Domain Generalization
by Mutual-Information Regularization
with Pre-trained Models

Junbum Cha[1(✉)], Kyungjae Lee[2], Sungrae Park[3], and Sanghyuk Chun[4]

[1] Kakao Brain, Seongnam, South Korea
junbum.cha@kakaobrain.com
[2] Chung-Ang University, Seoul, South Korea
kyungjae.lee@ai.cau.ac.kr
[3] Upstage AI Research, Seoul, South Korea
sungrae.park@upstage.ai
[4] NAVER AI Lab, Seoul, South Korea
sanghyuk.c@navercorp.com

Abstract. Domain generalization (DG) aims to learn a generalized model to an unseen target domain using only limited source domains. Previous attempts to DG fail to learn domain-invariant representations only from the source domains due to the significant domain shifts between training and test domains. Instead, we re-formulate the DG objective using mutual information with the oracle model, a model generalized to any possible domain. We derive a tractable variational lower bound via approximating the oracle model by a pre-trained model, called Mutual Information Regularization with Oracle (MIRO). Our extensive experiments show that MIRO significantly improves the out-of-distribution performance. Furthermore, our scaling experiments show that the larger the scale of the pre-trained model, the greater the performance improvement of MIRO. Code is available at https://github.com/kakaobrain/miro.

1 Introduction

Emerging studies on the generalizability of deep neural networks have revealed that the existing models, which assume independent and identically distributed (i.i.d.) training and test distribution, are not robust to significant distribution shifts between training and test distribution, *e.g.*, backgrounds [60], geographic distribution [57], demographic statistics [50,65], textures [3,23], or day-to-night shifts [17,40]. Domain generalization (DG) aims to learn robust representations against distribution shifts from multiple source domains during training. The trained model is evaluated on an unseen domain to measure the robustness. The

existing DG approaches have tried to learn invariant features across multiple domains [2,11,13,22,32,53,70]. However, recent studies [24,29] have shown that simple baselines without learning invariant features are comparable to or even outperform the existing DG methods on the diverse DG benchmarks with a fair evaluation protocol in realistic settings (*e.g.*, using ResNet-50 instead of ResNet-18 [25]). We presume that it is because training and test distributions differ too significantly to learn domain-invariant features by the training distribution only.

Instead of learning domain-invariant features, we let a model learn similar features to "oracle" representations, *i.e.*, an optimal model generalized to *any* domain. In particular, we re-formulate the DG problem by maximizing the mutual information (MI) between the oracle model representations and the target model representations while preserving the training loss on source domains. However, the oracle model is not achievable in practice. Hence, we use a large pre-trained model (*e.g.*, ImageNet pre-trained ResNet-50 [25]) as an approximation. With this approximation, we derive a tractable variational lower bound of the proposed maximization problem, named Mutual Information Regularization with Oracle (MIRO). At a high level, our MIRO objective consists of two objectives: an original target task (*i.e.*, an ERM objective) and a regularization term between the pre-trained model and the current target model. Note that the standard DomainBed benchmark [24] uses the ImageNet pre-trained ResNet-50 as the initialization of a DG method, thus, we use the pre-trained ResNet as the initialization and the approximation of the oracle model at the same time.

While a naive fine-tuning approach of a large pre-trained model can harm the robustness against distribution shifts [31,59], our proposed algorithm remarkably improves the robustness against unseen domains during fine-tuning in a plug-and-play manner to any scale of the backbone model and datasets.

In our experiment, we observe that the naive fine-tuning of a larger pre-trained model can fail to provide better performances, even though the larger pre-trained model is trained with more data and domains. For example, ERM with the ResNet pre-trained on ImageNet (trained with 1.3M images) shows 64.2% of averaged accuracy, while ERM with the ViT pre-trained on CLIP (trained with 400M image-caption pairs) shows 61.1%. On the other hand, we show that our method can significantly improve the average DG performances with backbone models at different scales, *e.g.*, ImageNet pre-trained ResNet (64.2% → 65.9%), 400M image-text pre-trained ViT (CLIP) [45] (61.1% → 73.7%) and Instagram 3.6B pre-trained RegNet (SWAG) [52] (68.0% → 74.1%). Especially, we observe that the pre-trained knowledge by larger pre-trained models, such as SWAG and CLIP, is more effective to learn domain generalized features than the ImageNet pre-trained model: MIRO with the ViT pre-trained on CLIP outperforms MIRO with the ResNet pre-trained on ImageNet in contrast to the naive fine-tuning. Furthermore, our feature-level regularization method is easily combined with the existing parameter space ensemble methods [13,59] (74.1% → **77.3%** average DG accuracy by combining with SWAD [13] and pre-trained RegNet).

Our contribution is as follows: (1) We re-formulate the DG objective by mutual information with the oracle model. Then, we approximate the oracle by a large pre-trained model to derive a tractable approximation of the target objective. We propose Mutual Information Regularization with Oracle (MIRO) to solve our objective. (2) We analyze the pre-trained models in terms of the MI with the oracle model. Our analysis shows that naive fine-tuning of pre-trained models can harm the MI with the oracle, on the other hand, MIRO shows high MI with the oracle. (3) We compare MIRO with state-of-the-art DG methods on DomainBed. MIRO outperforms all methods in all settings, including varying optimizers and pre-trained models. We also provide extensive analysis to understand MIRO. For example, we observe that MIRO shows stronger DG performances with larger pre-trained models, such as SWAG [52] or CLIP [45].

2   Related Works

Domain Generalization. Learning domain-invariant features from source domains has been a major branch in the DG field. The main idea is discarding biased knowledge to a specific domain while preserving invariant features over source domains, by minimizing feature divergences between the source domains [22,35,37,39,41,53,68], simulating domain shifts based on meta-learning [5,11,19,32,34,67], robust optimization [2,13,30,49,51], or augmenting source domain examples [4,12,42,43,47,64,69,70]. However, even if the model learns invariant representation to source domains, it can still be biased toward the source domains which causes limited performance on unseen target domains. That is, learning invariant representation across source domains is not enough to achieve the underlying objective of domain generalization [11,14,15]. To compensate for the issue, this paper employs pre-trained models, which provide general representations across various domains including unseen target domains.

Exploiting Pre-trained Models There have been numerous attempts to exploit pre-trained models in various fields. Transfer learning [36,62] and knowledge distillation [1,54] employ pre-trained models to improve in-domain performance when dataset or architecture shift occurs between pre-training and fine-tuning. Continual learning utilizes the pre-trained model to maintain old task performance when learning new tasks [38]. Recently, several studies targeting the out-of-distribution generalization are emerging [31,59]. Kumar *et al.* [31] show that naive fine-tuning distorts the pre-trained features and propose a simple baseline, named LP-FT, to alleviate the distortion. WiSE-FT [59] focuses on zero-shot models. It combines pre-trained and fine-tuned weights to preserve the generalizability of the pre-trained zero-shot models. In this paper, we propose a MI-based regularization method, MIRO, to exploit the generalizability of the pre-trained representation in the training process.

