Domain Invariant Representation Learning with Domain Density Transformations

A. Tuan Nguyen [1] Toan Tran [2] Yarin Gal [1] Atilim Gunes Baydin [1]

Abstract

Domain generalization refers to the problem where we aim to train a model on data from a set of source domains so that the model can generalize to unseen target domains. Naively training a model on the aggregate set of data (pooled from all source domains) has been shown to perform suboptimally, since the information learned by that model might be domain-specific and generalize imperfectly to target domains. To tackle this problem, a predominant approach is to find and learn some domain-invariant information in order to use it for the prediction task. In this paper, we propose a theoretically grounded method to learn a domain-invariant representation by enforcing the representation network to be invariant under all transformation functions among domains. We also show how to use generative adversarial networks to learn such domain transformations to implement our method in practice. We demonstrate the effectiveness of our method on several widely used datasets for the domain generalization problem, on all of which we achieve competitive results with state-of-the-art models.

1. Introduction

Domain generalization refers to the machine learning scenario where the model is trained on multiple source domains so that it is expected to generalize well to unseen target domains. The key difference between domain generalization (Khosla et al., 2012; Muandet et al., 2013; Ghifary et al., 2015) and domain adaptation (Zhao et al., 2019; Zhang et al., 2019; Combes et al., 2020; Tanwani, 2020) is that, in domain generalization, the learner does not have access to (even a small amount of) data of the target domain, making the problem much more challenging.

One of the most common domain generalization approaches is to learn an invariant representation across domains, aiming at a good generalization performance on target domains.

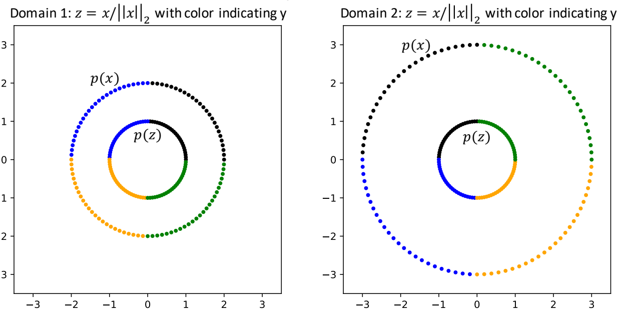


Figure 1. **An example of two domains**. For each domain, x is uniformly distributed on the outer circle (radius 2 for domain 1 and radius 3 for domain 2), with the color indicating class label y. After the transformation $z = x/||x||_2$, the marginal of z is aligned (uniformly distributed on the unit circle for both domains), but the conditional $p(y|z)$ is not aligned. Thus, using this representation for predicting y would not generalize well across domains.

In the representation learning framework, the prediction $y = f(x)$, where x is data and y is a label, is obtained as a composition $y = h \circ g(x)$ of a deep representation network $z = g(x)$, where z is a learned representation of data x, and a smaller classifier $y = h(z)$, predicting label y given representation z, both of which are shared across domains.

Current "domain-invariance"-based methods in domain generalization focus on either the marginal distribution alignment (Muandet et al., 2013) or the conditional distribution alignment (Li et al., 2018b;c), which are still prone to distributional shifts if the conditional or marginal (respectively) data distribution is not stable. In particular, the marginal alignment refers to making the representation distribution $p(z)$ to be the same across domains. This is essential since if $p(z)$ for the target domain is different from that of source domains, the classification network $h(z)$ would face out-of-distribution data because the representation z it receives as input at test time would be different from the ones it was trained with in source domains. Conditional alignment refers to aligning $p(y|z)$, the conditional distribution of the label given the representation, since if this conditional for the target domain is different from that of the source domains, the classification network (trained on the source domains) would give inaccurate predictions at test time. The formal definition of the two alignments is discussed in Section 3.

[1] University of Oxford [2] VinAI Research. Correspondence to: A. Tuan Nguyen <tuan.nguyen@cs.ox.ac.uk>.

In Figure 1 we illustrate an example where the representation z satisfies the marginal alignment but not the conditional alignment. Specifically, x is distributed uniformly on the circle with radius 2 (and centered at the origin) for domain 1 and distributed uniformly on the circle with radius 3 (centered at the origin) for domain 2. The representation z defined by the mapping $z = g(x) = x/||x||_2$ will align the marginal distribution $p(z)$, i.e., z is now distributed uniformly on the unit circle for both domains. However, the conditional distribution $p(y|z)$ is not aligned between the two domains (y is represented by color), which means using this representation for classification is suboptimal, and in this extreme case would lead to 0% accuracy in the target domain 2. This is an extreme case of misalignment but it does illustrate the importance of the conditional alignment. Therefore, we need to align both the marginal and the conditional distributions for a domain-invariant representation.

There have been several attempts recently to align both the marginal and conditional distribution in a domain adaptation problem, for example, (Tanwani, 2020), by leveraging a small set of labeled data of the target domain. However, it is challenging to apply this approach directly to domain generalization because we do not have access to data in the target domain.

In this paper, we focus on learning a domain-invariant representation that aligns both the marginal and the conditional distribution in domain generalization problems. We present theoretical results regarding the conditions for the existence of domain-invariant representations, and subsequently propose a method to learn such representations based on domain density transformation functions. A simple intuition for our approach is that if we enforce the representation to be invariant under the transformations among source domains, the representation will become more robust under other domain transformations.

