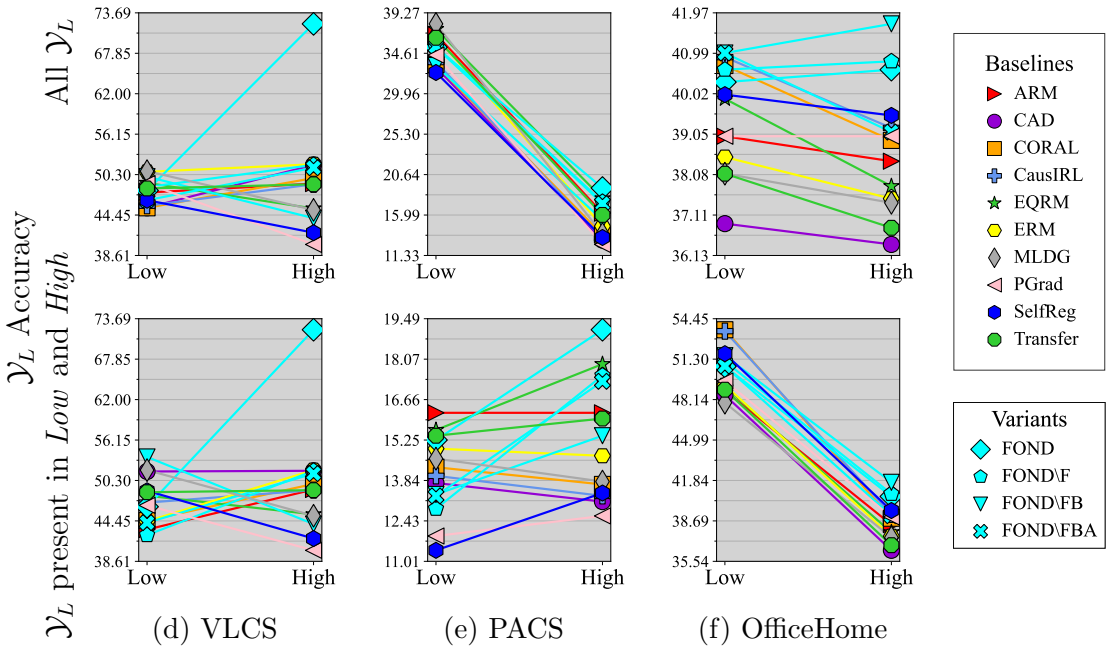
(a) VLCS (b) PACS (c) OfficeHome

Figure 5.1: **Visualizing the $\mathcal{Y}_L$ and $\mathcal{Y}_S$ performance trade-offs on the *High* shared-class distribution setting.** `FOND` outperforms all baselines on $\mathcal{Y}_L$ classes while retaining competitive $\mathcal{Y}_S$ class performance, especially as the total number of target classes increases (left to right). Plotted are the best domain-linked and overall accuracy baselines.

datasets (Table 5.2), the results for VLCS are the most prominent. In fact, each baseline struggles with PACS and OfficeHome. The theoretical analysis sheds light on this trend. Specifically, from Thm. 3.2.1, we see that the distribution divergence $d_{\mathcal{H}\Delta\mathcal{H}}(S^i, T)$ between source and target plays a major role in effective transfer, putting a fundamental limit on performance. Interestingly, `FOND` can leverage domain-shared classes to boost performance.

5.2.3 Impact on Domain-Shared Classes

While this work aims to improve domain-linked performance, a natural question arises regarding the impact on the domain-shared classes. In Figure 5.1, we present a comparative analysis of the $\mathcal{Y}_L$ vs. $\mathcal{Y}_S$ vs. overall accuracy to analyze this performance. Current research on domain generalization mainly concentrates on enhancing overall accuracy. However, this narrow focus hides important details about the strengths and weaknesses of each learning algorithm and which one to choose. For each dataset, `FOND` results in a striking improvement in $\mathcal{Y}_L$ accuracy compared to the best baselines while retaining competitive $\mathcal{Y}_L$ and overall accuracy. We also observe that different `FOND` variants (`FOND\F`) can be used to reconcile the trade-off in the real world.

