Invariant Learning on Domain Generalization with Variable Tasks

Yizhou Jiang, Tianren Zhang, Chongkai Gao, Haichuan Gao, and Feng Chen, *Senior Member, IEEE*

***Abstract*—Out-of-Distribution (OOD) generalization is a hot spot issue that covers all unpredictable distributional shifts in a broad sense, including multiple specific shifting modes in image classification, such as domain generalization, few-shot learning, etc. Among these works, a favored assumption is that a shared invariant representation can be extracted from all distributions, and serves for inference on novel ones. However, when suffering from undesired shifts in complex cases, such invariant assumptions may be violated, resulting in a significant decrease in performance. To analyze more general shifting problems, we propose the Joint Shift problem that decomposes complex distribution shifts into two basic components on domain and task, namely $P(x)$ and $P(y|x)$. Such a scenario with no explicit stable component poses a challenge for the existing invariant learning framework, and we further prove that the potential dependencies between the two distributions in raw data can cause inevitable conflicts in the invariant space, leading to reduced generalization ability. To tackle this problem, we propose Enforced Decorrelation Alignment (EDA) as a data augmentation method, which uses causal intervention to separate and randomly reassemble the domain and task components, and eliminates their internal correlation in a generated pseudo sample space for invariant learning. We demonstrate the ubiquity of Joint Shift in two experimental scenarios with implicit and explicit task variation and show significant effectiveness of the invariant features of EDA on Joint Shift generalization.**

***Index Terms*—Out-of-Distribution Generalization, Domain Shift, Domain Generalization, Invariant Learning.**

I. Introduction

Out-of-distribution (OOD) problems, as a major challenge for modern machine learning, require models to maintain high performance in testing scenarios with interaction on changeable environments, such as medical imaging [1], [2], robotics [3]–[5] and autonomous driving [6], [7]. It calls for resistance to distributional shifts that do not conform with the **i.i.d.** assumption, i.e., all training and test data are assumed to be independent and identically distributed. While the strict characterization of distributional shifts still remains an open problem, research has been extensively conducted in the field of computer vision to formalize some particular shifting modes on images. For example, domain generalization (DG) aims to solve a unitary task regardless of the divergence between training and test distributions. Specifically, it often learns from a set of related datasets with examples about the same task and abstract a universal model that is supposed to be available for novel domains [8], [9]. However, further research [10]–[12] points out that due to the complexity of OOD shifts, the tasks involved in various domains may also differ from each other, so those methods elaborately designed for domain shifts usually perform even worse than vanilla ones when suffering from such undesirable task shifts, limiting their versatility.

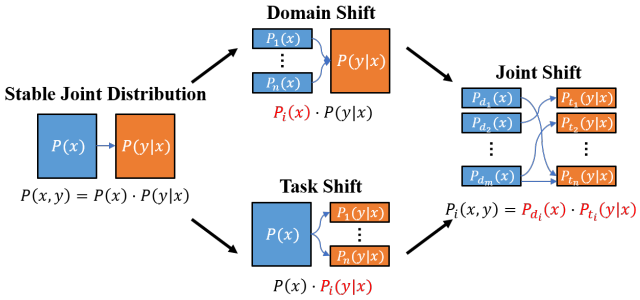

Fig. 1. Paradigms for different distribution shifting modes. Red annotations denote the variable factors of a joint distribution.

To analyze more general OOD scenarios, we formulate the Joint Shift problem, which offers a new perspective to model complex distribution shifts as a combination of two basic shifting modes, as shown in Fig.1. By reinspecting the previous works, we notice that the shift on $P(x)$, known as covariate shifts, is mainly covered in domain generalization [13], [14]. On the other hand, task shift related with $P(y|x)$ has more diversified forms. It can either happen implicitly in multi-domain image classification due to severe label imbalance [15], [16], correlation shift [10], [17], or explicitly in multi-task learning [18] or few-shot learning [12], [19], [20]. Accordingly, we decompose a joint distribution into the two components above: the marginal distribution of input samples referred to **Domain**, and the conditional distribution as **Task**. Their combination provides a sufficient depiction of a joint distribution, thus reflecting all possible shifting cases.

Learning from Joint Shift data poses a challenge to the idea of invariant assumption, which is widely adopted in multi-task learning [21], [22], few-shot learning (FS) [23]–[25] and domain generalization [13], [26], [27]. It essentially suggests that shifting distributions consist of two components: a variable one reflecting the expected shifting mode, and a

This work is supported in part by the National Natural Science Foundation of China under Grant 62176133 and 61836004, and in part by the Tsinghua-Guoqiang research program under Grant 2019GQG0006, and in part by the National Key Research and Development Program of China under Grant 2021ZD0200300. *(Corresponding author: Feng Chen.)*

Y. Jiang, T. Zhang, C. Gao, H. Gao, and F. Chen are with Department of Automation, Tsinghua University, Beijing 100084, China, and also with Beijing Innovation Center for Future Chip, Beijing 100086, China, and also with the LSBDPA Beijing Key Laboratory, Beijing 100084, China. (e-mail: jiangyz20@mails.tsinghua.edu.cn; zhang-tr19@mails.tsinghua.edu.cn; gck20@mails.tsinghua.edu.cn; ghc18@mails.tsinghua.edu.cn; chenfeng@mail.tsinghua.edu.cn)

0000–0000/00$00.00 © 2021 IEEE

shared 'invariant' to represent all remaining factors that are resistant to all distribution changes. For instance, a possible invariant representation z for DG can always meet the condition $P(y|z) = P(y|x)$ in all domains, indicating the shared feature for the same task. However, when neither domain or task is stable in the Joint Shift, the invariant cannot be defined directly to reflect certain semantics in the data. To make matters worse, this problem also cannot be handled as a direct combination of the domain and task variation, because their assumptions of invariant component conflict with each other, leading to instability of the algorithms in practice. We prove that such conflict on invariant can only be avoided if the components of domain and task are fully independent in the raw data distribution, which cannot always be guaranteed in practice.

To address this problem, our idea is to extract the invariant on a pseudo data space where the correlation between domain and task is actively eliminated. Inspired by Independent Causal Mechanisms [28]–[30], we suggest that all shifting joint distributions share the same latent invariant semantic, while domain and task determine its mapping modes to x and y. Our method, Enforced Decorrelation Alignment (EDA), can be regarded as independent interventions on both mapping processes. It explicitly separates the components related to the domain and task from the input data, and realigns them uniformly through random swap to synthesize new samples. Thus, their marginal distributions of domain and task are aligned on a shared pseudo data space to avoid interference correlations in raw data, enabling the invariant extraction. The reassembled data, including novel combinations of sample domains and tasks, can serve as data augmentation to extend the generalization boundary of training data, and the invariant representation can finally be utilized for predictions on unseen distributions.

Our contributions are as follows. **(1)** We propose and analyze the Joint Shift problem to model general distribution variation. **(2)** We develop Enforced Decorrelation Alignment (EDA), an algorithm for invariant learning on joint shifting data for enhanced generalization. **(3)** We demonstrate the impact of Joint Shift and verify the effectiveness of our method on two kinds of experiment scenarios based on DG with implicit and explicit task shifts .

II. Related Work

A. OOD Generalization

Generally, OOD generalization measures a learner's performance beyond training distribution, including all kinds of non-i.i.d. test paradigms. In practice, additional assumptions are often introduced to specify the shifting mode. Some narrow definitions [13], [31] only focus on the marginal distribution shift $P(x)$, while in a broader sense [32], [33], more scenarios are covered, including multi-task, meta-learning, lifelong learning [34], and their combinations [35], [36], etc. In our proposed Joint Shift problem, we focus on the combination of both sample-marginal and sample-conditioned distribution (also referred to co-variate and semantic in [37]), which encourages domain-level and task-level generalization simultaneously.

B. Domain-Level Distribution Shift

Domain is commonly formalized to depict data samples from different domains with co-variate shift, i.e., $P_{train}(X) \neq P_{test}(X)$. To make domain generalization feasible, an assumption on task invariance, $P_{train}(Y|X) = P_{test}(Y|X)$ is most widely adopted [31], [38], [39]. In practice, some modifications are also proposed by introducing an learnable feature extractor $\Phi(X) = Z$, including casual invariance $P_{train}(Y|Z) = P_{test}(Y|Z)$ [17], [40], label-conditioned invariance $P_{train}(Z|Y) = P_{test}(Z|Y)$ [41], [42], etc.

Three branches of implementations are mainly developed. (1) **Invariance Learning**: it is the dominant approach with a direct motivation that the feature independent of seen domains can also perform well on new ones. It attempts to minimize the feature divergence between domains, which can be achieved through regularization metrics [39], adversarial learning [38], [43], or both [16], [41]. (2) **Data Augmentation**: it synthesizes pseudo training samples to enhance data heterogeneity by interpolation between domains, which can be achieved through mixing strategy [44], [45], adversarial gradients [46]–[48], or other specially designed methods [49], [50]. (3) **Multi-task Learning**: it provides a quite distinct perspective by reinterpreting different domains as a family of tasks, so model ensemble [14], [51], [52] and meta-learning [53]–[55] are applied under such settings. On the Joint Shift scenario, our approach mainly integrates invariant learning and data augmentation, while the multi-task methods are not applicable due to the coupled shift in the domain and task. Meanwhile, with the variance of tasks, the existing assumption on invariant has to be modified in our work.