Furthermore, we introduce an implementation of our method in practice, in which the domain transformation functions are learned through the training process of generative adversarial networks (GANs) (Goodfellow et al., 2014; Choi et al., 2018). We conduct extensive experiments on several widely used datasets and observe a significant improvement over the naive baseline of training a model normally on the aggregate dataset from all domains. We also compare our methods against other state-of-the-art models and show that our method achieves competitive results.

Our contribution in this work is threefold:

- We shed light on the domain generalization problem by providing several theoretical observations: a necessary and sufficient condition for the existence of a domain-invariant representation and a connection between domain-independent representation and a marginally-aligned representation.
- We propose a theoretically grounded method for learning a domain-invariant representation based on domain density transformation functions. We also demonstrate that we can learn the domain transformation functions by GANs in order to implement our approach in practice.
- We show the effectiveness of our methods by performing experiments on widely used domain generalization datasets (e.g., Rotated MNIST, PACS and OfficeHome) and compare with relevant baselines (especially DGER (Zhao et al., 2020), a main baseline that also aims to learn domain invariant representations).

2. Related Work

Domain Generalization: Domain generalization is an important task in real-world machine learning problems since the data distribution of a target domain might vary from that of the source domains which a model is trained on. Therefore, extensive research has been developed focusing on learning a model that generalizes well to unseen target domains. While the literature is vast, here we cover the most important works that are related to ours. A predominant approach for domain generalization is domain invariance (Muandet et al., 2013; Li et al., 2018b;c; Arjovsky et al., 2019; Wang et al., 2020; Muandet et al., 2013; Akuzawa et al., 2019; Ilse et al., 2020; Zhao et al., 2020). Our method falls into this category since we propose a method that learns a domain-invariant representation (which we define as to align both the marginal distribution of the representation and the conditional distribution of the output given the representation). We consider DGER (Zhao et al., 2020), which also learns a representation that aligns both the marginal and conditional distribution via an adversarial loss and an entropy regularizer, one of the main baselines to ours. It should be noted that Zhao et al. (2020) assume the label is distributed uniformly on all domains, which is stronger than our assumption that the distribution of label is stable across domains (and not necessarily uniform). We also show later in our paper that the invariance of the distribution of class label across domains is indeed the necessary and sufficient condition for the existence of a domain-invariant representation. We provide a unified theoretical discussion about the two alignments and a method to learn a representation that aligns both the marginal and conditional distributions via domain density transformation functions for the domain generalization problem.

Another line of methods that received a recent surge in interest is applying the idea of meta-learning for domain generalization problems (Du et al., 2020; Balaji et al., 2018; Li et al., 2018a; Behl et al., 2019). The core idea behind

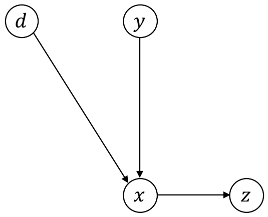


Figure 2. **Graphical model**. Each domain d defines a data distribution $p(x, y|d)$. We want to learn a representation z with a mapping from x so that $p(z|x)$ can be generalized between domains.

these works is that if we train a model that can adapt among source domains well, it would be more likely to adapt to unseen target domains.

Finally, there are approaches (Ding & Fu, 2017; Chattopadhyay et al., 2020; Seo et al., 2019) that make use of the domain specificity, together with domain invariance, for the prediction problem. The argument here is that domain invariance, while being generalized well between domains, might be insufficient for the prediction of each specific domain and thus domain specificity is necessary.

We would like to emphasize that our method is not a direct competitor of meta-learning based and domain specificity based methods. In fact, we expect that our method can be used in conjunction with these methods to get the best of both worlds for better performance.

Density transformation between domains: Since our method is based on domain density transformations, we will review briefly some related work here. To transform the data density between domains, one can use several types of generative models. Two common methods are based on GANs (Zhu et al., 2017; Choi et al., 2018; 2020) and normalizing flows (Grover et al., 2020). Although our method is not limited to the choice of the generative model used for learning the domain transformation functions, we opt to use GAN, specifically StarGAN (Choi et al., 2018), for its rich network capacity. This is just an implementation choice to demonstrate the use and effectiveness of our method in practice, and it is unrelated to our theoretical results.

Connection to contrastive learning: Our method can be interpreted intuitively as a way to learn a representation network that is invariant (robust) under domain transformation functions. On the other hand, contrastive learning (Chen et al., 2020a;b; Misra & Maaten, 2020) is also a representation learning paradigm where the model learns images' similarity. In particular, contrastive learning encourages the representation of an input to be similar under different transformations (usually image augmentations). However, the transformations in contrastive learning are not learned and do not serve the purpose of making the representation robust under domain transformations. Our method first learns the transformations between domains and then uses them to learn a representation that is invariant under domain shifts.

3. Theoretical Approach

3.1. Problem Statement

Let us define the data distribution for a domain $d \in \mathcal{D}$ by $p(x, y|d)$, where the variable $x \in \mathcal{X}$ represents the data and $y \in \mathcal{Y}$ is the corresponding label. The graphical model for our domain generalization framework is depicted in Figure 2, in which the joint distribution is presented as follows:

$$p(d, x, y, z) = p(d)p(y)p(x|y, d)p(z|x) \ . \tag{1}$$

In the domain generalization problem, the data distribution $p(x, y|d)$ varies between domains, thus we expect changes in the marginal data distribution $p(x|d)$ or the conditional data distribution $p(y|x, d)$ or both. In this paper, we assume that $p(y|d)$ is invariant across domains, i.e., y is not dependent on d—this assumption is shown to be the key condition for the existence of a domain-invariant representation (see Theorem 1). This is practically reasonable since in many classification datasets, the class distribution can be assumed to be unchanged across domains (usually uniform distribution among the classes, e.g., balanced datasets).