3   Methods

In this section, we first re-formulate the objective for the out-of-domain general-ization by introducing an oracle model. Then, we derive a tractable variational bound of the objective by approximating the oracle model to the pre-trained model. The final form consists of the empirical risk and the mutual informa-tion (MI) regularization by querying the approximated oracle, named Mutual Information Regularization with Oracle (MIRO). We empirically validate our approximation by MI between the oracle model and large pre-trained models.

3.1   Mutual Information Regularization with Oracle

The main idea of the proposed method is to guide the learning process using oracle representations of training datasets. In general, the problem of domain generalization (DG) is to find a model that minimizes an expected loss of *any* domain by using training datasets from only partial domains, which are called source domains. Many existing methods minimize an empirical loss averaged over source domains. More specifically, suppose that training samples $\{\mathcal{S}_d\}_{d=1}^m$ are given in m domains and we consider a hypothesis set $\mathcal{H}$ for optimization. Then, many existing DG frameworks can be formulated as follows:

$$\bar{h} = \arg\min_{h \in \mathcal{H}} \sum_{d=1}^m \mathcal{E}_{\mathcal{S}_d}(h), \tag{1}$$

where d indicates an individual source domain and $\mathcal{E}_{\mathcal{S}_d}$ is an empirical loss over the source domain d. Note that majority of existing DG methods can be inter-preted as the variant of Eq. (1). For example, if we choose a simple cross-entropy loss for $\mathcal{E}_{\mathcal{S}_d}$, then Eq. (1) becomes "ERM" baseline used in [24][1]. Otherwise, $\mathcal{E}_{\mathcal{S}_d}$ can be formulated as a regularized ERM, such as IRM [2] or CORAL [53]. However, the formulation (1) still suffers from learning domain-invariant repre-sentations using only partial domains when the target distribution differs signifi-cantly from the training distribution. For example, CORAL, the state-of-the-art method, shows inconsistent out-of-domain accuracies across domains in Domain-Net [44]. While CORAL achieves $\approx$50% top-1 accuracy on four *easy* domains (59.2% for Clipart, 46.6% for Painting, 59.8% for Real, 50.1% for Sketches), it only shows 13.4% for QuickDraw and 19.7% for Infographics where the domains show the significant distribution shift comparing to others.

   To alleviate this issue, we re-formulate the DG problem by employing *oracle* representations of source domains. Here, we define an oracle model as a model that can be generalized to *any* possible domain, not only for the source domains.

---

[1] Note that the terminology ERM can be unfair because other methods also minimize "empirical risk" but with different loss designs. We use the terminology "ERM" to indicate the cross-entropy baseline as suggested by Gulrajani and Lopez-Paz [24].

We define a model as a composition of a feature extractor f and a classifier g on the feature space where the whole classifier h can be written as $h = f \circ g$. Then, let f^* be a feature extractor of the oracle model. We first start from a strong assumption: we may assume that f^* is accessible during the training phase. Then, we can obtain additional information from f^* by querying the oracle representations of training samples in the source domains. By using the oracle representations, we can guide the learning process of a target model by maximizing MI between oracle representations and target ones. We formulate the proposed oracle-guided DG framework as follows:

$$\max_{h} \quad I(Z_{f^*}; Z_f)$$
$$\text{s.t.} \quad \mathcal{E}_{\mathcal{S}}(h) - \mathcal{E}_{\mathcal{S}}(\bar{h}) \leq \epsilon, \tag{2}$$

where Z_{f^*} is a random feature extracted by f^* and Z_f is a random feature extracted by a target model f. $I(Z_{f^*}; Z_f)$ is MI between Z_{f^*} and Z_f, and $\mathcal{E}_{\mathcal{S}}(\cdot) = \sum_{d=1}^{m} \mathcal{E}_{\mathcal{S}_d}(\cdot)$. The inequality constraint ensures the performance of the target model on the source domains. Maximizing the MI will inhibit the target model from overfitting domain-specific features in the limited source domains. Because we assume that the "oracle" is generalized well to *any* possible domain, the MI constraints (2) will be beneficial to learning robust representations.

Unfortunately, the oracle feature extractor f^* is not accessible in practice. Instead, we approximate the oracle feature extractor by using a pre-trained model f^0. Our assumption is that a model pre-trained on large-scale diverse datasets, such as ImageNet [48], contains information on diverse domains. In practice, we choose f^0 as the ImageNet pre-trained ResNet-50 [25], the standard initialization choice for evaluating DG algorithms [24]. We also consider models trained by larger diverse datasets, such as CLIP [45] (trained with 400M web crawled image-text pairs) and SWAG [52] (trained with 3.6B noisy image-hashtag pairs crawled from Instagram). Although using CLIP and SWAG is not a fair comparison to the existing DG benchmark, here, we emphasize that naive fine-tuning of large pre-trained models leads to inferior generalizability to extreme distribution shifts at test time [31,59]. In our experiments, we also observe a similar observation: naive fine-tuning of CLIP shows an inferior DG performance (61.1%) than ERM (64.2%).

Through the approximation of the oracle model, we derive a tractable variational bound of our objective (2). We assume a pre-trained model f^0 is located near f^* in terms of distance equipped on the hypothesis set of the feature extractors and it can provide approximated representation of f^*. Under this assumption, we can obtain a tractable objective by deriving an approximated lower bound of the MI. We first derive the variational lower bound of the MI as follows:

$$I(Z_{f^*}; Z_f) = \mathbb{E}_{Z_{f^*}, Z_f} \left[\log \frac{q(Z_{f^*} \mid Z_f)}{p(Z_{f^*})} \right] + KL(p(Z_{f^*} \mid Z_f) \| q(Z_{f^*} \mid Z_f))$$
$$\geq \mathbb{E}_{Z_{f^*}, Z_f} [\log q(Z_{f^*} \mid Z_f)] + H(Z_{f^*}), \qquad (3)$$

where q is the variational distribution with a mild regularity condition. More detailed derivation can be found in Barber and Agakov [7]. Then, we approximate the expectation in Eq. (3) by using f^0.

$$I(Z_{f^*}; Z_f) \geq \mathbb{E}_{Z_{f^*}, Z_f} [\log q(Z_{f^*} \mid Z_f)] + H(Z_{f^*})$$
$$\geq \mathbb{E}_{Z_{f^0}, Z_f} [\log q(Z_{f^0} \mid Z_f)] - C d_{2,\infty}(f^*, f^0) + H(Z_{f^*}), \qquad (4)$$

where C is a constant and $d_{2,\infty}(f^*, f^0) := \sup_x \| f^*(x) - f^0(x) \|_2$. Note that $d_{2,\infty}$ is a proper metric on the hypothesis set of feature extractor. The last inequality of Equation (4) is derived by using the first-order Taylor expansion and assuming the regularity condition of q (See Appendix). We would like to note that the inequality is tight enough due to Taylor's theorem. In other words, equality condition of the last inequality of Eq. (4) is $d_{2,\infty}(f^*, f^0) = 0$. Hence, $d_{2,\infty}(f^*, f^0)$ represents the effect of the pre-trained model f^0 on the approximation of the lower bound. Intuitively speaking, the lower bound shows that the smaller $d_{2,\infty}(f^*, f^0)$ is, the tighter the gap between the true lower bound and approximated one is. In summary, the MI between Z_{f^*} and Z_f can be maximized by maximizing the term $\mathbb{E}_{Z_{f^0}, Z_f} [\log q(Z_{f^0} \mid Z_f)]$.