C. Task-Level Distribution Shift

Task-level generalization refers to a series of works that estimate the learners' adaptability on multiple tasks $P_i(Y|X)$, where the task might be defined implicitly in concept drifting [56], or explicitly in multi-task learning [18], meta-learning [57], etc. Both scenarios of task variation are involved in the experiment of our work: implicit task drift caused by class imbalance, and explicit shift on few-shot classification, a supervised meta-learning where tasks are manually defined by several labeled instances.

In recent studies, cross-domain few-shot [12], [36], also called multi-domain few-shot [58], are proposed to extend few-shot learning to data with domain diversity. More specifically, it collects few-shot tasks from different datasets where the class label sets are disjoint, while all examples in a single task are from the same dataset, requiring the learner to extract common features that are applicable to all domains. This is not actually a DG setting, because each category only appears in a single domain. Hence, it lacks the ability to distinguish the same category across various domains. Regarding this point, the Joint Shift problem is an extended case of cross-domain few-shot when the examples in a single task may also come from different domains. The learning machine needs to distinguish them regardless of their domain, leading to better resistance to possible in-task domain shift.

III. Analysis on Shifting Joint Distribution

This section focuses on defining the Joint Shift problem as a general model for shifts in supervised learning. By factorizing a joint distribution into domains and tasks, we decompose all shifts into two corresponding separate components and adopt a composite function family as the learning objective to handle them respectively. In this process, an intermediate variable z is introduced as an interface between domain and task, which is supposed to be independent of all shifting components to ensure the generalization on novel distributions. However, by analyzing current invariant assumptions, we find it impossible to extract such identical invariant z from all distributions, which calls for a new algorithm.

A. Preliminary

The objective in a typical supervised learning task is a mapping function $f : X \to Y$ that can minimize the expected risk on a test distribution $p(x, y)$:

$$f_\theta^* = \arg\min_{f_\theta} \mathbb{E}_{x,y\sim p(x,y)}[\ell(f_\theta(x), y)],$$

where $\ell : \mathcal{Y} \times \mathcal{Y} \to \mathcal{R}$ is the risk function. With the i.i.d. assumption holding true, the Empirical Risk Minimization (ERM):

$$f_\theta^* = \arg\min_{f_\theta} \sum_{i=1}^{n} \ell(f_\theta(x_i), y_i)$$

serves as an approximation to the expected risk, which guarantees the performance when training data $\{x_i, y_i\}$ are sampled from the same distribution as the test environment.

B. Formulation of Joint Shift Problem

In applications, due to the possible distribution shifts on either training or test distributions, the i.i.d. assumption often fails. Throughout this paper, we are concerned with the two basic shifting modes on $P(x)$ and $P(y|x)$, regarded as **domains** and **tasks**. We introduce two latent variables, denoted by d and t, to represent the two changeable components for a more accurate description.

Definition 1 (Joint Distribution Decomposition)**.** *Assuming that tasks and domains are independent, i.e., $t \perp x$ and $d \perp y|x$, a joint distribution can be factorized as two separate components:*

$$P_i(x, y) = P(x, y; d_i, t_i) = P(x; d_i) \cdot P(y|x; t_i). \quad (1)$$

Each $P_i(x, y)$ corresponds to the i-th local i.i.d. sampling, with d_i and t_i encoding its domain and task. The two variables are used to parameterize a joint distribution and decompose the variation into two independent parts. From this, many existing models, as DG and FS, can be regarded as special cases of multivariate distribution shifting: DG supposes a consistent task with different domains $P(x; d_i)$ among all distributions, while FS mainly focuses on the variation on $P(y|x)$ for different t_i. In general, the domain d and task t is provided as additional known conditions or can be inferred from specific examples.

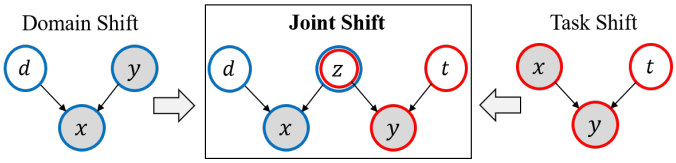

Fig. 2. Relation of components in different shifting models. The unshaded nodes refers to their possible shifting components. In Joint Shift cases, there is no direct and decisive relation between x and y, so z is introduced to depict their internal correlation.

The learning target of Joint Shift is extended to a function family $\mathbb{F}(x; d_i, t_i)$. To further specify the form of the target function modulated by d and t for practical uses, we introduce an extra intermediate variable z to divide the mapping from x to y into two phases: $x \to z$ and $z \to y$, depending only on d and t respectively. Thus, the learning objective can be formally phrased as:

Definition 2 (Joint Shift Problem)**.** *Supervised learning objective for Joint Shift is defined as a composite function family:*

$$\mathbb{F} = \{f_i | f_i = h_{t_i} \circ g_{d_i}\},$$

where $g_{d_i} : X \to Z$, $h_{t_i} : Z \to Y$, s.t.

$$f_i^*(x) = \arg\min_f \{\mathbb{E}_{P_i\sim\mathbb{P}}[\mathbb{E}_{x,y\sim P_i}[\ell(h_{t_i}(g_{d_i}(x)), y)]]\}, \quad (2)$$

where $P_i(x, y) = P(x; d_i) \cdot P(y|x; t_i)$ is sampled from an underlying distribution $\mathbb{P}$.

In this way, the newly introduced variable z is a sufficient representation of x to predict y regardless of the domains or tasks, and any joint distribution $P(x, y)$ can be regarded as a variation of d and t on a distribution of $P(z)$. Besides, in practice, as the input space of x in different domains often shares no intersection, a single mapping function $g : X \to Z$ is sufficient to serve for all domains instead of g_{d_i}. Fig.2 is an intuitive graphical illustration of the Joint Shift problem, which can be seen as a combination of two causal models for domain and task shifts.

This definition is not restricted to the form of domains or tasks in applications. More detailed implementations and the learning procedure will be discussed in Section VI as the experiment settings, where domain d is manually labeled for training and is to be inferred in the test, while the task t can be inferred through a small set of data in few-shot cases.

C. Infeasibility of Invariant Assumptions

Although a general learning framework for joint shifts is proposed, its generalization ability on novel distributions is still not guaranteed, which mainly depends on an appropriate representation of z. In Def.2, z is still indefinite considering diversified decomposition modes of $x \to z \to y$. It is obvious that some trivial solutions of z, e.g., $z = x$ or $z = y$, are of no service to the generalization on shifting d or t. To promote applicability, some current works make further assumptions that z serves as an invariant representation shared by all possible distributions. In other words, they suggest that different input source distributions can be mapped to a shared latent distribution of z to eliminate the variant components in

the raw data where $p(y|z)$ is learned for i.i.d. prediction. As an effect, it enhances the generalization by transforming novel distributions to an invariant space of z.

Nevertheless, we prove that such invariant assumptions are not applicable for joint shifts, as the influence of multivariate shift acts on both x and y that cannot be depicted by a single invariant. Specifically, we will illustrate the dilemma with two practical constraints from domain generalization to show the incompatibility of invariant on multivariate shift.

A most basic assumption is to transform the input x to z with an globally invariant prior $P(z)$, thus mapping each local distribution to an i.i.d. joint distribution $z \times y$ [13], [38], [39]. Nonetheless, such a fixed invariant prior $P(z)$ is not available even if only domain shifts are considered:

Proposition 1. *There does not exist an invariant distribution $P(z)$ such that for any $P_i(x, y)$, $\exists f_i : x \to z$, s.t. $\forall i, P_i(y|f(x)) = P(y|z)$ and $P_i(f(x)) = P(z)$,*

The proof is simple by examining the marginal shift of Y:

Proof. Considering the product of the two equations, we have: $P_i(y, f(x)) = P(y|z) \cdot P(z) = P(y, z)$. The marginal distribution of y in the specific task t is thus derived through $p_i(y) = \int p_i(y, f(x))\, df(x) = \int p(y, z)\, dz = p(y)$. It suggests that y from any source P_i must share a same marginal distribution, i.e. $\forall i,\ p_i(y) = p(y)$, which is not necessarily true under given conditions. □

Thus, this method requires the consistency of $P(y|d)$ for all domains when denoting each P_i by a d_i. As $p(d)$ and $p(y)$ are often known in the training data, it is also equivalent to consistent $P(d|y)$ through Bayes rule, indicating that the distribution of domains should be unrelated with the labels and tasks, which must be ensured by the training data itself or other pre-processing methods as in [15].

To avoid this defect by excluding the influence of $P(y)$, a substitute invariant $P(z|y)$ is proposed [16], [59], where z serves as the key representation of label y that is shared among all domains. But such assumption an can still not adapt to changeable tasks t_i in Joint Shift.

Proposition 2. *There does not exist an invariant posterior distribution $P(z|y)$ such that for any $P_i(x, y)$, $\exists f_i : x \to z$, s.t. $\forall i, P_i(z|y) = P(z|y)$.*

Proof. We denote $P_i(x, y) = P(x, y; t_i)$. A simple example is sufficient to illustrate this problem: consider two different labels, y_1 and y_2 from a distribution $p_1(x, y)$ for task t_1, where $p_1(z|y_1, t_1) \neq p_1(z|y_2, t_1)$. Then we manually define a new task t_2, which denotes the y_1 in t_1 as y_2. so that $p_2(z|y_2, t_2) = p_1(z|y_1, t_1)$. However, for $p_2(x, y)$ corresponding to t_2, we have $p_2(z|y_2, t_2) = p(z|y_2) = p_1(z|y_2, t_1) \neq p_1(z|y_1, t_1)$, which is a contradiction. □

The above example shows an extreme case where different tasks are highly negatively correlated. In multi-task cases, such assumption on invariant representation can only be available when $\forall i, p(z|y, t_i)$ is observed, requiring excessive sampling. It shows that such an explicit invariant is still unable to handle the problem when both components of joint distribution are unstable.