Our aim is to find a domain-invariant representation z represented by the mapping $p(z|x)$ that can be used for the classification of label y and be generalized among domains. In practice, this mapping can be deterministic (in that case, $p(z|x) = \delta_{g_\theta(x)}(z)$ with some function g_θ, where δ is the Dirac delta distribution) or probabilistic (e.g., a normal distribution with the mean and standard deviation outputted by a network parameterized by θ). For all of our experiments, we use a deterministic mapping for an efficient inference at test time, while in this section, we present our theoretical results with the general case of a distribution $p(z|x)$.

In most existing domain generalization approaches, the domain-invariant representation z is defined using one of the two following definitions:

Definition 1. ***(Marginal Distribution Alignment)*** *The representation z is said to satisfy the marginal distribution alignment condition if $p(z|d)$ is invariant w.r.t. d.*

Definition 2. ***(Conditional Distribution Alignment)*** *The representation z is said to satisfy the conditional distribution alignment condition if $p(y|z, d)$ is invariant w.r.t. d.*

However, when the data distribution varies between domains, it is crucial to align both the marginal and the conditional distribution of the representation z. To this end, this paper aims to learn a representation z that satisfies both the marginal and conditional alignment conditions. We justify our assumption of independence between y and d (thus $p(y|d) = p(y)$) by the following theorem, which shows that this assumption turns out to be the necessary and sufficient condition for learning a domain-invariant representation.

Theorem 1. *The invariance of $p(y|d)$ across domains is the necessary and sufficient condition for the existence of a domain-invariant representation (that aligns both the marginal and conditional distribution).*

Proof. i) If there exists a representation z defined by the mapping $p(z|x)$ that aligns both the marginal and conditional distribution, then $\forall d, d', y$ we have:

$$\begin{aligned} p(y, z|d) &= p(z|d)p(y|z, d) \\ &= p(z|d')p(y|z, d') = p(y, z|d'). \end{aligned} \tag{2}$$

By marginalizing both sides of Eq 2 over z, we get $p(y|d) = p(y|d')$.

ii) If $p(y|d)$ is unchanged w.r.t. the domain d, then we can always find a domain invariant representation, for example, $p(z|x) = \delta_0(z)$ for the deterministic case (that maps all x to 0), or $p(z|x) = \mathcal{N}(z; 0, 1)$ for the probabilistic case.

These representations are trivial and not of our interest since they are uninformative of the input x. However, the readers can verify that they do align both the marginal and conditional distribution of data.

□

It is also worth noting that methods which learn a domain independent representation, for example, Ilse et al. (2020), only align the marginal distribution. This comes directly from the following remark:

Remark 1. *A representation z satisfies the marginal distribution alignment condition if and only if $I(z, d) = 0$, where $I(z, d)$ is the mutual information between z and d.*

Proof.

- If $I(z, d) = 0$, then $p(z|d) = p(z)$, which means $p(z|d)$ is invariant w.r.t. d.
- If $p(z|d)$ is invariant w.r.t. d, then $\forall z, d$:

$$\begin{aligned} p(z) &= \int p(z|d')p(d')\mathrm{d}d' = \int p(z|d)p(d')\mathrm{d}d' \\ &\quad (\text{since } p(z|d') = p(z|d) \forall d') \\ &= p(z|d) \int p(d')\mathrm{d}d' = p(z|d) \\ &\implies I(z, d) = 0 \end{aligned} \tag{3}$$

□

The question still remains that how we can learn a non-trivial domain invariant representation that satisfies both of the distribution alignment conditions. This will be discussed in the following subsection.

3.2. Learning a Domain-Invariant Representation with Domain Density Transformation Functions

To present our method, we will make some assumptions about the data distribution. Specifically, for any two domains d, d', we assume that there exists an invertible and differentiable function denoted by $f_{d,d'}$ that transforms the density $p(x|y, d)$ to $p(x'|y, d')$ $\forall y$. Let $f_{d,d'}$ be the inverse of $f_{d',d}$, i.e., $f_{d',d} := (f_{d,d'})^{-1}$.

Due to the invertibility and differentiability of f's, we can apply the change of variables theorem (Rudin, 2006; Bogachev, 2007). In particular, with $x' = f_{d,d'}(x)$ (and thus $x = f_{d',d}(x')$), we have

$$p(x|y, d) = p(x'|y, d') \left|\det J_{f_{d',d}}(x')\right|^{-1} \tag{4}$$

where $J_{f_{d',d}}(x')$ is the Jacobian matrix of the function $f_{d',d}$ evaluated at x'.

Multiplying both sides of Eq 4 with $p(y|d) = p(y|d')$, we get

$$p(x, y|d) = p(x', y|d') \left|\det J_{f_{d',d}}(x')\right|^{-1} \tag{5}$$

and marginalizing both sides of the above equation over y gives us

$$p(x|d) = p(x'|d') \left|\det J_{f_{d',d}}(x')\right|^{-1} \tag{6}$$

By using Eq 4 and Eq 6, we can prove the following theorem, which offers a way to learn a domain-invariant representation, given the transformation functions f's between domains.