Algorithm 1: Mutual Information Regularization with Oracle (MIRO)

Input: feature extractor f, classifier g, mean encoder μ, variance encoder Σ, regularization coefficient λ, batch size N.
Init: initialize f to pre-trained feature extractor f^0.
Output: learned feature extractor f and learned classifier g.
for sampled mini-batch $(\mathbf{x}, \mathbf{y})$ **do**
$\quad \mathbf{z}_f = f(\mathbf{x})$
$\quad \mathbf{z}_{f^0} = f^0(\mathbf{x})$
$\quad \mathcal{L} =$
$\quad \frac{1}{N} \sum_i^N \left[\texttt{CrossEntropy}\left(g(z_f^i), y^i \right) + \lambda \left(\log |\Sigma(z_f^i)| + \| z_{f^0}^i - \mu(z_f^i) \|_{\Sigma(z_f^i)^{-1}}^2 \right) \right]$
$\quad$ update f, g, μ, Σ to minimize $\mathcal{L}$
end

Finally, to consider the constraint term, we introduce the Lagrangian method to Eq. (2), then we can derive an objective function from Equation (4):

$$R(h) = \mathbb{E}_{Z_{f^0}, Z_f} [\log q(Z_{f^0} \mid Z_f)] - \beta \mathcal{E}_S(h), \qquad (5)$$

where β indicates the Lagrangian multiplier. Note that the entropy of Z_{f^*} and $d_{2,\infty}(f^*, f^0)$ are omitted, since they are independent to our optimization target $h = f \circ g$. In the implementation, we model the variational distribution as a

Gaussian distribution with mean vector $\mu(Z_f)$ and covariance matrix $\Sigma(Z_f)$ and replace the multiplier β with the regularization coefficient λ. Then, our final loss function becomes:

$$\textbf{(MIRO)} \quad \mathcal{L}(h) = \mathcal{E}_\mathcal{S}(h) + \lambda \mathbb{E}_{Z_{f^0}, Z_f} \left[\log |\Sigma(Z_f)| + \|Z_{f^0} - \mu(Z_f)\|^2_{\Sigma(Z_f)^{-1}} \right], \tag{6}$$

where $\|x\|_A = \sqrt{x^\mathsf{T} A x}$ and constants independent on h are omitted. Then, we optimize the loss function using a stochastic gradient method. The entire learning process is summarized in Algorithm 1. In the following sections, we empirically justify our approximation of f^* and explain implementation details for the mean and variance encoders of the Gaussian distribution q.

3.2   Mutual Information Analysis with the Oracle Model

Here, we empirically show how our approximation by pre-trained models is close to the oracle model and how our algorithm is effective to learn representations having high mutual information (MI) to the underlying oracle model. More specifically, we compare MI between the candidate models and the oracle model on the PACS dataset. Since the *true* oracle model is not achievable in practice, we train an oracle model by directly optimizing a model on the entire domains. We train two oracle models with ResNet-50 and RegNetY-16GF backbones, where the average validation accuracies across all domains are 97.2% and 98.4%, respectively. We estimate MI between models by mutual information neural estimation (MINE) [9]. We describe the full details in Appendix.

Figure 1 illustrates the empirical MI between the candidate models and the oracle model. In the figures, we first observe that the larger and more powerful

(a) ResNet-50   (b) RegNetY-16GF
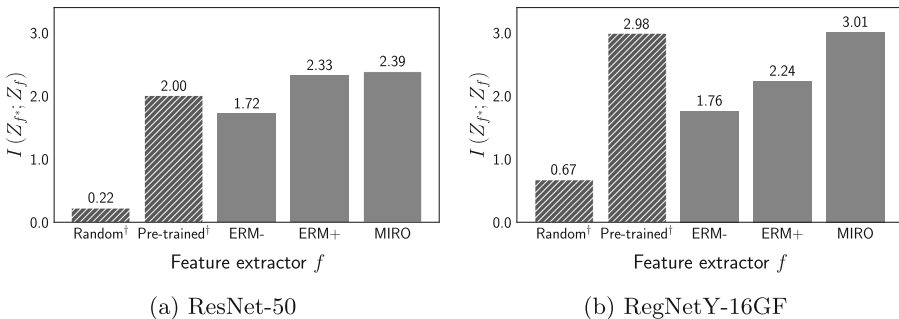
Fig. 1. Mutual information $I(Z_{f^*}; Z_f)$ with oracle model. The mutual information is estimated by MINE [9] in PACS. Oracle model is trained using all of the four domains. *Random* and *Pre-trained* indicate random and pre-trained model initialization, respectively. *ERM-* and *ERM+* are trained from random and pre-trained model initialization, respectively. † indicates models without fine-tuning. The experiments are repeated with two pre-trained models: ImageNet 1.3M pre-trained ResNet-50 and Instagram 3.6B pre-trained RegNetY-16GF.

pre-trained backbone ("Pre-trained" in Fig. 1b) shows higher MI than the smaller backbone ("Pre-trained" in Fig. 1a). Both pre-trained models consistently outperform "Random" in MI regardless of the backbone models. Our observations imply that a larger and stronger model is closer to the oracle model in terms of MI. Similarly, we observe that ERM+ always shows high MI than ERM−. However, interestingly, in Fig. 1b, we observe that fine-tuning significantly harms MI of the pre-trained model ("Pre-trained" vs. "ERM+") when the pre-trained model becomes larger and more powerful. Our observation is aligned in the same line as the previous studies on fine-tuning of large models [31,59]. Lastly, in both scenarios of ImageNet pre-trained ResNet (Fig. 1a) and SWAG pre-trained RegNet (Fig. 1b), our MIRO shows the highest MI with the oracle model. Note that MI with the oracle model may not be completely aligned with the DG performance, but in practice, we observed that the evaluation ranking of the candidates is the same as the MI ranking; MIRO scores the best, followed by ERM+ and ERM−. Detailed results are provided in Appendix.

3.3 Features and Encoders Design

Multi-scale Features. One can only use the last-level features for our regularization. However, high-level features can include pre-training task-related information, often irrelevant to the target task. Instead, we use the intermediate outputs by each model block, *i.e.*, stem output, blocks 1, 2, 3, and 4 for ResNet [25] and RegNet [46], and stem output, blocks 3, 6, 9, and 12 for ViT-B.