In short, although invariance is vital for OOD generalization, most classic invariant assumptions are inapplicable when suffering from joint shift due to unpredictable correlations of domains and tasks in raw data $P(x, y)$. To solve this problem, we modify the raw distribution to eliminate the correlation between two shifting modes.

IV. Invariance Learning for Joint Shift

This section focuses on constructing an invariant space from joint shift distributions. Since the invariance cannot be directly defined from such data, we propose an approach to recombine the components and generate pseudo data space, so as to ensure an i.i.d. prior $P(z)$. To further form a specific learning algorithm, we construct a probabilistic graphical model to represent the pseudo distribution and maximize its likelihood within a variational framework.

A. Enforced Decorrelation Alignment

As proved in Prop.1, it is unreasonable to suggest a shared invariant $P(z)$ for all distributions due to the possible marginal shift of $P(y)$. Inspired by the causal intervention in [40], we proposed Enforced Decorrelation Alignment (EDA), a data augmentation method, to address this problem. It conducts domain transfer on the raw data and generates pseudo examples to balance the domain for different tasks, so that the marginal $P(y)$ can be aligned for all domains. The correlation between domains and tasks can thus be actively eliminated, ensuring the invariance of $P(z)$.

EDA acts on the feature of each input sample x. For a better illustration of our idea, we divide it into two parts, which is also shown in Fig.4:

Definition 3. *An input sample x can be characterized by a pair of features (a, z). z is a sufficient **semantic** for task, i.e.,$\forall t, P(y|x; t) = P(y|z; t)$. The residual visual characteristics that only related with domains d is denoted as **appearance** a.*

The marginal distributions of appearance and semantic, i.e., $P(a)$ & $P(z)$, are related to the shift on domain $P(x)$ and task $P(y|x)$. Thus, the invariant assumption of $P(z)$ can be rephrased as: $z \perp a|d$. To avoid their dependency, we explicitly separate the two components and reconstruct a regularized pseudo space $P'(a, z) = P(a) \cdot P(z)$ as a substitute for the original feature space $P(a, z)$. In other words, after extracting the appearance and semantic features from input examples, we randomly shuffle and recombine them to generate pseudo examples, and modify other supervised information synchronously. As both features are collected from different inputs, the pseudo data no longer maintain the correlation in original joint distributions, and the marginal distributions of $P(x)$ and $P(y)$ are aligned for all raw distributions. This method not only offers an aligned data space, but also increases the diversity.

The realization of EDA is rather simple, as shown in Fig.3. Within each supervised input batches $\{x_i, d_i\}, i = 1, ..., n$

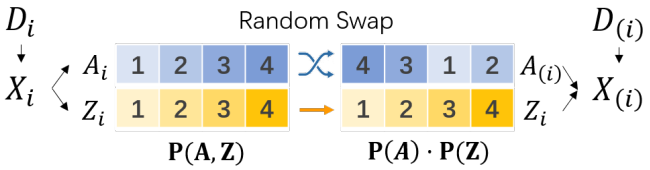

Fig. 3. Schematic diagram of the EDA operation.

from multiple domains, a set of $\{z_i, a_i\}$ can be inferred. We randomly shuffle all a_i into a new order $a_{(i)}$ and record a corresponding $d_{(i)}$. Then we reassemble these latent code and regenerate a batch of new input $x_{(i)} \sim p(z_i, a_{(i)})$, each sharing a same semantic but different domain with the original x_i. The augmented data $\{x_{(i)}, y_i\}$ are then included into training data with the shuffled domain $d_{(i)}$, so that in the augmented data space, z is independent of a and is also an appropriate representation for the raw data:

Proposition 3. *The raw data distribution P_{raw} and the augmented distribution P_{aug} satisfy that: $\forall a, d, z$,*

$$\mathbb{E}_{raw}[H(a|d)] = \mathbb{E}_{aug}[H(a|z,d)] = \mathbb{E}_{aug}[H(a|d)], \tag{3}$$

The first equation implies that the variation of z in the augmentation does not affect the feature extraction on the original data, while the second equation implies that the relevance of domain and task semantic on the augmented space is fully eliminated. From a causal view, the random domain alignment applies intervention on the appearance a for each sample to align a unified marginal distribution $p(z)$ throughout all sampling environments. Considering the counterfactual operator "DO" used in casual learning as applied in [29], [60], the operation of EDA can be seen as interventions on z:

Proof. The swapping procedure of domain d_i and appearance a_i are always synchronized in EDA, so for each domain d_i in the original sample space, the swapping procedure can be regarded as an adjustment on the distribution of z while keeping the appearance a_i unchanged. As $P_{aug}(z)$ is strictly identical to the marginal distribution $P_{raw}(z)$ and is independent of any variables in augmented distribution, we have:

$$\mathbb{E}_{raw}[H(z)] = \mathbb{E}_{aug}[H(z)] = \mathbb{E}_{aug}[H(z|d,a)],$$

and thus:

$$\begin{aligned}
&\mathbb{E}_{raw}[H(a|d)] \\
=&\mathbb{E}_{raw}[H(a|d, DO(z))] \\
=&\mathbb{E}_{aug}[H(a|d,z)] \\
=&\mathbb{E}_{aug}[H(a|d,z)] - \mathbb{E}_{raw}[H(z)] + \mathbb{E}_{raw}[H(z)] \\
=&\mathbb{E}_{aug}[H(a|d,z) - H(z) + H(z|a,d)] \\
=&\mathbb{E}_{aug}[H(a|d)]
\end{aligned} \tag{4}$$

□

This proposition indicates that in the augmented distribution, the independence $z \perp a|d$ is proved to be valid in each domain. Therefore, the existence of invariant prior $P(z)$ is guaranteed.

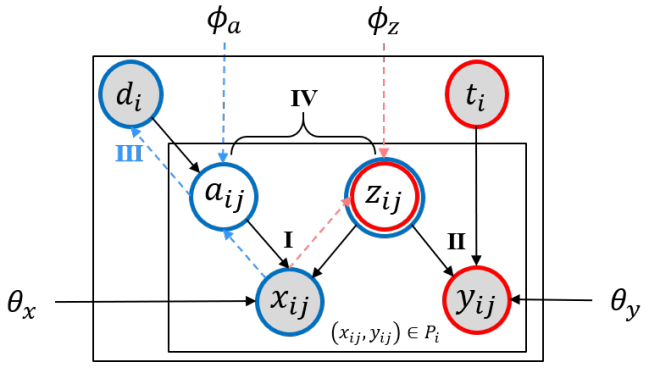

Fig. 4. A graphic model for Joint Shift problems. Solid lines denote the generative model; dashed lines denote the variational approximation. Shaded nodes may explicitly be provided or inferred in training. The index marks refer to items in Equ.5.

B. Graphical Model for Variational Learning

To better illustrate all these auxiliary variables and their relationships, we use a probabilistic graphical model to represent Joint Shift formulation with invariant $P(z)$ ensured by EDA, see Fig.4. As defined in Def.3, for a local data distribution P_i, each input sample x_{ij} is characterized by a semantic z_{ij} for prediction, and a appearance a_{ij} that is sampled from a distribution determined by *domain* d_i. The invariant assumption is reflected through the d-separation property, i.e., $z \perp a|d$, suggesting that z subjects to a global invariant distribution regardless of its sampling source. The output y_{ij} is represented by $p_i(y|z;t)$ with *task* t_i. In few-shot cases, the task t can be explicitly inferred as $\hat{t} = \arg\max_t P(t|x,y)$. Prediction on any assigned tasks is conducted through $p(y|x;t) = \mathbb{E}_z\ p(y|z;t) \cdot p(z|x)$, which requires $p(z|x)$ and $p(y|z;t)$ that can generalize to unseen domains and tasks.

This model is used for representing supervised data with joint shifts, and its parameters can be learned through variational inference, marked by dashed lines in Fig.4. For practical usage, we approximate the posterior $p(a, z|x;\theta_x)$ with a variational posterior $q(a, z|x;\phi)$ which decomposes as the product of $q(a|x;\phi_a)$ and $q(z|x;\phi_z)$, where ϕ_a and ϕ_z are the variational parameters for semantic z and appearance a. Another posterior $q(d|a;\phi_d)$ is used to estimate the probability of domain d for given examples. The log-likelihood for the supervised data in training can thus be derived as follows:

$$\begin{aligned}
&\log p_\theta(x,y,t,d) \\
=&\log p(t) + \log p(x|d) + \log p(y|x,t) + \log p(d) \\
\geq&\ \underbrace{\mathbb{E}_{z\sim q_{\phi_z}(z|x), a\sim q_{\phi_a}(a|x)}[\log p_{\theta_x}(x|z,a)]}_{\textbf{I}\ \ reconstruction} \\
&+ \underbrace{\mathbb{E}_{z\sim q_{\phi_z}(z|x)}[\log p_{\theta_y}(y|z,t)]}_{\textbf{II}\ \ task\ prediction} \\
&+ \underbrace{\mathbb{E}_{a\sim q_{\phi_a}(a|x)}[\log q_{\theta_d}(d|a)]}_{\textbf{III}\ \ domain\ prediction} \\
&- \underbrace{\{2 * KL[q_{\phi_z}(z|x)\|p(z)] + KL[q_{\phi_a}(a|x)\|p(a|d)]\}}_{\textbf{IV}\ \ KL\ divergence} \\
&+ \log p(t) \\
=&\ ELBO + \log p(t)
\end{aligned} \tag{5}$$

where $\log p(t)$ is non-optimiazble.