Theorem 2. *Given an invertible and differentiable function $f_{d,d'}$ (with the inverse $f_{d',d}$) that transforms the data density from domain d to d' (as described above). Assuming that the representation z satisfies:*

$$p(z|x) = p(z|f_{d,d'}(x)),\ \forall x \tag{7}$$

Then it aligns both the marginal and the conditional of the data distribution for domain d and d'.

Proof. i) Marginal alignment: $\forall z$ we have:

$$\begin{aligned} p(z|d) &= \int p(x|d)p(z|x)\mathrm{d}x \\ &= \int p(f_{d',d}(x')|d)p(z|f_{d',d}(x')) \left|\det J_{f_{d',d}}(x')\right| \mathrm{d}x' \end{aligned}$$

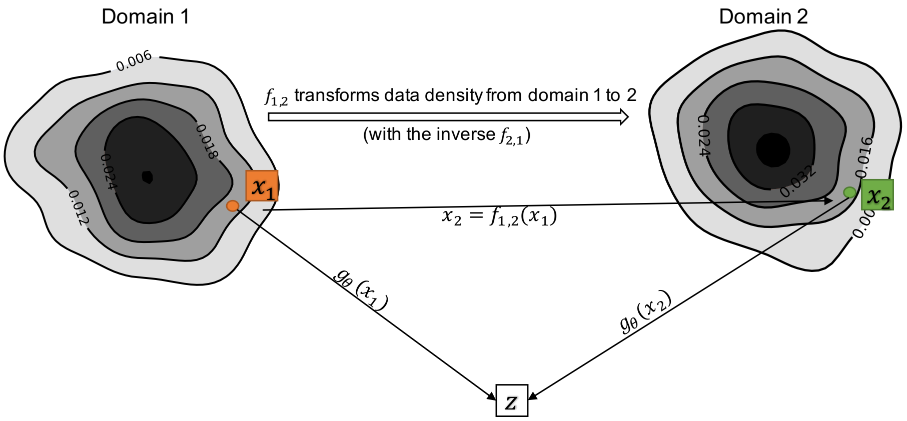


Figure 3. **Domain density transformation**. If we know the function $f_{1,2}$ that transforms the data density from domain 1 to domain 2, we can learn a domain invariant representation network $g_\theta(x)$ by enforcing it to be invariant under $f_{1,2}$, i.e., $g_\theta(x_1) = g_\theta(x_2)$ for any $x_2 = f_{1,2}(x_1)$.

(by applying variable substitution in multiple integral: $x' = f_{d,d'}(x)$)

$$= \int p(x'|d') \left|\det J_{f_{d',d}}(x')\right|^{-1} p(z|x') \left|\det J_{f_{d',d}}(x')\right| \mathrm{d}x'$$

(since $p(f_{d',d}(x')|d) = p(x'|d') \left|\det J_{f_{d',d}}(x')\right|^{-1}$ due to Eq 6 and $p(z|f_{d',d}(x')) = p(z|x')$ due to definition of z in Eq 7)

$$\begin{aligned} &= \int p(x'|d')p(z|x')\mathrm{d}x' \\ &= p(z|d') \end{aligned} \tag{8}$$

ii) Conditional alignment: $\forall z, y$ we have:

$$\begin{aligned} p(z|y,d) &= \int p(x|y,d)p(z|x)\mathrm{d}x \\ &= \int p(f_{d',d}(x')|y,d)p(z|f_{d',d}(x')) \left|\det J_{f_{d',d}}(x')\right| \mathrm{d}x' \end{aligned}$$

(by applying variable substitution in multiple integral: $x' = f_{d,d'}(x)$)

$$= \int p(x'|y,d') \left|\det J_{f_{d',d}}(x')\right|^{-1} p(z|x') \left|\det J_{f_{d',d}}(x')\right| \mathrm{d}x'$$

(since $p(f_{d',d}(x')|y,d) = p(x'|y,d') \left|\det J_{f_{d',d}}(x')\right|^{-1}$ due to Eq 4 and $p(z|f_{d',d}(x')) = p(z|x')$ due to definition of z in Eq 7)

$$\begin{aligned} &= \int p(x'|y,d')p(z|x')\mathrm{d}x' \\ &= p(z|y,d') \end{aligned} \tag{9}$$

Note that

$$p(y|z,d) = \frac{p(y,z|d)}{p(z|d)} = \frac{p(y|d)p(z|y,d)}{p(z|d)} \tag{10}$$

Since $p(y|d) = p(y) = p(y|d'), p(z|y,d) = p(z|y,d')$ and $p(z|d) = p(z|d')$, we have:

$$p(y|z,d) = \frac{p(y|d')p(z|y,d')}{p(z|d')} = p(y|z,d') \tag{11}$$

□

This theorem indicates that, if we can find the functions f's that transform the data densities among the domains, we can learn a domain-invariant representation z by encouraging the representation to be invariant under all the transformations f's. This idea is illustrated in Figure 3. We therefore can use the following learning objective to learn a domain-invariant representation $z = g_\theta(x)$:

$$\mathbb{E}_d \left[\mathbb{E}_{p(x,y|d)} \left[l(y, g_\theta(x)) + \mathbb{E}_{d'}[||g_\theta(x) - g_\theta(f_{d,d'}(x))||_2^2] \right] \right] \tag{12}$$

where $l(y, g_\theta(x))$ is the prediction loss of a network that predicts y given $z = g_\theta(x)$, and the second term is to enforce the invariant condition in Eq 7.