Design of the Mean and Variance Encoders. The multi-level structure increases the feature size, resulting in a computational cost increase. We alleviate the issue by employing simple yet effective architectures, identity function for the mean encoder and a bias-only model with diagonal covariance for the variance encoder. We also tested more complicated architectures, but only computational cost was increased without performance improvement.

4 Experiments

4.1 Experiment Setups and Implementation Details

Evaluation Protocols and Datasets. We employ DomainBed evaluation protocols [13,24] for a fair comparison. The five benchmark datasets are used: PACS [33] (4 domains, 7 classes, and 9,991 images), VLCS [21] (4 domains, 5 classes, and 10,729 images), OfficeHome [56] (4 domains, 65 classes, and 15,588 images), TerraIncognita [8] (4 domains, 10 classes, and 24,788 images), and DomainNet [44] (6 domains, 345 classes, and 586,575 images). All performance scores are evaluated by *leave-one-out cross-validation*, where averaging all cases that use a single domain as the target (test) domain and the others as the source (training) domains. Every experiment is repeated three times. We leave 20% of source domain data for validation. We use training-domain validation for the model selection and the hyperparameter search following DomainBed [24].

Implementation Details. We use ResNet-50 [25] pre-trained in the ImageNet [48] as default. The model is optimized using Adam [28] optimizer. A mini-batch contains all domains and 32 examples per domain. The regularization coefficient λ is tuned in [1.0, 0.1, 0.01, 0.001]. The other hyperparameters, such as batch size, learning rate, dropout rate, and weight decay, are tuned in the similar search space proposed in Cha et al. [13]. We provide full details in Appendix.

4.2   Main Results

Comparison with Domain Generalization Methods. We provide exhaustive out-of-domain performance comparisons on five DG benchmarks in Table 1. Compared to ERM, the proposed MI regularization significantly improves performance on every benchmark dataset, resulting in +1.7pp average improvement. Compared with the state-of-the-art methods, MIRO achieves the best performances in all benchmarks, except PACS. Especially, MIRO remarkably outperforms previous methods: +1.3pp in OfficeHome (mDSDI [11]; 69.2% → 70.5%) and +1.8pp in TerraIncognita (SagNet [42]; 48.6% → 50.4%). Considering the extensive experiment setup with 5 datasets and 22 target domains, the results demonstrate the effectiveness of MIRO to the diverse visual data types.

The second part of Table 1 shows the performance with stochastic weight averaging densely (SWAD) [13], a state-of-the-art optimizer for DG by seeking flat minima. Since SWAD is an orthogonal direction to MIRO, we also evaluate the combination of MIRO and SWAD. As shown in the table, the combination of MIRO and SWAD achieves the best performance in all datasets, resulting in +0.8pp average improvement compared to the previous best results.

In the last part of Table 1, we push the limits of the out-of-domain performance by employing a large-scale backbone, RegNetY-16GF pre-trained by SWAG [52]; a weakly-supervised pre-trained model using 3.6 billion noisy Instagram images and hashtags. As shown in our previous study on MI with the oracle model, the pre-trained RegNet has higher MI than ImageNet pre-trained ResNet (Fig. 1). In the experiments, we first observe that the improvement gap by MIRO becomes remarkably large compared to the ResNet pre-trained model (from +1.7pp to +6.1pp). We presume that this significantly large gap originated from the negative effect of naive fine-tuning as observed by previous works [31,59] and our study (Fig. 1b). As shown in Fig. 1b, MIRO keeps MI with the oracle model high, resulting in remarkable performance gains on large-scale models. We further explore the effect of the scalability of pre-trained models in the later section. Finally, by combining MIRO with RegNet backbone and SWAD, we achieve the best domain generalization results (77.3%) on our evaluation benchmark.

MIRO with Various Pre-trained Models. In this subsection, we investigate the robustness of the proposed method to the choice of pre-trained models. In Table 2, we explore the performance changes of MIRO by varying pre-training datasets, methods, and backbones. From the pre-training method perspective,

Table 1. Comparison with domain generalization methods. Out-of-domain accuracies on five domain generalization benchmarks are shown. We highlight the **best results** in bold. The results marked by †, ‡ are the reported numbers from Gulrajani and Lopez-Paz [24] and Cha et al. [13], respectively. The results of Fish, SelfReg, and mDSDI are the reported ones from each paper. Average accuracies and standard errors are reported from three trials.