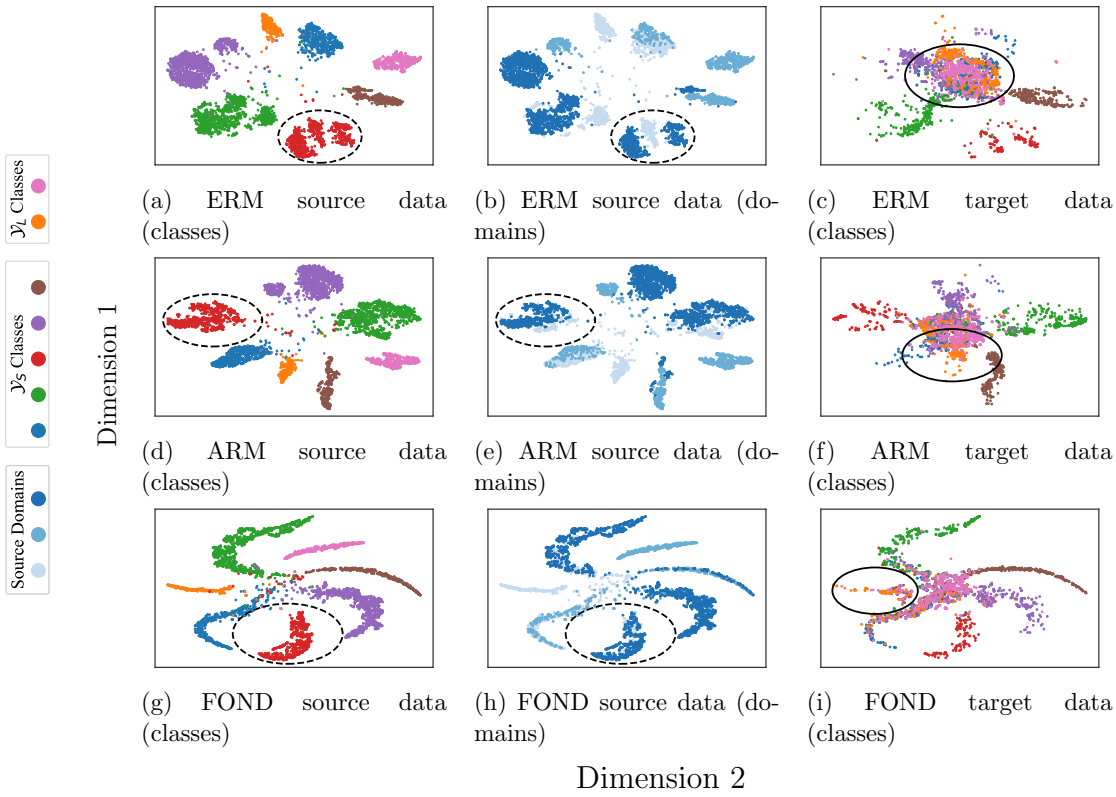


(d) VLCS (e) PACS (f) OfficeHome

Figure 5.2: **Tracking all domain-linked $\mathcal{Y}_L$ classes (top) versus only those present in both *Low* and *High* settings (bottom).** The domain-linked classes tracked in the *High* setting are a subset of those in the *Low* setting; refer to Subsection. 5.1.2. We report the average over domain-linked classes in the *Low* and *High* settings in Table 5.2 (and the top plot). However, we also track the performance changes of only those domain-linked classes in both *Low* and *High* settings.

5.2.4 Visualizing Learned Representations

In Figure 5.3 we visualize latent representations from the PACS-*High* dataset via t-SNE plots [65]; source-domain (*Photo*, *Art* and *Sketch*) representations are colored by class and domain. Target-domain (*Cartoon*) representations are colored by class. To gain insight into the method's strong performance during the *High* shared-class distribution setting, we analyze the representations learned by ERM (naive-baseline), ARM (top-performing-baseline) and `FOND` (top-performing-method); refer to Table 5.2. On source-domain data, the ERM class-colored clusters are distinctly sub-clustered by domain (e.g., broken circle in Figure 5.3a and Figure 5.3b). Whereas ARM (Figure 5.3d and Figure 5.3e) and `FOND` (Figure 5.3g and Figure 5.3h) demonstrate more domain-invariant representations since their classes are not as distinctly clustered by domain. Furthermore, for domain-linked $\mathcal{Y}_L$ class samples, `FOND` (e.g., solid circle in Figure 5.3i) yields more generalizable representa-



(a) ERM source data (classes) (b) ERM source data (domains) (c) ERM target data (classes)

(d) ARM source data (classes) (e) ARM source data (domains) (f) ARM target data (classes)

(g) FOND source data (classes) (h) FOND source data (domains) (i) FOND target data (classes)

Figure 5.3: **t-SNE learned representation visualization for the PACS-*High* dataset**. Each row visualizes the representations of the naive (ERM), top-performing-baseline (ARM) and our (`FOND`) algorithm. Source-domain (*Photo*, *Art* and *Sketch*) representations are colored by class and domain. Target-domain (*Cartoon*) representations are colored by class. We highlight domain-linked $\mathcal{Y}_L$ class generalization (solid circle) and domain-invariant learning (broken circle) in Subsection 5.2.4.

tions than ARM (Figure 5.3f) and ERM (Figure 5.3c). While all methods struggle on the pink class, `FOND` empirically maintains top performance.