The ELBO provides a lower bound for the log-likelihood of the joint distributions of observed data, and the whole training dataset can be factorized as the product of ELBO within each local distribution. Thus, the graphic model can be used to represent any supervised joint distributions and for inference on unseen ones with determined tasks.

V. Algorithm Implementation

This section provides a network structure and its corresponding algorithm based on the previous derivation. With a set of neural networks, we implement variational learning on the graphical model of Fig.4. The augmented data of EDA is added to the loss function, and other auxiliary terms as an information bottleneck and cycle consistency loss are also introduced as regularization for enhanced performance.

A. Network Structure

In practice, neural networks are adopted to parameterize both conditional distributions and variational posteriors for maximization of ELBO in Equ.5, as shown in Fig.5. The encoder embeds an input image into a pair of latent variables $[z_i, a_i]$, which is randomly shuffled and recombined through the EDA. Following the classical VAE [61], the original pairs then go through the Decoder to calculate the reconstruction loss, while the recombined pairs generate pseudo examples X_{swap}. The augmented data are sent into Encoder once again to obtain new latent pairs $[z_i', a'_{(i)}]$. All these latent variables serve the corresponding downstream tasks: z and z' for label classification $p(y|z;t)$ under a specific task embedding t, while a and a' for variational posterior of domain $q(d|a)$.

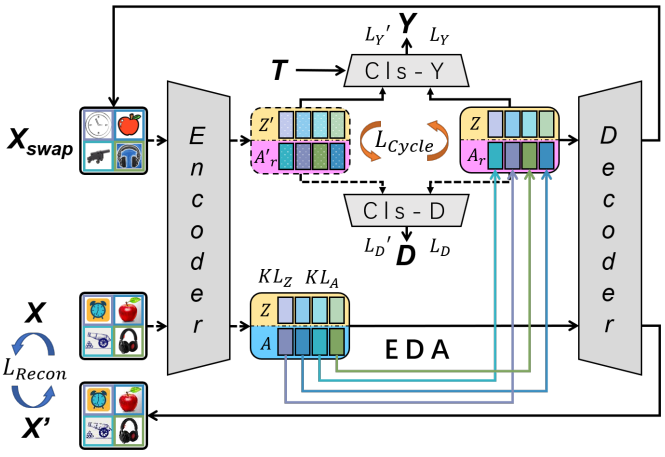

Fig. 5. An overview of our method. Solid and dashed lines denote the generative and variational approximation process respectively. Encoder and Decoder serve for the transformation between variable pairs $[z, a]$ and their corresponding inputs x, while Cls-Y and Cls-D are classifiers to predict labels and domains through z and d. The task code T depends on specific task forms.

A basic loss function directly derived from the ELBO in Equ.5 is as follows:

$$\begin{aligned}&\mathcal{L}_{ELBO}(x, y, d, t)\\ =&\text{BCE}(x, x') + \text{KL}(z) + \text{KL}(a) +\\ &\mu \cdot [\text{CE}(Cls_Y(z), y; t) + \text{CE}(Cls_D(a), d)],\end{aligned} \tag{6}$$

where $[z, a] = Encoder(x), X' = Decoder(z, a)$. CE and BCE refer to the Cross Entropy Loss and Binary Cross Entropy. KL refers to the KL divergence of the variable to a predetermined prior as standard normal distribution.

The regularization term of EDA is obtained by feature recombination as:

$$\mathcal{L}_{EDA} = \text{CE}(Cls_Y(z'), y; t) + \text{CE}(Cls_D(a'), d), \tag{7}$$

where $[z', a'_r] = Encoder(Decoder(z, a_r))$.

For implicit task-shifting scenarios, the task embedding t is assumed to be constant. In explicit multi-task cases as few-shot learning when the task is defined by a set of support examples, we collect their features to represent task t. Practically, we adopt a Feature-Wise Linear Modulation [62]–[64] based method for Cls_D and use the mean values of z on the support set as the modulation vector t.

B. Enhanced Representation of z

Our algorithm enables us to encode the domain, task, and a shared semantic z just by optimizing the marginal likelihood for observed data. However, as [65] has pointed out, due to the unpredictable information allocation, the latent variables may not necessarily include sufficient information as a good representation for all downstream tasks. Additional regularization methods are introduced to alleviate such problems, including information bottleneck and cycle consistency.

During prediction, the inference flow is $x \to z \to y$, where z acts as a sufficient representation for x on estimating y. We expect that z is capable of handling all prediction tasks $p(y|z;t_i)$ as an information bottleneck [66] to enhance its robustness. More specifically, we adopt an modification as:

$$\begin{aligned}&\min_z I(x; z), \quad s.t. \quad I(x; y|z) = 0\\ &Lagrange(z, \beta) = I(x; z) + \beta\{H(y|z) - H(y|x, z)\}\\ =&\mathbb{E}_{x\sim p(x)}[KL(p(z|x)\|p(z))] + \beta\mathbb{E}_{y\sim p(y|z)}[\log p(y|z)]\\ &-\beta H(y|x)\end{aligned}, \tag{8}$$

where β is the Lagrange multiplier, similar to that of β-VAE [67], [68]. Note that the $H(y|x)$ can not be optimized, and the other two items is already calculated in 6, so the bottleneck constraint is eventually rewritten as a penalty:

$$\mathcal{L}_{BTN} = \text{KL}(z) + \text{CE}(Cls_Y(z), y; t), \tag{9}$$

Thus, the information bottleneck on semantic z ameliorates its representation ability.

Further, we introduce additional cycle loss to enhance the reliability of pseudo examples. Unlike real images, the consistency of the EDA augmented images through the autoencoding process cannot benefit from reconstruction loss, which may affect the accuracy of regenerated latent codes z' and a'_r. So the cycle loss is exerted on $[z, a_r]$, as

$$\mathcal{L}_{Cycle} = \text{MSE}(z, z') + \text{MSE}(a_r, a_r'), \tag{10}$$

which can promote the stability and identifiability of the generated images in the early stage of training. The overall loss function is in the form of:

$$\mathcal{L} = \mathcal{L}_{ELBO} + \lambda \cdot \mathcal{L}_{EDA} + \beta \cdot \mathcal{L}_{BTN} + \gamma \cdot \mathcal{L}_{Cycle}. \tag{11}$$

In practice, the weight of regularity terms, $\mathcal{L}_{EDA}$ and $\mathcal{L}_{BTN}$, should be increased in a two-stage training process for improved convergence.

VI. Experiments

In this section, we would like to answer the following questions through experimental evaluation: (1) In what scenarios should Joint Shift on data distribution be considered? (2) Can our method disentangle the joint features of domain and task? (3) Can the learned invariant representation promote generalization on novel distributions? Experiments on two practical cases are conducted as the demonstrations.

A. Datasets and Baselines

Both variations in domains and tasks are to be considered in the Joint Shift problem: the former is related to the dataset, and the latter is determined by the learning scheme.

Domain Shifts: We adopt two multi-domain datasets from popular benchmarks of DG to analyze domain shifts. *Digits-DG* [69] is a digit recognition dataset with 24k examples that consists of 4 widely used datasets, i.e., *MNIST* [70], *MNIST-M* [71], *SVHN* [72] and *SYN* [71]. *Office-Home* [73] is a general object recognition dataset that contains 15.5k images of 65 classes from 4 domains (Art, Clipart, Product, and Real World) in home and office life. We follow the leave-one-domain-out evaluation as [9].

For domain generalization, we choose six representative baselines from two branches: invariance learning and data augmentation. (1) *DeepCoral* [39] proposes a regularization metric to directly match the statistics of all feature distributions; (2) *MMLD* [43] separates domains by clustering and then realigns them through adversarial training; (3) *MMD-AAE* [41] imposes the MMD [74] regularization on adversarial autoencoders to align feature distributions. (4) *MixStyle* [45] conducts linear interpolation as Mixup [44] on feature-level to synthesize novel domains; (5) *CrossGrad* [46] directly perturbs input with adversarial gradients from a domain classifier to eliminate the domain-related appearance; (6) *JiGen* [49] designs a secondary Jigsaw puzzle task to recover each image from its shuffled parts.

Task Shifts: Two scenarios are studied to cover both implicit and explicit task-shifting modes. (1) *Domain Generalization with class imbalance* follows the standard settings of DG with variable label distributions across different domains, leading to implicit task shift; (2) *Domain Generalization Few-Shot Learning* is a combination of DG and FS, which constructs few-shot tasks from multi-domain datasets, where the task is always defined explicitly by a small set of labeled data.

For few-shot learning, we choose three classic methods that correspond to the three main categories of meta-learning: model-based, optimization-based and metric-based, as demonstrative approaches for Joint Shift. (1) *FiLM* [62], [63] designs a widely used conditional learning layer for multi-tasks. (2) *MAML* [75] develops a meta-optimization to initialize a base model to fit all tasks with a few gradient steps. (3) *Relation Network* [76] uses a learnable deep metric to measure the distance between examples. These FS approaches are combined with DG methods or simply vanilla ERM, serving as baselines for our second scenario. We selectively conduct 10 joint baselines in total, as some DG methods can suffer from severe degradation when applied to few-shot environments.