Algorithm	PACS	VLCS	OfficeHome	TerraInc	DomainNet	Avg.
MMD† [35]	84.7 ± 0.5	77.5 ± 0.9	66.3 ± 0.1	42.2 ± 1.6	23.4 ± 9.5	58.8
Mixstyle‡ [70]	85.2 ± 0.3	77.9 ± 0.5	60.4 ± 0.3	44.0 ± 0.7	34.0 ± 0.1	60.3
GroupDRO† [49]	84.4 ± 0.8	76.7 ± 0.6	66.0 ± 0.7	43.2 ± 1.1	33.3 ± 0.2	60.7
IRM† [2]	83.5 ± 0.8	78.5 ± 0.5	64.3 ± 2.2	47.6 ± 0.8	33.9 ± 2.8	61.6
ARM† [67]	85.1 ± 0.4	77.6 ± 0.3	64.8 ± 0.3	45.5 ± 0.3	35.5 ± 0.2	61.7
VREx† [30]	84.9 ± 0.6	78.3 ± 0.2	66.4 ± 0.6	46.4 ± 0.6	33.6 ± 2.9	61.9
CDANN† [37]	82.6 ± 0.9	77.5 ± 0.1	65.8 ± 1.3	45.8 ± 1.6	38.3 ± 0.3	62.0
DANN† [22]	83.6 ± 0.4	78.6 ± 0.4	65.9 ± 0.6	46.7 ± 0.5	38.3 ± 0.1	62.6
RSC† [26]	85.2 ± 0.9	77.1 ± 0.5	65.5 ± 0.9	46.6 ± 1.0	38.9 ± 0.5	62.7
MTL† [10]	84.6 ± 0.5	77.2 ± 0.4	66.4 ± 0.5	45.6 ± 1.2	40.6 ± 0.1	62.9
Mixup† [58,61,63]	84.6 ± 0.6	77.4 ± 0.6	68.1 ± 0.3	47.9 ± 0.8	39.2 ± 0.1	63.4
MLDG† [32]	84.9 ± 1.0	77.2 ± 0.4	66.8 ± 0.6	47.7 ± 0.9	41.2 ± 0.1	63.6
Fish [51]	85.5 ± 0.3	77.8 ± 0.3	68.6 ± 0.4	45.1 ± 1.3	42.7 ± 0.2	63.9
ERM‡ [55]	84.2 ± 0.1	77.3 ± 0.1	67.6 ± 0.2	47.8 ± 0.6	44.0 ± 0.1	64.2
SagNet† [42]	$\mathbf{86.3} \pm 0.2$	77.8 ± 0.5	68.1 ± 0.1	48.6 ± 1.0	40.3 ± 0.1	64.2
SelfReg [27]	85.6 ± 0.4	77.8 ± 0.9	67.9 ± 0.7	47.0 ± 0.3	42.8 ± 0.0	64.2
CORAL† [53]	86.2 ± 0.3	78.8 ± 0.6	68.7 ± 0.3	47.6 ± 1.0	41.5 ± 0.1	64.5
mDSDI [11]	86.2 ± 0.2	$\mathbf{79.0} \pm 0.3$	69.2 ± 0.4	48.1 ± 1.4	42.8 ± 0.1	65.1
MIRO	85.4 ± 0.4	$\mathbf{79.0} \pm 0.0$	70.5 ± 0.4	$\mathbf{50.4} \pm 1.1$	$\mathbf{44.3} \pm 0.2$	**65.9**
Combined with SWAD [13]						
ERM + SWAD‡	88.1 ± 0.1	79.1 ± 0.1	70.6 ± 0.2	50.0 ± 0.3	46.5 ± 0.1	66.9
CORAL + SWAD‡	88.3 ± 0.1	78.9 ± 0.1	71.3 ± 0.1	51.0 ± 0.1	46.8 ± 0.0	67.3
MIRO + SWAD	$\mathbf{88.4} \pm 0.1$	$\mathbf{79.6} \pm 0.2$	$\mathbf{72.4} \pm 0.1$	$\mathbf{52.9} \pm 0.2$	$\mathbf{47.0} \pm 0.0$	**68.1**
Using RegNetY-16GF backbone with SWAG pre-training [52]						
ERM	89.6 ± 0.4	78.6 ± 0.3	71.9 ± 0.6	51.4 ± 1.8	48.5 ± 0.6	68.0
MIRO	$\mathbf{97.4} \pm 0.2$	$\mathbf{79.9} \pm 0.6$	$\mathbf{80.4} \pm 0.2$	$\mathbf{58.9} \pm 1.3$	$\mathbf{53.8} \pm 0.1$	**74.1**
ERM + SWAD	94.7 ± 0.2	79.7 ± 0.2	80.0 ± 0.1	57.9 ± 0.7	53.6 ± 0.6	73.2
MIRO + SWAD	$\mathbf{96.8} \pm 0.2$	$\mathbf{81.7} \pm 0.1$	$\mathbf{83.3} \pm 0.1$	$\mathbf{64.3} \pm 0.3$	$\mathbf{60.7} \pm 0.0$	**77.3**

we examine two image self-supervised pre-training methods (Barlow Twins [66] and MoCo v3 [16]), one image-language self-supervised pre-training method (CLIP [45]), and one weakly-supervised pre-training method (SWAG [52]), as well as ImageNet supervised pre-training baseline (ImageNet ERM). From the pre-training scale perspective, we employ the ImageNet [48] dataset of 1.3 million examples, the CLIP dataset of 400 million examples, and the Instagram

Table 2. Comparison with various pre-training datasets, methods, and backbones. We compare the performance changes according to the scale of the dataset, the method, and the backbone architecture of pre-training. ResNet-50 architecture is used as default. `OH`, `TI`, and `DN` indicate `OfficeHome`, `TerraIncognita`, and `DomainNet`, respectively. Every accuracy is averaged over three trials.

| Dataset (size)	Pre-training	Alg.	PACS	VLCS	OH	TI	DN	Avg.
ImageNet (1.3M)	ERM	ERM	84.2	77.3	67.6	47.8	44.0	64.2
		MIRO	85.4	79.0	70.5	50.4	44.3	65.9 (+1.7)
	Barlow Twins	ERM	78.7	77.3	57.6	36.9	41.7	58.4
		MIRO	80.7	79.4	63.7	43.2	42.6	61.9 (+3.5)
	MoCo v3	ERM	86.7	77.3	61.8	49.1	43.8	63.7
		MIRO	86.3	78.5	66.8	48.4	44.7	65.0 (+ 1.3)
CLIP (400M)	CLIP (ResNet)	ERM	64.3	69.8	28.2	32.9	29.5	44.9
		MIRO	76.6	78.9	59.5	49.0	42.0	61.2 (+ 16.3)
	CLIP (ViT)	ERM	83.4	75.9	66.4	35.3	44.4	61.1
		MIRO	95.6	82.2	82.5	54.3	54.0	73.7 (+12.6)
Instagram (3.6B)	SWAG (RegNet)	ERM	89.6	78.6	71.9	51.4	48.5	68.0
		MIRO	97.4	79.9	80.4	58.9	53.8	74.1 (+6.1)

dataset of 3.6 billion examples. We use ResNet-50 [25] backbone architecture as default, but a bigger model is also used for the large-scale pre-training, such as ViT-B [18] for CLIP or RegNetY-16GF [46] for SWAG.

As shown in the table, MIRO improves performances compared with the baseline ERM in all experiments. For the ImageNet pre-training, applying MIRO results in performance improvements of +1.7pp, +3.5pp, and +1.3pp for ERM (supervised learning), Barlow Twins, and MoCo v3, respectively. For the large-scale pre-training, such as CLIP and SWAG, MIRO brings larger performance improvements of +16.3pp, +12.6pp, and +6.1pp for CLIP, CLIP-ViT, and SWAG, respectively. These experiments demonstrate the robustness of the proposed method to the pre-training methods, datasets, and backbone architectures.

Notably, performance improvements of MIRO are remarkable with large-scale pre-trained models, such as CLIP, CLIP-ViT, and SWAG. This is consistent with our observation in Sect. 3.2. Our method helps large-scale pre-trained models (in terms of the pre-training dataset size) not to be biased to the training source domains compared to naive fine-tuning. Especially, naive fine-tuning of CLIP-ViT (61.1%) shows worse out-of-domain performance than fine-tuning ImageNet pre-trained model (64.2%). In contrast, MIRO can leverage the pre-trained knowledge from CLIP-ViT, resulting in superior performance (73.7%) compared with the ImageNet pre-trained model (65.9%). In our later analysis, we show that the knowledge of large-scale pre-trained models is more beneficial to domain generalization than the knowledge of ImageNet pre-trained models.

Table 3. Comparison with methods exploiting pre-trained models. Out-of-domain accuracies on five domain generalization benchmarks are shown. Average accuracies and standard errors are reported from three trials.