5.2.5 Analyzing `FOND` Variants

We observed that the performance between variants (introduced in Table. 4.1) also depends on the abovementioned factors. In this section, we seek practical insights into each variant.

Shared-Class Settings. This, to reiterate was examined by creating *High* and *Low* shared-class settings for each dataset. The fairness objective — `FOND` variant — benefits domain-linked classes as the number of domain-shared samples increases. Consistent with this hypothesis, we observe a $> 7\%$ increase over other variants when moving from the *Low* to *High* shared-class setting in Table 5.2. However, in the *Low* shared-class setting, across all datasets, it is better to pay more attention to inter-domain positive — via α — and intra-domain negative pairs — via β. This is because, with less domain-shared data, it is harder to assert the fairness objective. In alignment, we observe that the fairness-free variant, `FOND\F`, outperforms all other variants in the *Low* setting.

Domain Variations. When domains are not significantly distinguishable (i.e., VLCS dataset), there is no perceived benefit in domain-aware regularizers (i.e. β, α inclusive variants); therefore, the class-aware fairness constraint (`FOND` variant) is best yielding a 10% average improvement over all other variants. With significant domain variances (e.g., PACS), focusing on harder negatives via the β parameter improves domain-linked performance across both *Low* and *High* settings, i.e., `FOND` with β (`FOND\F`) outperforms the variant without it (`FOND\FB`) by 2.1% on average.

Number of Classes. Given a sufficient number of domain-shared classes (*High* setting), the fairness objective is more accessible to impose with fewer classes and, therefore, consistently yielded the best results on the low-class PACS and VLCS datasets. When the number of classes increases ($\sim 10\times$ in OfficeHome), the number of positive inter-domain pairs increases, and we observe that the α-only variant `FOND\FB` is the strongest.

Chapter 6

Conclusion

6.1 Summary of Contributions

Domain generalization (DG) in real-world settings often suffers from data scarcity, leading to classes only observed in specific domains, i.e. they are *domain-linked*. Traditional DG methods often neglect the performance of classes observed only in these (domain-linked classes, focusing instead on overall accuracy. This thesis shifts this focus by improving out-of-distribution generalization for these domain-linked classes, transferring domain-invariant knowledge from domain-shared classes with minimal performance trade-offs through the `FOND` methodology. A significant contribution of this research is the identification of the severe performance discrepancy between domain-linked and domain-shared class performance in existing DG methods, which serves to stimulate further research in this area. By leveraging existing domain adaptation theory, this thesis analyzes when and why domain-shared classes usually generalize better than domain-linked ones, providing a theoretical foundation for the approach.

The experiments conducted in this research were designed to evaluate the effectiveness of `FOND`, its variants and baselines in addressing the generalization challenges when optimizing for both domain-linked and domain-shared classes. Therefore, this thesis evaluated domain-linked class generalization with respect to varying shared-class distributions, types of inter-domain shifts and the number of target classes. The findings demonstrate that `FOND` not only improves the generalization of domain-linked classes but also maintains competitive performance for domain-shared classes. This empirical evidence underscores the robustness and versatility of the proposed method in service to future research.

6.2 Limitations

Despite the promising results, the proposed method has limitations. `FOND`, like all baseline models, struggles in settings with *Low* shared-class availability. While useful, the binary distinction between domain-linked and domain-shared classes may not fully capture the complexities of real-world data distributions. Additionally, the computational demands of DG research, particularly the extensive validation cycles required for each dataset, algorithm, hyper-parameter search space, and shared-class distribution setting, highlight the need for more resource-efficient methods. Furthermore, the fairness-inspired objective $\mathcal{L}_{disc}$ is conditioned on a class being represented in multiple domains, which in the real world may result from representation inequalities of protected attributes and/or classes only observed in specific domains (or rarely observed in others). Therefore, careful consideration is required when deploying fairness-based DG research since they could make decisions that unfairly impact particular groups.