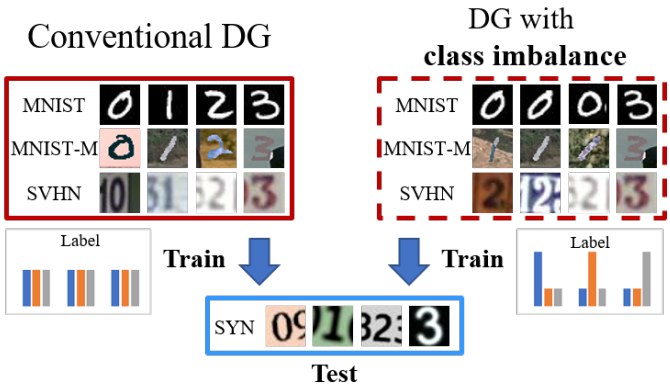

Fig. 6. An illustrative comparison of the training data in conventional DG and DG with class imbalance.

B. Domain Generalization with class imbalance

In a standard domain generalization problem, the proportion of categories can differ significantly across all domains, see Fig.6. As analyzed in [15], [16], although a single $P(y|x)$ is still sufficient for such a case theoretically, it conflicts with the causal structures in real-world images [77], leading to weak generalization ability. By adjusting the unbalance degree of class labels, we conduct three experiments with different extent of implicit task variation and measure the resistance of algorithms. The maximum KL-divergence of $P(y)$ between domains is used to quantify the extent of domain imbalance.

Implementations: For Digits-DG, all algorithms are based on the same backbone with 3 convolution layers ($channels = 32, 64, 64$, $kernel\ size = 4$) and 3 linear layers ($channels = 128, 64, 10$) with other settings similar to that in [69]. More detailed settings of our method is provided in Table.I, and its decoder structure is exactly symmetrical to the encoder. The training process is divided into two stages: for the first stage, only $\mathcal{L}_{ELBO}$ and $\mathcal{L}_{BTN}$ are involved in the loss function for better initial features, and the EDA augmentation is then employed to disentangle them. For Office-Home, we follow the implementation of [8] but replace the backbone with a smaller version, ResNet18 [78], which is more widely used in other baselines. To obtain a corresponding decoder with

TABLE I
Details of network structures and hyper-parameters.

		Digits-DG	Office-Home
Encoder Backbone		Conv (32-64-64) Linear 192	ResNet-18 without FC
Reparameterization	A	Linear (128-64)	Linear (256-128-128)
	Z	Linear (64-64)	Linear (128-128-128)
Classifier	Y	Linear (64-32-10)	Linear (128-256-256-65)
	D	Linear (64-32-3)	Linear (128-128-3)
Loss parameter	Stage 1	$\mu = 1.0E5$, $\lambda = 0$ $\beta = 0.5$, $\gamma = 0$	$\mu = 1.0E4$, $\lambda = 0$ $\beta = 0.5$, $\gamma = 0$
	Switch point	Epoch = 20	Epoch = 5
	Stage 2	$\mu = 1.0E5$, $\lambda = 5.0E4$ $\beta = 0.75$, $\gamma = 100$	$\mu = 1.0E4$, $\lambda = 100$ $\beta = 1$, $\gamma = 1$
Optimizer	AutoEncoder	Adam, lr=0.001	Adam, lr=0.005
	Classifier	Adam, lr=0.001	Adam, lr=0.01

TABLE II
RESULTS ON STANDARD DIGITS-DG ($KL = 0$) FOR DG.

	MNIST	MNIST-M	SVHN	SYN	Average
ERM	95.0	58.8	61.8	78.8	73.6
DeepCoral	95.3	**61.6**	61.4	80.0	74.6
MMLD	94.4	55.3	61.3	**91.5**	75.6
MMD-AAE	95.5	59.7	65.2	77.9	74.6
MixStyle	95.2	50.4	**71.9**	**91.7**	**77.3**
CrossGrad	95.9	59.8	68.1	80.3	76.0
JiGen	**96.5**	60.9	63.7	74.2	73.8
EDA	**95.7**	**61.4**	**69.1**	83.5	**77.4**

TABLE III
RESULTS ON IMBALANCED DIGITS-DG ($KL \approx 0.659$) FOR DG.

	MNIST	MNIST-M	SVHN	SYN	Average
ERM	90.1	51.9	58.4	61.8	65.6
DeepCoral	87.3	**58.9**	53.3	**63.3**	**65.7**
MMLD	66.3	39.7	48.1	55.4	52.4
MMD-AAE	80.8	47.1	56.7	62.4	61.8
MixStyle	85.3	48.9	**61.8**	62.0	64.5
CrossGrad	76.4	48.7	53.0	57.6	58.9
JiGen	**90.8**	50.1	56.8	58.1	64.0
EDA	**91.5**	**60.4**	**66.1**	**66.8**	**71.2**
DGER	92.5	57.3	62.1	65.4	69.3

good initial pretrained features, we adopt the upsampling-convolution structure proposed in [79] to construct the corresponding ResNet-18 decoder and initialize it by unsupervised training on the training domains together with a frozen encoder pre-trained on ImageNet. All hyper-parameters are selected based on the performance on the source validation set, as adopted in [8], [80].

Results on standard Digits-DG: The first experiment is set on Digits-DG where the labels of digits 0-9 are uniformly distributed in all domains, with $KL = 0$. From the results shown in Table.II, we find that for such a simple problem, most DG algorithms, including our EDA, cannot make significant improvement beyond the ERM baseline. Our algorithm achieves competitive performance with a slightly higher average accuracy; but does not achieve the best in any single test item. One reason is that in this case, due to the stability of $P(y)$, the task $P(y|x)$ can almost be regarded as the same, so only shifts on domains need to be considered.

Results on imbalanced Digits-DG: To study the impact of implicit task shifts, we retain three disjoint classes in each training domain and reduce the number of examples in the rest categories to 1/3. Thus, there is a high correlation between domains and label distributions with $KL \approx 0.659$. The results are reported in Table.III. Due to the decrease in sample size, the accuracy of vanilla ERM reduces by 8.1%, and the other baselines for DG suffer more severe performance loss from 8.9%-21.1%. It shows that an implicit Joint Shift problem does exist in this scenario, and the classic methods of DG can be extremely fragile when faced with such task shift. In comparison, the average accuracy of EDA only drops by 6.2%, indicating a strong resistance to imbalanced data. An additional baseline DGER [16] is also conducted with manual label balancing required, but its performance is still slightly inferior to ours. From this experiment, we conclude that the variation on task should not be ignored even in conventional DG settings.

TABLE IV
RESULTS ON OFFICE-HOME ($KL \approx 0.029$) FOR DG.

	Art	Clipart	Product	RealWorld	Average
ERM	58.7	49.3	74.3	76.1	64.6
DeepCoral	**61.6**	49.0	74.2	**76.4**	**65.3**
MMLD	56.6	**51.1**	72.2	71.7	62.9
MMD-AAE	57.3	47.2	71.8	75.1	62.9
MixStyle	61.5	47.3	**74.5**	75.3	64.7
CrossGrad	58.4	49.4	73.9	75.8	64.4
JiGen	53.0	47.5	71.5	72.8	61.2
EDA	**62.2**	**50.1**	**76.7**	**80.5**	**67.4**

Results on Office-Home: Slight divergence of label distribution across domains exists widely in real-world environments, such as the Office-Home dataset with $KL \approx 0.029$. From Table.IV, the performance is roughly similar to that in Digits-DG. EDA achieves more evident improvement over all other baselines in all test items except MMLD for Clipart. Especially, it shows better performance on test domains with more diversified textures, which may indicate a strong ability on reconstructing the key information from complicated data. This experiment proves that our method also works well in more practical application scenarios where Joint Shift exists implicitly.

Ablation Study and Visualization: To better evaluate the ability of EDA on disentangling joint features, we focus on the imbalanced Digits-DG setting for ablation analysis on the loss function Equ.11 in Table.V. Pure ELBO works even worse than vanilla ERM, as its self-supervised features may over-fit the training domains. The performance improvement is mainly from EDA, with other terms playing auxiliary roles. We further conduct visualization on a and z feature by Spectral Embedding [81], a feature extraction algorithm based on clustering, see Fig.7. The feature of pure ELBO shows obvious relevance between a and z, indicating that different components are mixed together in both features. In contrast, the EDA method can better disentangle the semantics of the domain and task spontaneously, thus ensuring the independence hypothesis $a \perp z|d$.

The feature disentanglement is also verified by the augmentation effect of EDA, see Fig.8, where images are generated by EDA with assigned appearances and labels. The resulting effect is similar to style transfer [82], [83], where an image can be rendered into multiple domains while remaining the same

TABLE V
ABLATION STUDY FOR EVALUATING COMPONENTS OF THE LOSS FUNCTION.

	MNIST	MNIST-M	SVHN	SYN	Average
ELBO Only	92.2	56.5	65.4	73.0	71.8
+EDA	**96.3**	57.8	66.3	79.1	74.9
+EDA+BN	94.1	59.7	60.1	81.3	73.8
+EDA+CYCLE	95.9	**60.3**	**68.7**	**82.4**	**76.8**
+EDA+BN+CYCLE	**95.7**	**61.4**	**69.1**	**83.5**	**77.4**

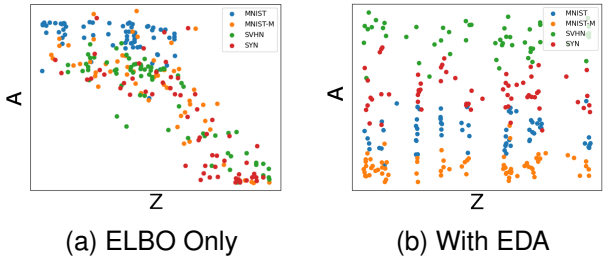

Fig. 7. Visualization of features a and z from imbalanced DIGIT-DG. Each axes correspond to the first principal component to a and z respectively.