Assume that we have a set of K sources domain $D_s = \{d_1, d_2, ..., d_K\}$, the objective function in Eq. 12 becomes:

$$\mathbb{E}_{d,d' \in D_s, p(x,y|d)} \left[l(y, g_\theta(x)) + ||g_\theta(x) - g_\theta(f_{d,d'}(x))||_2^2 \right] \tag{13}$$

In the next section, we show how one can incorporate this idea into real-world domain generalization problems with generative adversarial networks.

4. Domain Generalization with Generative Adversarial Networks

In practice, we will learn the functions f's that transform the data distributions between domains and one can use several generative modeling frameworks, e.g., normalizing flows (Grover et al., 2020) or GANs (Zhu et al., 2017; Choi et al., 2018; 2020) to learn such functions. One advantage of normalizing flows is that this transformation is naturally invertible by design of the neural network. In addition, the determinant of the Jacobian of that transformation can be efficiently computed. However, due to the fact that we do not need access to the Jacobian when the training process of the generative model is completed, we propose the use of GANs to inherit its rich network capacity. In particular, we use the StarGAN (Choi et al., 2018) model, which is designed for image domain transformations.

The goal of StarGAN is to learn a unified network G that transforms the data density among multiple domains. In particular, the network $G(x, d, d')$ (i.e., G is conditioned on the image x and the two different domains d, d') transforms an image x from domain d to domain d'. Different from the original StarGAN model that only takes the image x and the desired destination domain d' as its input, in our implementation, we feed both the original domain d and desired destination domain d' together with the original image x to the generator G.

The generator's goal is to fool a discriminator D into thinking that the transformed image belongs to the destination domain d'. In other words, the equilibrium state of StarGAN, in which G completely fools D, is when G successfully transforms the data density of the original domain to that of the destination domain. After training, we use $G(., d, d')$ as the function $f_{d,d'}(.)$ described in the previous section and perform the representation learning via the objective function in Eq 13.

Three important loss functions of the StarGAN architecture are:

- Domain classification loss $\mathcal{L}_{cls}$ that encourages the generator G to generate images that correctly belongs to the desired destination domain d'.
- The adversarial loss $\mathcal{L}_{adv}$ that is the classification loss of a discriminator D that tries to distinguish between real images and the fake images generated by G. The equilibrium state of StarGAN is when G completely fools D, which means the distribution of the generated images (via $G(x, d, d'), x \sim p(x|d)$) becomes the distribution of the real images of the destination domain $p(x'|d')$. This is our objective, i.e., to learn a function that transforms domains' densities.
- Reconstruction loss $\mathcal{L}_{rec} = \mathbb{E}_{x,d,d'}[||x - G(x', d', d)||_1]$ where $x' = G(x, d, d')$ to ensure that the transformations preserve the image's content. Note that this also aligns with our interest since we want $G(., d', d)$ to be the inverse of $G(., d, d')$, which will minimize $\mathcal{L}_{rec}$ to zero.

We can enforce the generator G to transform the data distribution within the class y (e.g., $p(x|y, d)$ to $p(x'|y, d')$ $\forall y$) by sampling each minibatch with data from the same class y, so that the discriminator will distinguish the transformed images with the real images from class y and domain d'. However, we found that this constraint can be relaxed in practice, and the generator almost always transforms the image within the original class y.

As mentioned earlier, after training the StarGAN model, we can use the generator $G(., d, d')$ as our $f_{d,d'}(.)$ function and learn a domain-invariant representation via the learning objective in Eq 13. We name this implementation of our method DIR-GAN (domain-invariant representation learning with generative adversarial networks).

5. Experiments

5.1. Datasets

To evaluate our method, we perform experiments in three datasets that are commonly used in the literature for domain generalization.

Rotated MNIST. In this dataset by Ghifary et al. (2015), 1,000 MNIST images (100 per class) (LeCun & Cortes, 2010) are chosen to form the first domain (denoted $\mathcal{M}_0$), then rotations of $15°, 30°, 45°, 60°$ and $75°$ are applied to create five additional domains, denoted $\mathcal{M}_{15}, \mathcal{M}_{30}, \mathcal{M}_{45}, \mathcal{M}_{60}$ and $\mathcal{M}_{75}$. The task is classification with ten classes (digits 0 to 9).

PACS (Li et al., 2017) contains 9,991 images from four different domains: art painting, cartoon, photo, sketch. The task is classification with seven classes.

OfficeHome (Venkateswara et al., 2017) has 15,500 images of daily objects from four domains: art, clipart, product and real. There are 65 classes in this classification dataset.