6.3 Future Research

One of the primary goals of this thesis is to stimulate further research on developing domain generalization methods that do not yield significant performance discrepancies between domain-linked and domain-shared classes. Consequently, several avenues for future research should be pursued to build on the insights gained from this research.

6.3.1 Adaptive Methods for Varying *Sharedness* Levels

Future research is needed to develop methods that can dynamically adapt to varying levels of class sharedness, moving beyond the binary distinction between domain-linked and domain-shared classes. In the real world, classes are often distributed unevenly within and across domains. Therefore, creating models that flexibly handle different degrees of class overlap between domains would be an impactful solution. A similar challenge has been observed in the context of data scarcity in long-tailed classification ($50 - 1000$ classes) [20]. Still, we show that DG models struggle even for a modest number of classes, highlighting a need to develop methods that can address this challenging scenario.

6.3.2 Interdisciplinary Applications and Real-World Datasets

This thesis analyzes the challenges of domain-linked class generalization on well-established but well-curated datasets with distinct domain shifts. To validate their effectiveness in real-world deployments, more research is needed to evaluate `FOND` and similar DG methods in various interdisciplinary fields, such as healthcare, finance, and autonomous systems. These settings present a unique challenge because most DG approaches assume that source (training) domains are distinctly different, whereas, in the real world, these differences may not be as noticeable. Furthermore, collaborations with domain experts can provide valuable insights into practical challenges and help refine the models for specific use cases.

6.3.3 Effects of Including Domain-Shared Classes but Optimizing for Domain-Linked Ones

While this research focused on improving domain-linked class generalization, we were still interested in developing a classification model that remained effective for domain-shared classes. However, it would be very informative to investigate if including domain-shared classes during training for a classifier that would only perform inference on the set of domain-linked classes could be beneficial. The driving motivation behind this hypothesis is that including auxiliary classes during training may help a classification model identify better domain-invariant features.

Here is a thought experiment. Referring back to Figure 1.1, let us say we wanted to train a classifier for the triangle and diamond classes, which are only expressed in the blue and green domains. Suppose we wanted this classifier to learn domain-invariant features like colour invariance. In that case, it might be helpful for the model to observe samples from the domain-shared circle class featured in both the red and blue domains.

6.3.4 Leveraging The Prior Knowledge of Foundation Models

Lastly, leveraging the cross-domain prior knowledge embedded in large foundation models can substantially improve domain generalization. Pre-trained on extensive and diverse datasets, foundation models possess rich representations that may be fine-tuned for specific downstream tasks [69] and now recently domain generalization [5].

References

[1] Isabela Albuquerque, João Monteiro, Tiago H. Falk, and Ioannis Mitliagkas. Adversarial target-invariant representation learning for domain generalization. *ArXiv*, abs/1911.00804, 2019.

[2] Parvin Esmaeili Ataabadi, Behzad Soleimani Neysiani, Mohammad Zahiri Nogorani, and Nazanin Mehraby. Semi-supervised medical insurance fraud detection by predicting indirect reductions rate using machine learning generalization capability. In *2022 8th International Conference on Web Research (ICWR)*, pages 176–182, 2022.

[3] Shai Ben-David, John Blitzer, Koby Crammer, Alex Kulesza, Fernando C Pereira, and Jennifer Wortman Vaughan. A theory of learning from different domains. *Machine Learning*, 79:151–175, 2009.

[4] Shai Ben-David, John Blitzer, Koby Crammer, and Fernando C Pereira. Analysis of representations for domain adaptation. In *NIPS*, 2006.

[5] Yasser Benigmim, Subhankar Roy, Slim Essid, Vicky Kalogeiton, and Stéphane Lathuilière. Collaborating foundation models for domain generalized semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3108–3119, 2024.

[6] Paul Bergmann, Michael Fauser, David Sattlegger, and Carsten Steger. Mvtec ad — a comprehensive real-world dataset for unsupervised anomaly detection. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9584–9592, 2019.

[7] Meng Cao and Songcan Chen. Mixup-induced domain extrapolation for domain generalization. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 11168–11176, 2024.

[8] Junbum Cha, Kyungjae Lee, Sungrae Park, and Sanghyuk Chun. Domain generalization by mutual-information regularization with pre-trained models. In *European conference on computer vision*, pages 440–457. Springer, 2022.

[9] Chaoqi Chen, Luyao Tang, Leitian Tao, Hong-Yu Zhou, Yue Huang, Xiaoguang Han, and Yizhou Yu. Activate and reject: Towards safe domain generalization under category shift. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 11552–11563, October 2023.