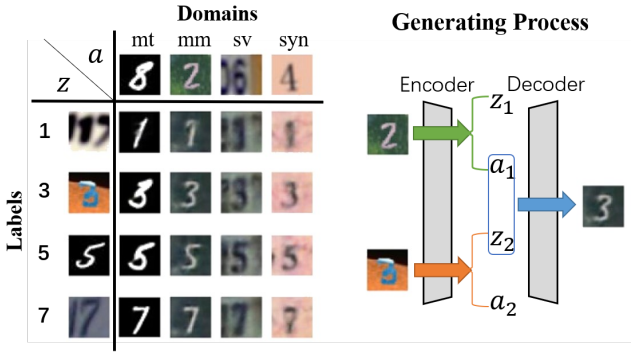

Fig. 8. The generating process of EDA and its augmentation results. Images in the same row share a same semantic z, but hold different a according to their column.

semantic contents. However, our appearance contains more visual information, including the font, stroke and rotation, which often remain unchanged in typical style transfer approaches, so our method can further exclude the appearance contained in z. Besides, the cycle consistency loss $\mathcal{L}_{Cycle}$ is vital for image quality due to the lack of direct supervision on the generated pseudo examples.

C. Domain Generalization Few-Shot Learning

In this section, we consider the possible domain shift on few-shot learning as an example for Joint Shifts with explicit task definitions. We define Domain Generalization Few-Shot Learning as a combination of DG and FS, where examples from different domains can be arbitrarily mixed and collected for few-shot learning. A typical few-shot meta-task, known as N-way K-shot, is defined by a *support* set of $K \cdot N$ examples from N classes, each with K examples, and a *query* set that is sampled from the same N classes as test data. The goal is to carry out a classification on the query set according to the support set regardless of the domains. In our experiment, the training data can contain examples from multiple domains, and in test sessions, three main generalization modes are covered, see Fig.9. *Support Shift* represents the domain variation of the support set, indicating the shift on task definition, while *Query Shift* corresponds to the domain generalization on the query set, indicating the robustness of the few-shot classifier. Besides, with additional assumptions that the query and support set are always collected from the same domain, the problem setting turns into a special mode of *Cross Domain* few-shot that is discussed as another paradigm.

Implementations: Most methods for few-shot learning still work on such problems regardless of the domain shifts,

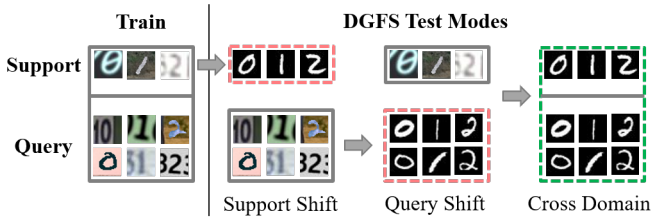

Fig. 9. The three Domain Generalization test modes in Few-Shot Learning.

and can also be combined with other domain generalization approaches. For our method, we simply insert a single conditioning layer FiLM [62] before the label classifier Cls_Y, and the feature of support examples is concatenated as a task representation. Meanwhile, in order to avoid the interference of frequent task changes on feature extraction in the initial stages of training, the unsupervised pre-training of autoencoder in Sec.VI-B for Office-Home is also adopted to Digits-DG. Other implementation details refer to the Task Conditioning in [63]. For other DG baselines, we insert FiLM before their multi-layer classifier for modulation. Besides, we also test two other classical FS methods, MAML [75] and Relation Network [76], and combine them with MMD or MixStyle to align or mix their intermediate features, in hope of enhancing their resistance to domain shift. Pre-trained features are also adopted for all baselines if possible. For a more clear analysis on test result, different from the classic few-shot test scenario, we follow the in-distribution setting proposed in [84], which does not require disjoint sets of training and test classes for task generation, thus ignoring the interference caused by novel test classes. All numerical results are based on the setting of 5-way 5-shot.

Results and Analysis: The main results on Support Shift and Query Shift are shown in Table.VI for Digits-DG and Table.VII for Office-Home with similar performance for each algorithm. With concerns about the disentangle difficulty caused by Joint Shift, our method significantly outperforms other baselines in most test items. Conclusions are made for explicit joint shifts from the perspectives of FS and DG.

(1) It is hard for the existing FS algorithms to handle distinct domain differences within a single training meta-task, resulting in abnormally poor performance. In severe cases, the algorithm may not even converge in training, as in the Relation Network for MNIST test. Such defects can exist on either model-based or metric-based methods, probably because that too many domain-related features are encoded in the task modeling, so feature extraction and task representation cannot be well separated. Ideally, the embedding module in those approaches should extract domain-invariant representations regardless of the task shifts, which is achieved by the forced style transfer of EDA. Besides, we find that in Joint Shift scenarios, optimization-based algorithms such as MAML are relatively weak, due to their theoretical requirements on consistent task space. The Relation Network particularly outperforms other methods on those test domains with relatively simple textures, such as SYN and Clipart.

(2) The combinations of DG and FS methods often result

TABLE VI
RESULTS ON DIGITS-DG FOR DOMAIN GENERALIZATION FEW-SHOT LEARNING.

FS	DG	Support Shift					Query Shift				
		MNIST	MNIST-M	SVHN	SYN	Average	MNIST	MNIST-M	SVHN	SYN	Average
FiLM	Vanilla	82.5	80.4	87.9	84.4	83.8	**89.4**	80.0	72.8	75.3	**79.4**
	MMD	85.2	79.4	90.1	82.5	84.3	88.7	81.1	75.7	72.2	79.4
	MixStyle	**87.9**	68.6	84.3	86.4	81.8	84.8	57.4	**76.8**	75.6	73.7
	CrossGrad	79.3	62.8	79.7	81.6	75.9	80.7	58.6	64.3	62.1	66.4
MAML	Vanilla	83.3	73.4	77.1	76.6	77.6	87.2	60.1	68.7	74.7	72.7
	MMD	82.2	67.8	78.3	74.5	75.7	84.3	69.5	71.3	69.6	73.7
	MixStyle	84.3	56.4	81.2	86.1	77.0	86.7	66.4	73.6	78.1	76.2
Relation Network	Vanilla	25.3	**83.5**	90.6	**90.7**	72.5	22.6	**81.6**	64.8	75.6	61.2
	MMD	23.8	78.4	88.7	74.7	66.4	21.7	81.1	64.4	**79.9**	61.8
	MixStyle	81.2	80.1	**91.4**	**92.3**	**86.3**	77.1	76.5	67.1	**81.2**	75.5
EDA		**90.2**	**89.1**	**93.5**	87.8	**90.2**	**89.4**	**88.7**	**77.1**	78.9	**83.5**

TABLE VII
RESULTS ON OFFICE-HOME FOR DOMAIN GENERALIZATION FEW-SHOT LEARNING.

FS	DG	Support Shift					Query Shift				
		Art	Clipart	Product	Real World	Average	Art	Clipart	Product	Real World	Average
FiLM	Vanilla	68.7	62.3	74.2	78.7	71.0	56.5	51.7	70.1	77.3	63.9
	MMD	66.9	59.8	71.2	78.1	69.0	53.2	48.1	64.6	74.2	60.0
	MixStyle	71.5	60.8	**76.1**	78.3	**71.7**	**64.4**	51.5	**72.0**	72.1	**65.0**
	CrossGrad	60.2	50.6	70.1	71.2	63.0	57.7	52.2	68.8	76.4	63.8
MAML	Vanilla	62.1	58.4	69.7	80.1	67.6	52.2	49.3	71.1	79.2	63.0
	MMD	57.9	54.3	70.4	**82.1**	66.2	47.9	45.1	68.7	**79.6**	60.3
	MixStyle	70.1	51.3	67.8	74.3	65.9	53.5	48.9	72.2	72.6	61.8
Relation Network	Vanilla	68.7	62.3	74.2	78.7	71.0	56.5	51.7	70.1	77.3	63.9
	MMD	67.3	**67.1**	70.8	77.6	70.7	48.7	**54.7**	63.8	73.4	60.2
	MixStyle	**76.2**	52.2	75.4	76.4	70.1	57.2	47.6	70.9	74.4	62.5
EDA		**78.5**	**67.6**	**80.3**	**85.4**	**78.0**	**67.1**	**53.5**	**79.4**	**85.2**	**71.3**

in even worse performance than vanilla FS baselines, which verifies our previous theoretical analysis. The frequent task shifts in FS training will lead to the inconsistent semantic representation of examples, which greatly hinders the cross-domain alignment in the DG algorithm. Nevertheless, some data augmentation based on prior designs like MixStyle are less affected, and can still assist in narrowing domain gaps. Therefore, they are suitable to be combined with FS. Our EDA utilizes augmented data to eliminate the potential correlation between domains and tasks and can thus align different domain distributions regardless of their corresponding tasks.