Algorithm	PACS	VLCS	OfficeHome	TerraInc	DomainNet	Avg.
CRD [54]	82.3 ± 1.0	76.6 ± 0.9	67.6 ± 0.4	44.0 ± 1.9	42.1 ± 0.1	62.5
VID [1]	84.9 ± 0.3	76.2 ± 0.2	64.6 ± 0.5	48.3 ± 1.3	42.5 ± 0.1	63.3
LP-FT [31]	84.6 ± 0.8	76.7 ± 1.5	65.0 ± 0.2	47.1 ± 0.7	43.0 ± 0.1	63.3
L^2-SP [62]	83.6 ± 0.3	78.8 ± 0.4	65.0 ± 0.3	47.9 ± 2.1	42.5 ± 0.2	63.6
DELTA [36]	83.1 ± 1.1	77.7 ± 0.4	68.5 ± 0.3	45.7 ± 0.9	42.8 ± 0.1	63.6
LwF [38]	83.1 ± 0.8	77.2 ± 0.7	70.0 ± 0.2	49.2 ± 1.2	42.7 ± 0.1	64.5
MIRO	$\mathbf{85.4} \pm 0.4$	$\mathbf{79.0} \pm 0.0$	$\mathbf{70.5} \pm 0.4$	$\mathbf{50.4} \pm 1.1$	$\mathbf{44.3} \pm 0.2$	**65.9**
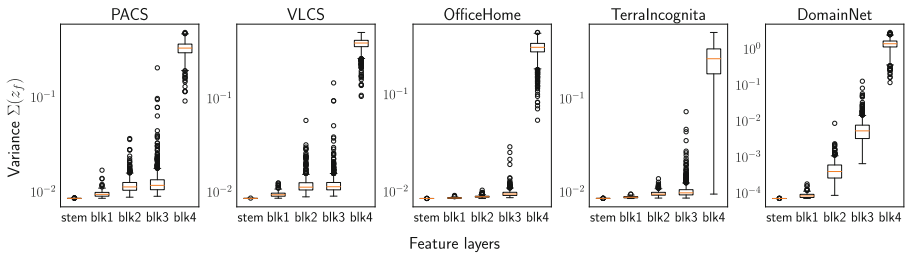
Fig. 2. Distribution of $\Sigma(z_f)$. We plot the estimated variances, $\Sigma(z_f)$, for each layer. X-axis indicates the feature layer where the features z_f are collected. In all datasets, the variances increase as the layer is closer to the output

Comparison with Methods Exploiting Pre-trained Models. Other DG methods simply employ pre-trained models as weight initialization, while MIRO additionally exploits it in the training process. This is the first approach to exploit pre-trained models in domain generalization, but there are several studies in other fields for different purposes. Table 3 provides a comparison of the methods applicable to our DG settings. We exclude the methods that require additional information other than pre-trained models (*e.g.*, pre-training datasets) or are restricted to a specific model. As shown in the table, MIRO outperforms the comparison methods with large margins. These results demonstrate the effectiveness of our method design for the out-of-domain generalization.

4.3 Analysis of MIRO

Loss Function Interpretation: Σ Distribution Analysis. We can interpret the variance term of MIRO, $\Sigma(z_f)$ in Eq. (6), as control variables of the distance loss between pre-trained features z_{f^0} and current learning features z_f. During the training phase, if the variance values become smaller then the model will preserve MI with the pre-trained model. On the contrary, when the model needs to learn new information, the variance will increase. We illustrate the learned

Table 4. Performance improvements in `Camelyon17` medical dataset. Even in the large distribution shift setup between pre-training and target datasets, MIRO consistently outperforms ERM. Every accuracy is averaged over three trials.

Pretrain	Algorithm	1	2	3	4	5	Avg.
ImageNet ERM	ERM	97.1	94.7	95.7	96.4	90.7	94.9
	MIRO	97.5	94.5	95.6	96.7	93.7	95.6 (+0.7)
SWAG	ERM	97.0	94.1	95.3	96.0	89.5	94.4
	MIRO	97.4	95.5	96.5	96.1	90.9	95.3 (+0.9)

variances in Fig. 2. The figure shows that pre-trained information is preserved well in lower layers, while task-specific new information is learned in higher layers. This result is consistent with the interpretation that high layer features represent more task-specific semantic information than low layer features [20]; task shifts during fine-tuning make higher layer features learn more semantics than lower layers.

Case Study on `Camelyon17`: Large Distribution Shift Between Pre-training and Fine-Tuning. As shown in Eq. (4), the tightness of the lower bound is directly connected to the divergence between the representations of oracle and pre-trained models. Therefore, we investigate the case that there is a large shift between pre-trained and target datasets using the medical dataset [6,29], `Camelyon17`. This dataset consists of whole-slide images of histological lymph node sections from the five hospitals, where each hospital corresponds to each domain. The task is to predict whether the image contains tumor tissue of breast cancer. There is a large gap between the pre-training distribution (ImageNet or Instagram-3.6B) and the fine-tuning distribution (`Camelyon17`). Detailed visual examples are provided in Appendix. The results in Table 4 demonstrate MIRO
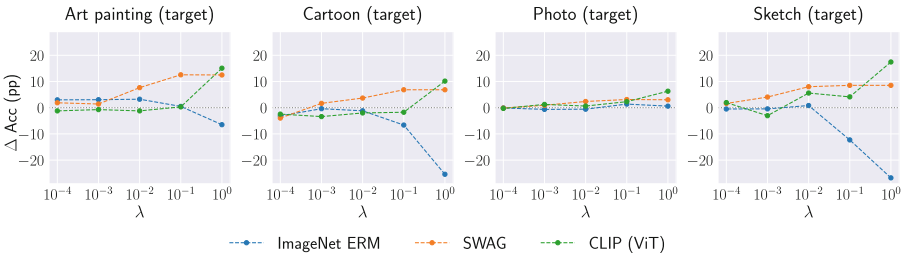
Fig. 3. Comparison of three pre-trained models according to λ. Y-axis indicates the performance difference of MIRO to ERM. λ is the intensity of the mutual information regularization. We compare three models: ResNet-50 pre-trained in ImageNet [25], RegNetY-16GF pre-trained by SWAG [52], and ViT-B pre-trained by CLIP [45]

leads the model to learn robust representations even in the large distribution shift setup between pre-training and fine-tuning.

Relationship Between the Pre-training Scale and the Intensity of the MI Regularization. Our method has a control parameter λ, which controls the balance between the cross-entropy loss and the MI regularization loss. If λ becomes larger, it implies that the strength of MI regularization becomes stronger, while it weakens the strength of the ERM objective. Intuitively, if the pre-trained knowledge is informative enough to the target task, larger λ will improve the performances, while if the pre-trained knowledge is uninformative to the target task, then larger λ can harm the performances, because of the penalty on the ERM objective. We compare three pre-trained models (ImageNet pre-trained model, SWAG, and CLIP-ViT) by varying λ. Figure 3 shows how the out-of-domain performance of MIRO with different pre-trained backbones changes by λ. The additional results on different datasets are given in Appendix.