[10] Irene Y. Chen, Emma Pierson, Sherri Rose, Shalmali Joshi, Kadija Ferryman, and Marzyeh Ghassemi. Ethical machine learning in healthcare. *Annual Review of Biomedical Data Science*, 4(1):123–144, 2021.

[11] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pages 1597–1607. PMLR, 13–18 Jul 2020.

[12] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey E. Hinton. A simple framework for contrastive learning of visual representations. *ArXiv*, abs/2002.05709, 2020.

[13] Mathieu Chevalley, Charlotte Bunne, Andreas Krause, and Stefan Bauer. Invariant causal mechanisms through distribution matching. *arXiv preprint arXiv:2206.11646*, 2022.

[14] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, K. Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 248–255, 2009.

[15] Cian Eastwood, Alexander Robey, Shashank Singh, Julius Von Kügelgen, Hamed Hassani, George J Pappas, and Bernhard Schölkopf. Probable domain generalization via quantile risk minimization. *Advances in Neural Information Processing Systems*, 35:17340–17358, 2022.

[16] Chen Fang, Ye Xu, and Daniel N. Rockmore. Unbiased metric learning: On the utilization of multiple datasets and web images for softening bias. *2013 IEEE International Conference on Computer Vision*, pages 1657–1664, 2013.

[17] Chelsea Finn, P. Abbeel, and Sergey Levine. Model-agnostic meta-learning for fast adaptation of deep networks. In *International Conference on Machine Learning*, 2017.

[18] Yaroslav Ganin, E. Ustinova, Hana Ajakan, Pascal Germain, H. Larochelle, François Laviolette, Mario Marchand, and Victor S. Lempitsky. Domain-adversarial training of neural networks. In *Journal of machine learning research*, 2015.

[19] Rui Gong, Wen Li, Yuhua Chen, and Luc Van Gool. Dlow: Domain flow for adaptation and generalization. *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2472–2481, 2018.

[20] Xiao Gu, Yao Guo, Zeju Li, Jianing Qiu, Qianming Dou, Yuxuan Liu, Benny P. L. Lo, and Guangxu Yang. Tackling long-tailed category distribution under domain shifts. In *European Conference on Computer Vision*, 2022.

[21] Ishaan Gulrajani and David Lopez-Paz. In search of lost domain generalization. In *International Conference on Learning Representations*, 2021.

[22] Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 16000–16009, 2022.

[23] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[24] Shoubo Hu, Kun Zhang, Zhitang Chen, and Lai-Wan Chan. Domain generalization via multidomain discriminant analysis. *Uncertainty in artificial intelligence : proceedings of the ... conference. Conference on Uncertainty in Artificial Intelligence*, 35, 2019.

[25] Zeyi Huang, Haohan Wang, Eric P. Xing, and Dong Huang. Self-challenging improves cross-domain generalization. In *European Conference on Computer Vision*, 2020.

[26] Seogkyu Jeon, Kibeom Hong, Pilhyeon Lee, Jewook Lee, and Hyeran Byun. Feature stylization and domain-aware contrastive learning for domain generalization. In *Proceedings of the 29th ACM International Conference on Multimedia*, pages 22–31, 2021.

[27] Prannay Khosla, Piotr Teterwak, Chen Wang, Aaron Sarna, Yonglong Tian, Phillip Isola, Aaron Maschinot, Ce Liu, and Dilip Krishnan. Supervised contrastive learning. In *Advances in Neural Information Processing Systems*, volume 33, pages 18661–18673. Curran Associates, Inc., 2020.

[28] Niki Kilbertus, Mateo Rojas Carulla, Giambattista Parascandolo, Moritz Hardt, Dominik Janzing, and Bernhard Schölkopf. Avoiding discrimination through causal reasoning. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017.

[29] Daehee Kim, Seunghyun Park, Jinkyu Kim, and Jaekoo Lee. Selfreg: Self-supervised contrastive regularization for domain generalization. *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 9599–9608, 2021.

[30] Minyoung Kim, Da Li, and Timothy Hospedales. Domain generalisation via domain adaptation: An adversarial fourier amplitude approach. *arXiv preprint arXiv:2302.12047*, 2023.

[31] Taehoon Kim and Bohyung Han. Randomized adversarial style perturbations for domain generalization. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 2317–2325, 2024.

[32] Juli Kumari, Ela Kumar, and Deepak Kumar. A structured analysis to study the role of machine learning and deep learning in the healthcare sector with big data analytics. *Archives of Computational Methods in Engineering*, 30(6):3673–3701, 2023.