Table 1. Rotated Mnist leave-one-domain-out experiment. Reported numbers are mean accuracy and standard deviation among 5 runs

	Domains						
Model	$\mathcal{M}_0$	$\mathcal{M}_{15}$	$\mathcal{M}_{30}$	$\mathcal{M}_{45}$	$\mathcal{M}_{60}$	$\mathcal{M}_{75}$	Average
HIR (Wang et al., 2020)	90.34	99.75	99.40	96.17	99.25	91.26	96.03
DIVA (Ilse et al., 2020)	93.5	99.3	99.1	99.2	99.3	93.0	97.2
DGER (Zhao et al., 2020)	90.09	99.24	99.27	99.31	99.45	90.81	96.36
DA (Ganin et al., 2016)	86.7	98.0	97.8	97.4	96.9	89.1	94.3
LG (Shankar et al., 2018)	89.7	97.8	98.0	97.1	96.6	92.1	95.3
HEX (Wang et al., 2019)	90.1	98.9	98.9	98.8	98.3	90.0	95.8
ADV (Wang et al., 2019)	89.9	98.6	98.8	98.7	98.6	90.4	95.2
DIR-GAN (ours)	97.2(±0.3)	99.4(±0.1)	99.3(±0.1)	99.3(±0.1)	99.2(±0.1)	97.1(±0.3)	**98.6**

Table 2. PACS leave-one-domain-out experiment. Reported numbers are mean accuracy and standard deviation among 5 runs

		PACS				
Model	Backbone	Art Painting	Cartoon	Photo	Sketch	Average
DGER (Zhao et al., 2020)	Resnet18	80.70	76.40	96.65	71.77	81.38
JiGen (Carlucci et al., 2019)	Resnet18	79.42	75.25	96.03	71.35	79.14
MLDG (Li et al., 2018a)	Resnet18	79.50	77.30	94.30	71.50	80.70
MetaReg (Balaji et al., 2018)	Resnet18	83.70	77.20	95.50	70.40	81.70
CSD (Piratla et al., 2020)	Resnet18	78.90	75.80	94.10	76.70	81.40
DMG (Chattopadhyay et al., 2020)	Resnet18	76.90	80.38	93.35	75.21	81.46
DIR-GAN (ours)	Resnet18	82.56(± 0.4)	76.37(± 0.3)	95.65(± 0.5)	79.89(± 0.2)	**83.62**

Table 3. OfficeHome leave-one-domain-out experiment. Reported numbers are mean accuracy and standard deviation among 5 runs

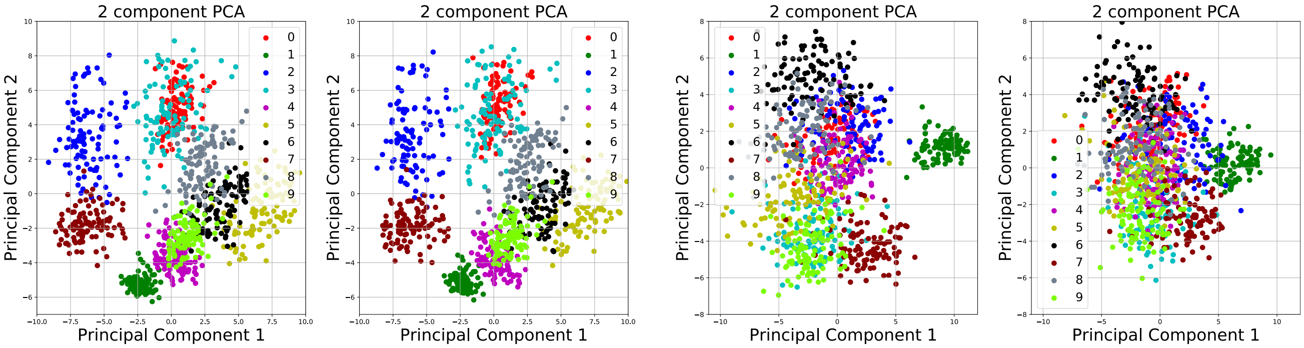
		OfficeHome				
Model	Backbone	Art	ClipArt	Product	Real	Average
D-SAM (D'Innocente & Caputo, 2018)	Resnet18	58.03	44.37	69.22	71.45	60.77
JiGen (Carlucci et al., 2019)	Resnet18	53.04	47.51	71.47	72.79	61.20
DIR-GAN (ours)	Resnet18	56.69(±0.4)	50.49(±0.2)	71.32(±0.4)	74.23(±0.5)	**63.18**

5.2. Experimental Setting

For all datasets, we perform "leave-one-domain-out" experiments, where we choose one domain as the target domain, train the model on all remaining domains and evaluate it on the chosen domain. Following standard practice, we use 90% of available data as training data and 10% as validation data, except for the Rotated MNIST experiment where we do not use a validation set and just report the performance of the last epoch.

For the **Rotated MNIST** dataset, we use a network of two 3x3 convolutional layers and a fully connected layer as the representation network g_θ to get a representation z of 64 dimensions. A single linear layer is then used to map the representation z to the ten output classes. This architecture is the deterministic version of the network used by Ilse et al. (2020). We train our network for 500 epochs with the Adam optimizer (Kingma & Ba, 2014), using the learning rate 0.001 and minibatch size 64, and report performance on the test domain after the last epoch.

For the **PACS** and **OfficeHome** datasets, we use a Resnet18 (He et al., 2016) network as the representation network g_θ. As a standard practice, the Resnet18 backbone is pre-trained on ImageNet. We replace the last fully connected layer of the Resnet with a linear layer of dimensions (512, 256) so that our representation has 256 dimensions. As with the Rotated MNIST experiment, we use a single layer to map from the representation z to the output. We train the network for 100 epochs with plain stochastic gradient descent (SGD) using learning rate 0.001, momentum 0.9, minibatch size 64, and weight decay 0.001. Data augmentation is also standard practice for real-world computer vision datasets like PACS and OfficeHome, and during the training we augment our data as follows: crops of random size and aspect ratio, resizing to 224 × 224 pixels, random horizontal flips, random color jitter, randomly converting the image tile to grayscale with 10% probability, and normalization using the ImageNet channel means and standard deviations.