First, we observe that the ImageNet pre-trained backbone has a negative correlation between the performance difference and λ in target domains. When distribution shifts significantly differ, such as cartoon and sketch domains, we can observe an apparent negative correlation. We presume that it is because the ImageNet samples barely contain non-photo images, such as art painting or sketch images. On the other hand, we observe that MIRO with SWAG and CLIP-ViT backbones make significant performance improvements by choosing larger λ. In other words, SWAG and CLIP-ViT pre-trained knowledge are helpful to learn robust features for various target domains compared to the ImageNet pre-trained model. Furthermore, it implies that larger pre-trained models trained with massive diverse domain images show less sensitivity to the choice of λ, not only bringing remarkable performance improvements as shown in Table 2.

5   Conclusion

Traditional domain generalization (DG) approaches focus to learn a robust representation using multiple source domains. However, in the recent trends of scaling up pre-training, the use of a large-scale pre-trained model becomes more important than the use of DG algorithms for the real-world DG. In line with this trend, we propose Mutual Information Regularization with Oracle (MIRO) to robustly exploit the pre-trained model by approximating an oracle model. To do this, we first re-formulate the domain generalization objective by introducing a concept of an oracle model. Then, we derive a tractable variational bound of the objective by approximating the oracle model with the pre-trained model. Our experimental results demonstrate both the effectiveness and the potential of the proposed method. MIRO achieves state-of-the-art performance in the DomainBed benchmarks. Furthermore, when combining MIRO with large-scale pre-trained backbones, such as CLIP [45] or SWAG [52], the performance improvements remarkably increases. We hope that this study promotes a new

research direction of exploiting pre-trained backbones to learn robust representations for domain generalization.

References

1. Ahn, S., Hu, S.X., Damianou, A., Lawrence, N.D., Dai, Z.: Variational information distillation for knowledge transfer. In: Computer Vision and Pattern Recognition (2019)
2. Arjovsky, M., Bottou, L., Gulrajani, I., Lopez-Paz, D.: Invariant risk minimization. arXiv preprint arXiv:1907.02893 (2019)
3. Bahng, H., Chun, S., Yun, S., Choo, J., Oh, S.J.: Learning de-biased representations with biased representations. In: International Conference on Machine Learning (2020)
4. Bai, H., et al.: Decaug: Out-of-distribution generalization via decomposed feature representation and semantic augmentation. In: AAAI Conference on Artificial Intelligence (2021)
5. Balaji, Y., Sankaranarayanan, S., Chellappa, R.: Metareg: Towards domain generalization using meta-regularization. In: Neural Information Processing Systems (2018)
6. Bandi, P., et al.: From detection of individual metastases to classification of lymph node status at the patient level: the camelyon17 challenge. IEEE Trans. Med. Imaging **38**(2), 550–560 (2018)
7. Barber, D., Agakov, F.: The im algorithm: a variational approach to information maximization. In: Neural Information Processing Systems (2004)
8. Beery, S., Van Horn, G., Perona, P.: Recognition in Terra incognita. In: Ferrari, V., Hebert, M., Sminchisescu, C., Weiss, Y. (eds.) ECCV 2018. LNCS, vol. 11220, pp. 472–489. Springer, Cham (2018). https://doi.org/10.1007/978-3-030-01270-0_28
9. Belghazi, M.I., et al.: Mutual information neural estimation. In: International Conference on Machine Learning (2018)
10. Blanchard, G., Deshmukh, A.A., Dogan, U., Lee, G., Scott, C.: Domain generalization by marginal transfer learning. J. Mach. Learn. Res. **22**(2), 1–55 (2021)
11. Bui, M.H., Tran, T., Tran, A., Phung, D.: Exploiting domain-specific features to enhance domain generalization. In: Neural Information Processing Systems (2021)
12. Carlucci, F.M., D'Innocente, A., Bucci, S., Caputo, B., Tommasi, T.: Domain generalization by solving jigsaw puzzles. In: Computer Vision and Pattern Recognition (2019)
13. Cha, J., et al.: Swad: Domain generalization by seeking flat minima. In: Neural Information Processing Systems (2021)
14. Chattopadhyay, P., Balaji, Y., Hoffman, J.: Learning to balance specificity and invariance for in and out of domain generalization. In: Vedaldi, A., Bischof, H., Brox, T., Frahm, J.-M. (eds.) ECCV 2020. LNCS, vol. 12354, pp. 301–318. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-58545-7_18
15. Chen, J., Wang, J., Lin, W., Zhang, K., de Silva, C.W.: Preserving domain private representation via mutual information maximization. arXiv preprint arXiv:2201.03102 (2022)

16. Chen, X., Xie, S., He, K.: An empirical study of training self-supervised vision transformers. In: International Conference on Computer Vision (2021)
17. Dai, D., Van Gool, L.: Dark model adaptation: Semantic image segmentation from daytime to nighttime. In: International Conference on Intelligent Transportation Systems (2018)
18. Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., et al.: An image is worth 16x16 words: Transformers for image recognition at scale. In: International Conference on Learning Representations (2021)
19. Dou, Q., Castro, D.C., Kamnitsas, K., Glocker, B.: Domain generalization via model-agnostic learning of semantic features. In: Neural Information Processing System (2019)
20. Erhan, D., Bengio, Y., Courville, A., Vincent, P.: Visualizing higher-layer features of a deep network. University of Montreal **1341**(3), 1 (2009)
21. Fang, C., Xu, Y., Rockmore, D.N.: Unbiased metric learning: On the utilization of multiple datasets and web images for softening bias. In: International Conference on Computer Vision (2013)
22. Ganin, Y., et al.: Domain-adversarial training of neural networks. J. Mach. Learn. Res. **17**(1), 2030–2096 (2016)
23. Geirhos, R., Rubisch, P., Michaelis, C., Bethge, M., Wichmann, F.A., Brendel, W.: Imagenet-trained cnns are biased towards texture; increasing shape bias improves accuracy and robustness. In: International Conference on Learning Representations (2019)
24. Gulrajani, I., Lopez-Paz, D.: In search of lost domain generalization. In: International Conference on Learning Representations (2021)
25. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Computer Vision and Pattern Recognition (2016)
26. Huang, Z., Wang, H., Xing, E.P., Huang, D.: Self-challenging improves cross-domain generalization. In: Vedaldi, A., Bischof, H., Brox, T., Frahm, J.-M. (eds.) ECCV 2020. LNCS, vol. 12347, pp. 124–140. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-58536-5_8
27. Kim, D., Yoo, Y., Park, S., Kim, J., Lee, J.: Selfreg: Self-supervised contrastive regularization for domain generalization. In: International Conference on Computer Vision (2021)
28. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. In: International Conference on Learning Representations (2015)
29. Koh, P.W., et al.: Wilds: A benchmark of in-the-wild distribution shifts. In: International Conference on Machine Learning (2021)
30. Krueger, D., et al.: Out-of-distribution generalization via risk extrapolation (rex). arXiv preprint arXiv:2003.00688 (2020)
31. Kumar, A., Raghunathan, A., Jones, R., Ma, T., Liang, P.: Fine-tuning can distort pretrained features and underperform out-of-distribution. In: International Conference on Learning Representations (2022)
32. Li, D., Yang, Y., Song, Y.Z., Hospedales, T.: Learning to generalize: Meta-learning for domain generalization. In: AAAI Conference on Artificial Intelligence (2018)
33. Li, D., Yang, Y., Song, Y.Z., Hospedales, T.M.: Deeper, broader and artier domain generalization. In: International Conference on Computer Vision (2017)
34. Li, D., Zhang, J., Yang, Y., Liu, C., Song, Y.Z., Hospedales, T.M.: Episodic training for domain generalization. In: International Conference on Computer Vision (2019)
35. Li, H., Pan, S.J., Wang, S., Kot, A.C.: Domain generalization with adversarial feature learning. In: Computer Vision and Pattern Recognition (2018)