[33] Da Li, Yongxin Yang, Yi-Zhe Song, and Timothy M Hospedales. Deeper, broader and artier domain generalization. In *Proceedings of the IEEE international conference on computer vision*, pages 5542–5550, 2017.

[34] Da Li, Yongxin Yang, Yi-Zhe Song, and Timothy M. Hospedales. Learning to generalize: Meta-learning for domain generalization. In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence and Thirtieth Innovative Applications of Artificial Intelligence Conference and Eighth AAAI Symposium on Educational Advances in Artificial Intelligence*, AAAI'18/IAAI'18/EAAI'18. AAAI Press, 2018.

[35] Haoliang Li, Sinno Jialin Pan, Shiqi Wang, and Alex Chichung Kot. Domain generalization with adversarial feature learning. *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5400–5409, 2018.

[36] Jie Liu, Xiaoqing Guo, and Yixuan Yuan. Unknown-oriented learning for open set domain adaptation. In *European Conference on Computer Vision*, pages 334–350. Springer, 2022.

[37] Xiaofeng Liu, Bo Hu, Linghao Jin, Xu Han, Fangxu Xing, Jinsong Ouyang, Jun Lu, Georges EL Fakhri, and Jonghye Woo. Domain generalization under conditional and label shifts via variational bayesian inference. *arXiv preprint arXiv:2107.10931*, 2021.

[38] Yue Liu, Xihong Yang, Sihang Zhou, Xinwang Liu, Zhen Wang, Ke Liang, Wenxuan Tu, Liang Li, Jingcan Duan, and Cancan Chen. Hard sample aware network for contrastive deep graph clustering. In *Proceedings of the AAAI conference on artificial intelligence*, volume 37, pages 8914–8922, 2023.

[39] Aengus Lynch, Gbètondji JS Dovonon, Jean Kaddour, and Ricardo Silva. Spawrious: A benchmark for fine control of spurious correlation biases. *arXiv preprint arXiv:2303.05470*, 2023.

[40] Karima Makhlouf, Sami Zhioua, and Catuscia Palamidessi. Machine learning fairness notions: Bridging the gap with real-world applications. *Information Processing & Management*, 58(5):102642, 2021.

[41] Massimiliano Mancini, Zeynep Akata, Elisa Ricci, and Barbara Caputo. Towards recognizing unseen categories in unseen domains. In *Computer Vision – ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXIII*, page 466–483, Berlin, Heidelberg, 2020. Springer-Verlag.

[42] Saeid Motiian, Marco Piccirilli, Donald A. Adjeroh, and Gianfranco Doretto. Unified deep supervised domain adaptation and generalization. *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 5716–5726, 2017.

[43] Vaishnavh Nagarajan, Anders Andreassen, and Behnam Neyshabur. Understanding the failure modes of out-of-distribution generalization. *arXiv preprint arXiv:2010.15775*, 2020.

[44] A. Tuan Nguyen, Toan Tran, Yarin Gal, and Atilim Gunes Baydin. Domain invariant representation learning with domain density transformations. In *Advances in Neural Information Processing Systems*, volume 34, pages 5264–5275. Curran Associates, Inc., 2021.

[45] Joshua Niemeijer, Manuel Schwonberg, Jan-Aike Termöhlen, Nico M Schmidt, and Tim Fingscheidt. Generalization by adaptation: Diffusion-based domain extension for domain-generalized semantic segmentation. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 2830–2840, 2024.

[46] Xingchao Peng, Qinxun Bai, Xide Xia, Zijun Huang, Kate Saenko, and Bo Wang. Moment matching for multi-source domain adaptation. *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 1406–1415, 2019.

[47] Thai-Hoang Pham, Xueru Zhang, and Ping Zhang. Fairness and accuracy under domain generalization. In *The Eleventh International Conference on Learning Representations*, 2023.

[48] Fabrizio J. Piva, Daan de Geus, and Gijs Dubbelman. Empirical generalization study: Unsupervised domain adaptation vs. domain generalization methods for semantic segmentation in the wild. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 499–508, January 2023.

[49] Xiaorong Qin, Xinhang Song, and Shuqiang Jiang. Bi-level meta-learning for few-shot domain generalization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 15900–15910, June 2023.

[50] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR, 2021.