Comparison to Cross-Domain Few-Shot: Cross-Domain Few-Shot learning problem [12] is an emerging paradigm for few-shot learning, where all examples in a single meta-task must be collected from the same dataset, i.e., a pair of support and query set always share the same domain, while multiple domains can exist in different tasks, making it more like a domain adaptation implementation by adapting to the support set. It is not a typical form of Joint Shift, as each task is always directly bound to a specific domain in such settings. At this cost, learners in this paradigm lack the ability to distinguish the same category across domains. In Table.VIII, we conduct a most representative baseline, ProtoNet + FWT from BSCD-FSL benchmark [36] on Digits-DG and compare it with other basic few-shot approaches. Due to its special prior on the shared domain, its average accuracy is significantly higher than all other methods. However, it is not applicable to all other shifting modes as Support Shift or Query Shift. The metric learned by Relation Network can still work well on this problem, while our EDA is unable to collect features across domains within a single meta-task, resulting in poor performance.

TABLE VIII
COMPARISON OF METHODS UNDER THE SETTING OF CROSS-DOMAIN FEW-SHOT.

	MNIST	MNIST-M	SVHN	SYN	Average
Film Based	90.3	59.8	64.6	74.4	72.3
MAML	91.5	56.4	53.5	79.3	70.2
RelationNet	**93.8**	**63.1**	71.2	**89.4**	**79.4**
EDA	93.3	**65.8**	**71.5**	76.4	76.8
ProtoNet+FWT (With Prior)	**97.8**	59.3	**79.5**	**92.7**	**82.3**

VII. CONCLUSION AND DISCUSSION

This work is a tentative attempt towards handling all distribution shifts in arbitrary OOD cases, where a most ideal scenario is that the learning machine can learn certain knowledge from any supervised joint data pairs $\{x_i, y_i\}$ regardless of the distribution they subject to, and quickly adapt to any assigned new tasks. To analyze this problem, we consider the general decomposition of domain and task to transform the shift on the joint distribution into classical forms that we are familiar with.

Based on this, we proposed the Joint Shift problem, a new scenario to model such cases when domain and task variation is known, which poses a challenge to the existing invariance hypothesis. The invariant representation learned by our method is supposed to resist complex domain shifts and be utilized for various task modes.

Despite all this, data from the open-world can always be more changeable and less annotated, bringing about more problems. Can such a model on joint distribution be extended to semi-supervised learning? Can different tasks and domains involved in data be discovered and defined automatically? In future works, we will further strive to extend the practical forms of domain and task in our Joint Shift model, and apply it to more changeable scenarios with less prior knowledge on distribution shifts.

REFERENCES

[1] Haoliang Li, Yufei Wang, Renjie Wan, Shiqi Wang, Tie-Qiang Li, and Alex Kot. Domain Generalization for Medical Imaging Classification with Linear-Dependency Regularization. In *Advances in Neural Information Processing Systems*, volume 33, pages 3118–3129, 2020.

[2] Pulkit Khandelwal and Paul Yushkevich. Domain Generalizer: A Few-Shot Meta Learning Framework for Domain Generalization in Medical Imaging. In *Domain Adaptation and Representation Transfer, and Distributed and Collaborative Learning*, pages 73–84, 2020.

[3] Josh Tobin, Rachel Fong, Alex Ray, Jonas Schneider, Wojciech Zaremba, and Pieter Abbeel. Domain randomization for transferring deep neural networks from simulation to the real world. In *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 23–30, 2017.

[4] Konstantinos Bousmalis, Alex Irpan, Paul Wohlhart, Yunfei Bai, Matthew Kelcey, Mrinal Kalakrishnan, Laura Downs, Julian Ibarz, Peter Pastor, Kurt Konolige, Sergey Levine, and Vincent Vanhoucke. Using Simulation and Domain Adaptation to Improve Efficiency of Deep Robotic Grasping. In *2018 IEEE International Conference on Robotics and Automation*, pages 4243–4250, 2018.

[5] Haichuan Gao, Zhile Yang, Xin Su, Tian Tan, and Feng Chen. Adaptability Preserving Domain Decomposition for Stabilizing Sim2Real Reinforcement Learning. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 4403–4410, 2020.

[6] Julia Nitsch, Masha Itkina, Ransalu Senanayake, Juan Nieto, Max Schmidt, Roland Siegwart, Mykel J Kochenderfer, and Cesar Cadena. Out-of-distribution Detection for Automotive Perception. In *2021 IEEE International Intelligent Transportation Systems Conference*, pages 2938–2943, 2021.

[7] Angelos Filos, Panagiotis Tigkas, Rowan Mcallister, Nicholas Rhinehart, Sergey Levine, and Yarin Gal. Can Autonomous Vehicles Identify, Recover From, and Adapt to Distribution Shifts? In *Proceedings of the 37th International Conference on Machine Learning*, pages 3145–3153. PMLR, 2020.

[8] Ishaan Gulrajani and David Lopez-Paz. In Search of Lost Domain Generalization. *arXiv:2007.01434 [cs, stat]*, 2020.

[9] 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.

[10] Haoyue Bai, Rui Sun, Lanqing Hong, Fengwei Zhou, Nanyang Ye, Han-Jia Ye, SH Chan, and Zhenguo Li. Decaug: Out-of-distribution generalization via decomposed feature representation and semantic augmentation. In *AAAI Conference on Artificial Intelligence*, page 6705, 2021.

[11] Muhammad Ghifary, W. Bastiaan Kleijn, Mengjie Zhang, and David Balduzzi. Domain Generalization for Object Recognition With Multi-Task Autoencoders. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2551–2559, 2015.

[12] Hung-Yu Tseng, Hsin-Ying Lee, Jia-Bin Huang, and Ming-Hsuan Yang. Cross-Domain Few-Shot Classification via Learned Feature-Wise Transformation. 2020.

[13] Krikamol Muandet, David Balduzzi, and Bernhard Schölkopf. Domain Generalization via Invariant Feature Representation. In *Proceedings of the 30th International Conference on Machine Learning*, pages 10–18. PMLR, 2013.

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

[15] Ya Li, Xinmei Tian, Mingming Gong, Yajing Liu, Tongliang Liu, Kun Zhang, and Dacheng Tao. Deep Domain Generalization via Conditional Invariant Adversarial Networks. In *Proceedings of the European Conference on Computer Vision*, pages 624–639, 2018.

[16] Shanshan Zhao, Mingming Gong, Tongliang Liu, Huan Fu, and Dacheng Tao. Domain Generalization via Entropy Regularization. In *Advances in Neural Information Processing Systems*, volume 33, pages 16096–16107, 2020.

[17] Martin Arjovsky, Léon Bottou, Ishaan Gulrajani, and David Lopez-Paz. Invariant Risk Minimization. *arXiv preprint arXiv:1907.02893*, 2019.

[18] Rich Caruana. Multitask learning. *Machine learning*, 28(1):41–75, 1997.

[19] John Cai and Sheng Mei Shen. Cross-Domain Few-Shot Learning with Meta Fine-Tuning. 2020.

[20] An Zhao, Mingyu Ding, Zhiwu Lu, Tao Xiang, Yulei Niu, Jiechao Guan, and Ji-Rong Wen. Domain-Adaptive Few-Shot Learning. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 1390–1399, 2021.

[21] Amy Zhang, Clare Lyle, Shagun Sodhani, Angelos Filos, Marta Kwiatkowska, Joelle Pineau, Yarin Gal, and Doina Precup. Invariant Causal Prediction for Block MDPs. In *International Conference on Machine Learning*, pages 11214–11224. PMLR, 2020.

[22] Virginia Aglietti, Theodoros Damoulas, Mauricio Álvarez, and Javier González. Multi-task Causal Learning with Gaussian Processes. *Advances in Neural Information Processing Systems*, 33:6293–6304, 2020.

[23] Jingjing Li, Mengmeng Jing, Ke Lu, Zhengming Ding, Lei Zhu, and Zi Huang. Leveraging the invariant side of generative zero-shot learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7402–7411, 2019.

[24] Takeshi Teshima, Issei Sato, and Masashi Sugiyama. Few-shot domain adaptation by causal mechanism transfer. In *International Conference on Machine Learning*, pages 9458–9469. PMLR, 2020.

[25] Zhongqi Yue, Hanwang Zhang, Qianru Sun, and Xian-Sheng Hua. Interventional few-shot learning. *Advances in neural information processing systems*, 33:2734–2746, 2020.

[26] Mahsa Baktashmotlagh, Mehrtash T. Harandi, Brian C. Lovell, and Mathieu Salzmann. Unsupervised Domain Adaptation by Domain Invariant Projection. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 769–776, 2013.

[27] Maximilian Ilse, Jakub M. Tomczak, Christos Louizos, and Max Welling. DIVA: Domain Invariant Variational Autoencoders. In *Proceedings of the Third Conference on Medical Imaging with Deep Learning*, pages 322–348. PMLR, 2020.

[28] Jonas Peters, Dominik Janzing, and Bernhard Schölkopf. *Elements of causal inference: foundations and learning algorithms*. The MIT Press, 2017.

[29] Jovana Mitrovic, Brian McWilliams, Jacob C. Walker, Lars Holger Buesing, and Charles Blundell. Representation Learning via Invariant Causal Mechanisms. In *International Conference on Learning Representations*, 2020.

[30] Mengyue Yang, Furui Liu, Zhitang Chen, Xinwei Shen, Jianye Hao, and Jun Wang. CausalVAE: Disentangled Representation Learning via Neural Structural Causal Models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9593–9602, 2021.