(a) Domain $\mathcal{M}_{30}$ DIR-GAN (b) Domain $\mathcal{M}_{60}$ DIR-GAN (c) Domain $\mathcal{M}_{30}$ DeepAll (d) Domain $\mathcal{M}_{60}$ DeepAll

Figure 4. **Visualization of the representation space**. Each point indicates a representation z of an image x in the two dimensional space and its color indicates the label y. Two left figures are for our method DIR-GAN and two right figures are for the naive model DeepAll.

The StarGAN (Choi et al., 2018) model implementation is taken from the authors' original source code with no significant modifications. For each set of source domains, we train the StarGAN model for 100,000 iterations with a minibatch of 16 images per iteration.

The code for all of our experiments will be released for reproducibility. Please also refer to the source code for any other architecture and implementation details.

5.3. Results

Rotated MNIST Experiment. Table 1 shows the performance of our model on the Rotated MNIST dataset. The main baselines we consider in this experiment are HIR (Wang et al., 2020), DIVA (Ilse et al., 2020) and DGER (Zhao et al., 2020), which are domain invariance based methods. Our method recognizably outperforms those, illustrating the effectiveness of our method on learning a domain-invariant representation over the existing works. We also include other best-performing models for this dataset in the second half of the table. To the best of our knowledge, we set a new state-of-the-art performance on this Rotated MNIST dataset.

We further analyze the distribution of the representation z by performing principal component analysis to reduce the dimension of z from 64 to two principal components. We visualize the representation space for two domains $\mathcal{M}_{30}$ and $\mathcal{M}_{60}$, with each point indicating the representation z of an image x in the two-dimensional space and its color indicating the label y. Figures 4a and 4b show the representation space of our method (in domains $\mathcal{M}_{30}$ and $\mathcal{M}_{60}$ respectively). It is clear that both the marginal (judged by the general distribution of the points) and the conditional (judged by the positions of colors) are relatively aligned. Meanwhile, Figures 4c and 4d show the representation space with naive training (in domains $\mathcal{M}_{30}$ and $\mathcal{M}_{60}$ respectively), showing the misalignment in the marginal distribution (judged by the general distribution of the points) and the conditional distribution (for example, the distributions of blue points and green points).

PACS and OfficeHome. To the best of our knowledge, domain invariant representation learning methods have not been applied widely and successfully for real-world computer vision datasets (e.g., PACS and OfficeHome) with very deep neural networks such as Resnet, so the only relevant baseline to ours is DGER (Zhao et al., 2020) for the PACS experiment. Therefore, we include more baselines from other approaches (e.g., meta-learning based or domain-specificity based methods) for comparison. Table 2 and 3 show that DIR-GAN outperforms DGER significantly and achieves competitive performance compared to other state-of-the-art baselines.

6. Conclusion

To conclude, in this work we propose a theoretically grounded approach to learn a domain-invariant representation for the domain generalization problem by using domain transformation functions. We also provide some insights into domain-invariant representation learning with several theoretical observations. We then introduce an implementation for our method in practice with the domain transformations learned by a StarGAN architecture and empirically show that our approach outperforms other domain-invariance-based methods. Our method also achieves competitive results on several datasets when compared to other state-of-the-art models. In the future, we plan to incorporate our method into meta-learning based and domain-specificity based approaches for improved performance. We also plan to extend the domain-invariant representation learning framework to the more challenging scenarios, for example, where domain information is not available (i.e., we have a dataset pooled from multiple source domains but do not know the domain identification of each data instance).

References

Akuzawa, K., Iwasawa, Y., and Matsuo, Y. Adversarial invariant feature learning with accuracy constraint for domain generalization. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pp. 315–331. Springer, 2019.

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

Balaji, Y., Sankaranarayanan, S., and Chellappa, R. Metareg: Towards domain generalization using meta-regularization. *Advances in Neural Information Processing Systems*, 31: 998–1008, 2018.

Behl, H., Baydin, A. G., and Torr, P. H. Alpha maml: Adaptive model-agnostic meta-learning. In *6th ICML Workshop on Automated Machine Learning, Thirty-sixth International Conference on Machine Learning (ICML 2019), Long Beach, CA, US*, 2019.

Bogachev, V. I. *Measure theory*, volume 1. Springer Science & Business Media, 2007.

Carlucci, F. M., D'Innocente, A., Bucci, S., Caputo, B., and Tommasi, T. Domain generalization by solving jigsaw puzzles. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 2229–2238, 2019.

Chattopadhyay, P., Balaji, Y., and Hoffman, J. Learning to balance specificity and invariance for in and out of domain generalization. In *European Conference on Computer Vision*, pp. 301–318. Springer, 2020.

Chen, T., Kornblith, S., Norouzi, M., and Hinton, G. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pp. 1597–1607. PMLR, 2020a.

Chen, T., Kornblith, S., Swersky, K., Norouzi, M., and Hinton, G. Big self-supervised models are strong semi-supervised learners. *arXiv preprint arXiv:2006.10029*, 2020b.

Choi, Y., Choi, M., Kim, M., Ha, J.-W., Kim, S., and Choo, J. Stargan: Unified generative adversarial networks for multi-domain image-to-image translation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 8789–8797, 2018.