[51] Mohaimenul Azam Khan Raiaan, Md. Saddam Hossain Mukta, Kaniz Fatema, Nur Mohammad Fahad, Sadman Sakib, Most Marufatul Jannat Mim, Jubaer Ahmad, Mohammed Eunus Ali, and Sami Azam. A review on large language models: Architectures, applications, taxonomies, open issues and challenges. *IEEE Access*, 12:26839–26874, 2024.

[52] Alexandre Rame, Corentin Dancette, and Matthieu Cord. Fishr: Invariant gradient variances for out-of-distribution generalization. In Kamalika Chaudhuri, Stefanie Jegelka, Le Song, Csaba Szepesvari, Gang Niu, and Sivan Sabato, editors, *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pages 18347–18377. PMLR, 17–23 Jul 2022.

[53] Yuchen Ren, Zhendong Mao, Shancheng Fang, Yan Lu, Tong He, Hao Du, Yongdong Zhang, and Wanli Ouyang. Crossing the gap: Domain generalization for image captioning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2871–2880, June 2023.

[54] Joshua Robinson, Ching-Yao Chuang, Suvrit Sra, and Stefanie Jegelka. Contrastive learning with hard negative samples. In *International Conference on Learning Representations (ICLR)*, 2021.

[55] Yangjun Ruan, Yann Dubois, and Chris J. Maddison. Optimal representations for covariate shift. In *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*. OpenReview.net, 2022.

[56] Johannes Schneider and Michalis Vlachos. A survey of deep learning: From activations to transformers. In *International Conference on Agents and Artificial Intelligence*, 2023.

[57] Florian Schroff, Dmitry Kalenichenko, and James Philbin. Facenet: A unified embedding for face recognition and clustering. *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 815–823, 2015.

[58] Seonguk Seo, Yumin Suh, Dongwan Kim, Jongwoo Han, and Bohyung Han. Learning to optimize domain specific normalization for domain generalization. In *European Conference on Computer Vision*, 2019.

[59] Shiv Shankar, Vihari Piratla, Soumen Chakrabarti, Siddhartha Chaudhuri, Preethi Jyothi, and Sunita Sarawagi. Generalizing across domains via cross-gradient training. In *International Conference on Learning Representations*, 2018.

[60] Rui Shao, Xiangyuan Lan, Jiawei Li, and Pong C Yuen. Multi-adversarial discriminative deep domain generalization for face presentation attack detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10023–10031, 2019.

[61] Connor Shorten and Taghi M. Khoshgoftaar. A survey on image data augmentation for deep learning. *Journal of Big Data*, 6:1–48, 2019.

[62] Yang Shu, Zhangjie Cao, Chenyu Wang, Jianmin Wang, and Mingsheng Long. Open domain generalization with domain-augmented meta-learning. *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9619–9628, 2021.

[63] Baochen Sun and Kate Saenko. Deep coral: Correlation alignment for deep domain adaptation. In Gang Hua and Hervé Jégou, editors, *Computer Vision – ECCV 2016 Workshops*, pages 443–450, Cham, 2016. Springer International Publishing.

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

[65] Laurens van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of Machine Learning Research*, 9(86):2579–2605, 2008.

[66] Vladimir Vapnik, Esther Levin, and Yann Le Cun. Measuring the vc-dimension of a learning machine. *Neural Computation*, 6(5):851–876, 1994.

[67] Vladimir Naumovich Vapnik. The nature of statistical learning theory. In *Statistics for Engineering and Information Science*, 2000.

[68] Ramakrishna Vedantam, David Lopez-Paz, and David J Schwab. An empirical investigation of domain generalization with empirical risk minimizers. In *Advances in Neural Information Processing Systems*, volume 34, pages 28131–28143. Curran Associates, Inc., 2021.

[69] Raviteja Vemulapalli, Hadi Pouransari, Fartash Faghri, Sachin Mehta, Mehrdad Farajtabar, Mohammad Rastegari, and Oncel Tuzel. Knowledge transfer from vision foundation models for efficient training of small task-specific models. In *Forty-first International Conference on Machine Learning*, 2024.

[70] Hemanth Venkateswara, Jose Eusebio, Shayok Chakraborty, and Sethuraman Panchanathan. Deep hashing network for unsupervised domain adaptation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5018–5027, 2017.

[71] Anna Vettoruzzo, Mohamed-Rafik Bouguelia, Joaquin Vanschoren, Thorsteinn Rögnvaldsson, and KC Santosh. Advances and challenges in meta-learning: A technical review. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 46(7):4763–4779, 2024.