[31] Zheyan Shen, Jiashuo Liu, Yue He, Xingxuan Zhang, Renzhe Xu, Han Yu, and Peng Cui. Towards Out-Of-Distribution Generalization: A Survey. *arXiv preprint arXiv:2108.13624*, 2021.

[32] Haitham Bou Ammar, Eric Eaton, José Marcio Luna, and Paul Ruvolo. Autonomous cross-domain knowledge transfer in lifelong policy gradient reinforcement learning. In *Twenty-fourth international joint conference on artificial intelligence*, 2015.

[33] Ali Geisa, Ronak Mehta, Hayden S. Helm, Jayanta Dey, Eric Eaton, Jeffery Dick, Carey E. Priebe, and Joshua T. Vogelstein. Towards a theory of out-of-distribution learning. *arXiv preprint arXiv:2109.14501*, 2021.

[34] Sebastian Thrun. Is learning the n-th thing any easier than learning the first? *Advances in neural information processing systems*, 1995.

[35] Chelsea Finn, Aravind Rajeswaran, Sham Kakade, and Sergey Levine. Online meta-learning. In *International Conference on Machine Learning*, pages 1920–1930. PMLR, 2019.

[36] Yunhui Guo, Noel C Codella, Leonid Karlinsky, James V Codella, John R Smith, Kate Saenko, Tajana Rosing, and Rogerio Feris. A broader study of cross-domain few-shot learning. In *European Conference on Computer Vision*, pages 124–141. Springer, 2020.

[37] Jingkang Yang, Kaiyang Zhou, Yixuan Li, and Ziwei Liu. Generalized Out-of-Distribution Detection: A Survey. *arXiv preprint arXiv:2110.11334*, 2021.

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

[39] Baochen Sun and Kate Saenko. Deep coral: Correlation alignment for deep domain adaptation. In *European conference on computer vision*, pages 443–450. Springer, 2016.

[40] Divyat Mahajan, Shruti Tople, and Amit Sharma. Domain Generalization using Causal Matching. In *Proceedings of the 38th International Conference on Machine Learning*, pages 7313–7324. PMLR, 2021.

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

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

[43] Toshihiko Matsuura and Tatsuya Harada. Domain generalization using a mixture of multiple latent domains. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 11749–11756, 2020.

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

[45] Kaiyang Zhou, Yongxin Yang, Yu Qiao, and Tao Xiang. Domain Generalization with MixStyle. In *International Conference on Learning Representations*, 2020.

[46] Shiv Shankar, Vihari Piratla, Soumen Chakrabarti, Siddhartha Chaudhuri, Preethi Jyothi, and Sunita Sarawagi. Generalizing across domains via cross-gradient training. 2018.

[47] 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.

[48] Minghao Xu, Jian Zhang, Bingbing Ni, Teng Li, Chengjie Wang, Qi Tian, and Wenjun Zhang. Adversarial domain adaptation with domain mixup. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 6502–6509, 2020.

[49] Fabio M. Carlucci, Antonio D'Innocente, Silvia Bucci, Barbara Caputo, and Tatiana Tommasi. Domain Generalization by Solving Jigsaw Puzzles. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2229–2238, 2019.

[50] Jongbin Ryu, Gitaek Kwon, Ming-Hsuan Yang, and Jongwoo Lim. Generalized convolutional forest networks for domain generalization and visual recognition. In *International conference on learning representations*, 2019.

[51] Kaiyang Zhou, Yongxin Yang, Yu Qiao, and Tao Xiang. Domain Adaptive Ensemble Learning. *IEEE Transactions on Image Processing*, 30:8008–8018, 2021.

[52] Massimiliano Mancini, Samuel Rota Bulo, Barbara Caputo, and Elisa Ricci. Best sources forward: domain generalization through source-specific nets. In *2018 25th IEEE international conference on image processing*, pages 1353–1357, 2018.

[53] Da Li, Yongxin Yang, Yi-Zhe Song, and Timothy M Hospedales. Learning to generalize: Meta-learning for domain generalization. In *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.

[54] Yiying Li, Yongxin Yang, Wei Zhou, and Timothy Hospedales. Feature-critic networks for heterogeneous domain generalization. In *International Conference on Machine Learning*, pages 3915–3924. PMLR, 2019.

[55] Yuyang Zhao, Zhun Zhong, Fengxiang Yang, Zhiming Luo, Yaojin Lin, Shaozi Li, and Nicu Sebe. Learning to generalize unseen domains via memory-based multi-source meta-learning for person re-identification. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6277–6286, 2021.

[56] João Gama, Indrė Žliobaitė, Albert Bifet, Mykola Pechenizkiy, and Abdelhamid Bouchachia. A survey on concept drift adaptation. *ACM computing surveys*, 46(4):1–37, 2014.

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

[58] Yanbin Liu, Juho Lee, Linchao Zhu, Ling Chen, Humphrey Shi, and Yi Yang. A Multi-Mode Modulator for Multi-Domain Few-Shot Classification. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8453–8462, 2021.

[59] Muhammad Ghifary, W. Bastiaan Kleijn, Mengjie Zhang, and David Balduzzi. Domain Generalization for Object Recognition With Multi-Task Autoencoders. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2551–2559, 2015.

[60] Chang Liu, Xinwei Sun, Jindong Wang, Haoyue Tang, Tao Li, Tao Qin, Wei Chen, and Tie-Yan Liu. Learning Causal Semantic Representation for Out-of-Distribution Prediction. In *Advances in Neural Information Processing Systems*, volume 34, pages 6155–6170. Curran Associates, Inc., 2021.

[61] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.

[62] Ethan Perez, Florian Strub, Harm de Vries, Vincent Dumoulin, and Aaron Courville. FiLM: Visual Reasoning with a General Conditioning Layer. *Proceedings of the AAAI Conference on Artificial Intelligence*, 32(1), 2018.

[63] Boris Oreshkin, Pau Rodríguez López, and Alexandre Lacoste. TADAM: Task dependent adaptive metric for improved few-shot learning. In

Advances in Neural Information Processing Systems, volume 31. Curran Associates, Inc., 2018.

[64] James Requeima, Jonathan Gordon, John Bronskill, Sebastian Nowozin, and Richard E Turner. Fast and flexible multi-task classification using conditional neural adaptive processes. *Advances in Neural Information Processing Systems*, 32, 2019.

[65] Michael Tschannen, Olivier Bachem, and Mario Lucic. Recent advances in autoencoder-based representation learning. *arXiv preprint arXiv:1812.05069*, 2018.

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

[67] Irina Higgins, Loic Matthey, Arka Pal, Christopher Burgess, Xavier Glorot, Matthew Botvinick, Shakir Mohamed, and Alexander Lerchner. beta-vae: Learning basic visual concepts with a constrained variational framework. 2016.

[68] Christopher P Burgess, Irina Higgins, Arka Pal, Loic Matthey, Nick Watters, Guillaume Desjardins, and Alexander Lerchner. Understanding disentangling in beta-vae. *arXiv preprint arXiv:1804.03599*, 2018.

[69] Kaiyang Zhou, Yongxin Yang, Timothy Hospedales, and Tao Xiang. Learning to generate novel domains for domain generalization. In *European conference on computer vision*, pages 561–578. Springer, 2020.

[70] Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

[71] Yaroslav Ganin and Victor Lempitsky. Unsupervised domain adaptation by backpropagation. In *International conference on machine learning*, pages 1180–1189. PMLR, 2015.

[72] Yuval Netzer, Tao Wang, Adam Coates, Alessandro Bissacco, Bo Wu, and Andrew Y Ng. Reading digits in natural images with unsupervised feature learning. 2011.

[73] 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.

[74] Arthur Gretton, Karsten M Borgwardt, Malte J Rasch, Bernhard Schölkopf, and Alexander Smola. A kernel two-sample test. *The Journal of Machine Learning Research*, 13(1):723–773, 2012.

[75] Chelsea Finn, Pieter Abbeel, and Sergey Levine. Model-agnostic meta-learning for fast adaptation of deep networks. In *International conference on machine learning*, pages 1126–1135. PMLR, 2017.

[76] Flood Sung, Yongxin Yang, Li Zhang, Tao Xiang, Philip H. S. Torr, and Timothy M. Hospedales. Learning to Compare: Relation Network for Few-Shot Learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1199–1208, 2018.

[77] David Lopez-Paz, Robert Nishihara, Soumith Chintala, Bernhard Scholkopf, and Leon Bottou. Discovering Causal Signals in Images. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6979–6987, 2017.

[78] 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.

[79] Augustus Odena, Vincent Dumoulin, and Chris Olah. Deconvolution and checkerboard artifacts. *Distill*, 1(10):e3, 2016.

[80] Da Li, Jianshu Zhang, Yongxin Yang, Cong Liu, Yi-Zhe Song, and Timothy M Hospedales. Episodic training for domain generalization. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1446–1455, 2019.

[81] Ulrike Von Luxburg. A tutorial on spectral clustering. *Statistics and computing*, 17(4):395–416, 2007.

[82] Leon A Gatys, Alexander S Ecker, and Matthias Bethge. A neural algorithm of artistic style. *arXiv preprint arXiv:1508.06576*, 2015.

[83] Xun Huang and Serge Belongie. Arbitrary style transfer in real-time with adaptive instance normalization. In *Proceedings of the IEEE international conference on computer vision*, pages 1501–1510, 2017.

[84] Amrith Setlur, Oscar Li, and Virginia Smith. Two Sides of Meta-Learning Evaluation: In vs. Out of Distribution. In *Advances in Neural Information Processing Systems*, volume 34, pages 3770–3783, 2021.