36. Li, X., Xiong, H., Wang, H., Rao, Y., Liu, L., Huan, J.: Delta: Deep learning transfer using feature map with attention for convolutional networks. In: International Conference on Learning Representations (2019)
37. Li, Y., Gong, M., Tian, X., Liu, T., Tao, D.: Domain generalization via conditional invariant representations. In: AAAI Conference on Artificial Intelligence (2018)
38. Li, Z., Hoiem, D.: Learning without forgetting. IEEE Trans. Pattern Anal. Mach. Intell. **40**(12), 2935–2947 (2017)
39. Matsuura, T., Harada, T.: Domain generalization using a mixture of multiple latent domains. In: AAAI Conference on Artificial Intelligence (2020)
40. Michaelis, C., et al.: Benchmarking robustness in object detection: Autonomous driving when winter is coming. arXiv preprint arXiv:1907.07484 (2019)
41. Muandet, K., Balduzzi, D., Schölkopf, B.: Domain generalization via invariant feature representation. In: International Conference on Machine Learning (2013)
42. Nam, H., Lee, H., Park, J., Yoon, W., Yoo, D.: Reducing domain gap by reducing style bias. In: Computer Vision and Pattern Recognition (2021)
43. Nuriel, O., Benaim, S., Wolf, L.: Permuted adain: Reducing the bias towards global statistics in image classification. In: Computer Vision and Pattern Recognition (2021)
44. Peng, X., Bai, Q., Xia, X., Huang, Z., Saenko, K., Wang, B.: Moment matching for multi-source domain adaptation. In: International Conference on Computer Vision (2019)
45. Radford, A., et al.: Learning transferable visual models from natural language supervision. In: International Conference on Machine Learning (2021)
46. Radosavovic, I., Kosaraju, R.P., Girshick, R., He, K., Dollár, P.: Designing network design spaces. In: Computer Vision and Pattern Recognition (2020)
47. Robey, A., Pappas, G.J., Hassani, H.: Model-based domain generalization. In: Neural Information Processing Systems (2021)
48. Russakovsky, O., et al.: Imagenet large scale visual recognition challenge. Int. J. Comput. Vision **115**(3), 211–252 (2015)
49. Sagawa*, S., Koh*, P.W., Hashimoto, T.B., Liang, P.: Distributionally robust neural networks. In: International Conference on Learning Representations (2020)
50. Scimeca, L., Oh, S.J., Chun, S., Poli, M., Yun, S.: Which shortcut cues will dnns choose? a study from the parameter-space perspective. In: International Conference on Learning Representations (2022)
51. Shi, Y., et al.: Gradient matching for domain generalization. In: International Conference on Learning Representations (2022)
52. Singh, M., et al.: Revisiting weakly supervised pre-training of visual perception models. In: Computer Vision and Pattern Recognition (2022)
53. Sun, B., Saenko, K.: Deep CORAL: Correlation alignment for deep domain adaptation. In: Hua, G., Jégou, H. (eds.) ECCV 2016. LNCS, vol. 9915, pp. 443–450. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-49409-8_35
54. Tian, Y., Krishnan, D., Isola, P.: Contrastive representation distillation. In: International Conference on Learning Representations (2019)
55. Vapnik, V.: Statistical learning theory. Wiley, NY (1998)
56. Venkateswara, H., Eusebio, J., Chakraborty, S., Panchanathan, S.: Deep hashing network for unsupervised domain adaptation. In: Computer Vision and Pattern Recognition (2017)
57. de Vries, T., Misra, I., Wang, C., van der Maaten, L.: Does object recognition work for everyone? In: Computer Vision and Pattern Recognition Workshops (2019)

58. Wang, Y., Li, H., Kot, A.C.: Heterogeneous domain generalization via domain mixup. In: International Conference on Acoustics, Speech and Signal Processing (2020)
59. Wortsman, M., et al.: Robust fine-tuning of zero-shot models. In: Computer Vision and Pattern Recognition (2022)
60. Xiao, K.Y., Engstrom, L., Ilyas, A., Madry, A.: Noise or signal: The role of image backgrounds in object recognition. In: International Conference on Learning Representations (2020)
61. Xu, M., et al.: Adversarial domain adaptation with domain mixup. In: AAAI Conference on Artificial Intelligence (2020)
62. Xuhong, L., Grandvalet, Y., Davoine, F.: Explicit inductive bias for transfer learning with convolutional networks. In: International Conference on Machine Learning (2018)
63. Yan, S., Song, H., Li, N., Zou, L., Ren, L.: Improve unsupervised domain adaptation with mixup training. arXiv preprint arXiv:2001.00677 (2020)
64. Yang, F.E., Cheng, Y.C., Shiau, Z.Y., Wang, Y.C.F.: Adversarial teacher-student representation learning for domain generalization. In: Neural Information Processing Systems (2021)
65. Yang, K., Qinami, K., Fei-Fei, L., Deng, J., Russakovsky, O.: Towards fairer datasets: Filtering and balancing the distribution of the people subtree in the imagenet hierarchy. In: Conference on Fairness, Accountability, and Transparency (2020)
66. Zbontar, J., Jing, L., Misra, I., LeCun, Y., Deny, S.: Barlow twins: Self-supervised learning via redundancy reduction. In: International Conference on Machine Learning (2021)
67. Zhang, M., Marklund, H., Gupta, A., Levine, S., Finn, C.: Adaptive risk minimization: Learning to adapt to domain shift. In: Neural Information Processing Systems (2021)
68. Zhao, S., Gong, M., Liu, T., Fu, H., Tao, D.: Domain generalization via entropy regularization. In: Neural Information Processing Systems (2020)
69. Zhou, K., Yang, Y., Hospedales, T., Xiang, T.: Learning to generate novel domains for domain generalization. In: Vedaldi, A., Bischof, H., Brox, T., Frahm, J.-M. (eds.) ECCV 2020. LNCS, vol. 12361, pp. 561–578. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-58517-4_33
70. Zhou, K., Yang, Y., Qiao, Y., Xiang, T.: Domain generalization with mixstyle. In: International Conference on Learning Representations (2021)