[72] Riccardo Volpi, Hongseok Namkoong, Ozan Sener, John C Duchi, Vittorio Murino, and Silvio Savarese. Generalizing to unseen domains via adversarial data augmentation. *Advances in neural information processing systems*, 31, 2018.

[73] Jindong Wang, Cuiling Lan, Chang Liu, Yidong Ouyang, and Tao Qin. Generalizing to unseen domains: A survey on domain generalization. In *International Joint Conference on Artificial Intelligence*, 2021.

[74] Junchang Wang, Yang Li, Liyan Xie, and Yao Xie. Class-conditioned domain generalization via wasserstein distributional robust optimization. *ArXiv*, abs/2109.03676, 2021.

[75] Zeyu Wang, Klint Qinami, Ioannis Christos Karakozis, Kyle Genova, Prem Nair, Kenji Hata, and Olga Russakovsky. Towards fairness in visual recognition: Effective strategies for bias mitigation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8919–8928, 2020.

[76] Zhe Wang, Jake Grigsby, and Yanjun Qi. Pgrad: Learning principal gradients for domain generalization. In *The Eleventh International Conference on Learning Representations*, 2023.

[77] Fu-En Yang, Yuan-Chia Cheng, Zu-Yun Shiau, and Yu-Chiang Frank Wang. Adversarial teacher-student representation learning for domain generalization. In *Advances in Neural Information Processing Systems*, volume 34, pages 19448–19460. Curran Associates, Inc., 2021.

[78] Xufeng Yao, Yang Bai, Xinyun Zhang, Yuechen Zhang, Qi Sun, Ran Chen, Ruiyu Li, and Bei Yu. Pcl: Proxy-based contrastive learning for domain generalization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7097–7107, 2022.

[79] Hao-Tong Ye, Chuanlong Xie, Tianle Cai, Ruichen Li, Zhenguo Li, and Liwei Wang. Towards a theoretical framework of out-of-distribution generalization. In *Neural Information Processing Systems*, 2021.

[80] Guojun Zhang, Han Zhao, Yaoliang Yu, and Pascal Poupart. Quantifying and improving transferability in domain generalization. In *Advances in Neural Information Processing Systems*, 2021.

[81] Hongyi Zhang, Moustapha Cisse, Yann N. Dauphin, and David Lopez-Paz. mixup: Beyond empirical risk minimization. In *International Conference on Learning Representations*, 2018.

[82] Marvin Mengxin Zhang, Henrik Marklund, Nikita Dhawan, Abhishek Gupta, Sergey Levine, and Chelsea Finn. Adaptive risk minimization: Learning to adapt to domain shift. In A. Beygelzimer, Y. Dauphin, P. Liang, and J. Wortman Vaughan, editors, *Advances in Neural Information Processing Systems*, 2021.

[83] Michael Zhang, Nimit S Sohoni, Hongyang R Zhang, Chelsea Finn, and Christopher Re. Correct-n-contrast: a contrastive approach for improving robustness to spurious correlations. In *International Conference on Machine Learning*, pages 26484–26516. PMLR, 2022.

[84] Shaofeng Zhang, Meng Liu, Junchi Yan, Hengrui Zhang, Lingxiao Huang, Xiaokang Yang, and Pinyan Lu. M-mix: Generating hard negatives via multi-sample mixing for contrastive learning. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 2461–2470, 2022.

[85] Han Zhao, Chen Dan, Bryon Aragam, Tommi S Jaakkola, Geoffrey J Gordon, and Pradeep Ravikumar. Fundamental limits and tradeoffs in invariant representation learning. *Journal of machine learning research*, 23(340):1–49, 2022.

[86] Tao Zhong, Zhixiang Chi, Li Gu, Yang Wang, Yuanhao Yu, and Jin Tang. Meta-dmoe: Adapting to domain shift by meta-distillation from mixture-of-experts. *Advances in Neural Information Processing Systems*, pages 22243–22257, 2022.

[87] Fan Zhou, Zhuqing Jiang, Changjian Shui, Boyu Wang, and Brahim Chaib-draa. Domain generalization via optimal transport with metric similarity learning. *Neurocomputing*, 456:469–480, 2021.

[88] Kaiyang Zhou, Ziwei Liu, Yu Qiao, Tao Xiang, and Chen Change Loy. Domain generalization: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(4):4396–4415, 2023.

[89] Wei Zhu, Le Lu, Jing Xiao, Mei Han, Jiebo Luo, and Adam P. Harrison. Localized adversarial domain generalization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7108–7118, June 2022.