Choi, Y., Uh, Y., Yoo, J., and Ha, J.-W. Stargan v2: Diverse image synthesis for multiple domains. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 8188–8197, 2020.

Combes, R. T. d., Zhao, H., Wang, Y.-X., and Gordon, G. Domain adaptation with conditional distribution matching and generalized label shift. *arXiv preprint arXiv:2003.04475*, 2020.

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

Du, Y., Xu, J., Xiong, H., Qiu, Q., Zhen, X., Snoek, C. G., and Shao, L. Learning to learn with variational information bottleneck for domain generalization. In *European Conference on Computer Vision*, pp. 200–216. Springer, 2020.

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

Ganin, Y., Ustinova, E., Ajakan, H., Germain, P., Larochelle, H., Laviolette, F., Marchand, M., and Lempitsky, V. Domain-adversarial training of neural networks. *The Journal of Machine Learning Research*, 17(1):2096–2030, 2016.

Ghifary, M., Kleijn, W. B., Zhang, M., and Balduzzi, D. Domain generalization for object recognition with multi-task autoencoders. In *Proceedings of the IEEE International Conference on Computer Vision*, pp. 2551–2559, 2015.

Goodfellow, I. J., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y. Generative adversarial networks. *arXiv preprint arXiv:1406.2661*, 2014.

Grover, A., Chute, C., Shu, R., Cao, Z., and Ermon, S. Alignflow: Cycle consistent learning from multiple domains via normalizing flows. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pp. 4028–4035, 2020.

He, K., Zhang, X., Ren, S., and Sun, J. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 770–778, 2016.

Ilse, M., Tomczak, J. M., Louizos, C., and Welling, M. Diva: Domain invariant variational autoencoders. In *Medical Imaging with Deep Learning*, pp. 322–348. PMLR, 2020.

Khosla, A., Zhou, T., Malisiewicz, T., Efros, A. A., and Torralba, A. Undoing the damage of dataset bias. In *European Conference on Computer Vision*, pp. 158–171. Springer, 2012.

Kingma, D. P. and Ba, J. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

LeCun, Y. and Cortes, C. MNIST handwritten digit database. 2010. URL http://yann.lecun.com/exdb/mnist/.

Li, D., Yang, Y., Song, Y.-Z., and Hospedales, T. Deeper, broader and artier domain generalization. In *International Conference on Computer Vision*, 2017.

Li, D., Yang, Y., Song, Y.-Z., and Hospedales, T. Learning to generalize: Meta-learning for domain generalization. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32, 2018a.

Li, Y., Gong, M., Tian, X., Liu, T., and Tao, D. Domain generalization via conditional invariant representations. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32, 2018b.

Li, Y., Tian, X., Gong, M., Liu, Y., Liu, T., Zhang, K., and Tao, D. Deep domain generalization via conditional invariant adversarial networks. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 624–639, 2018c.

Misra, I. and Maaten, L. v. d. Self-supervised learning of pretext-invariant representations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 6707–6717, 2020.

Muandet, K., Balduzzi, D., and Schölkopf, B. Domain generalization via invariant feature representation. In *International Conference on Machine Learning*, pp. 10–18. PMLR, 2013.

Piratla, V., Netrapalli, P., and Sarawagi, S. Efficient domain generalization via common-specific low-rank decomposition. In *International Conference on Machine Learning*, pp. 7728–7738. PMLR, 2020.

Rudin, W. *Real and complex analysis*. Tata McGraw-hill education, 2006.

Seo, S., Suh, Y., Kim, D., Han, J., and Han, B. Learning to optimize domain specific normalization for domain generalization. *arXiv preprint arXiv:1907.04275*, 3(6):7, 2019.

Shankar, S., Piratla, V., Chakrabarti, S., Chaudhuri, S., Jyothi, P., and Sarawagi, S. Generalizing across domains via cross-gradient training. *arXiv preprint arXiv:1804.10745*, 2018.

Tanwani, A. K. Domain-invariant representation learning for sim-to-real transfer. *arXiv preprint arXiv:2011.07589*, 2020.

Venkateswara, H., Eusebio, J., Chakraborty, S., and Panchanathan, S. Deep hashing network for unsupervised domain adaptation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 5018–5027, 2017.

Wang, H., He, Z., Lipton, Z. C., and Xing, E. P. Learning robust representations by projecting superficial statistics out. *arXiv preprint arXiv:1903.06256*, 2019.

Wang, Z., Loog, M., and van Gemert, J. Respecting domain relations: Hypothesis invariance for domain generalization. *arXiv preprint arXiv:2010.07591*, 2020.

Zhang, Y., Liu, T., Long, M., and Jordan, M. Bridging theory and algorithm for domain adaptation. In *International Conference on Machine Learning*, pp. 7404–7413. PMLR, 2019.

Zhao, H., Des Combes, R. T., Zhang, K., and Gordon, G. On learning invariant representations for domain adaptation. In *International Conference on Machine Learning*, pp. 7523–7532. PMLR, 2019.

Zhao, S., Gong, M., Liu, T., Fu, H., and Tao, D. Domain generalization via entropy regularization. *Advances in Neural Information Processing Systems*, 33, 2020.

Zhu, J.-Y., Park, T., Isola, P., and Efros, A. A. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *Proceedings of the IEEE international conference on computer vision*, pp. 2223–2232, 2017.