DandelionNet: Domain Composition with Instance Adaptive Classification for Domain Generalization

Lanqing Hu[1,2], Meina Kan[1,2], Shiguang Shan[1,2,3], Xilin Chen[1,2]
[1]Institute of Computing Technology, Chinese Academy of Sciences, Beijing, 100090, China
[2]University of Chinese Academy of Sciences, Beijing, 100090, China
[3]Peng Cheng Laboratory, Shenzhen, 518055, China
lanqing.hu@vipl.ict.ac.cn,{kanmeina,sgshan,xlchen}@ict.ac.cn

Abstract

Domain generalization (DG) attempts to learn a model on source domains that can well generalize to unseen but different domains. The multiple source domains are innately different in distribution but intrinsically related to each other, e.g., from the same label space. To achieve a generalizable feature, most existing methods attempt to reduce the domain discrepancy by either learning domain-invariant feature, or additionally mining domain-specific feature. In the space of these features, the multiple source domains are either tightly aligned or not aligned at all, which both cannot fully take the advantage of complementary information from multiple domains. In order to preserve more complementary information from multiple domains at the meantime of reducing their domain gap, we propose that the multiple domains should not be tightly aligned but composite together, where all domains are pulled closer but still preserve their individuality respectively. This is achieved by using instance-adaptive classifier specified for each instance's classification, where the instance-adaptive classifier is slightly deviated from a universal classifier shared by samples from all domains. This adaptive classifier deviation allows all instances from the same category but different domains to be dispersed around the class center rather than squeezed tightly, leading to better generalization for unseen domain samples. In result, the multiple domains are harmoniously composite centered on a universal core, like a ***dandelion****, so this work is referred to as DandelionNet. Experiments on multiple DG benchmarks demonstrate that the proposed method can learn a model with better generalization and experiments on source free domain adaption also indicate the versatility.*

1. Introduction

In recent decades, the deep learning based methods achieves significant progress in a few fields, including object detection, face recognition, natural language process, etc. These methods mostly base on independent and identically distributed (i.i.d.) assumption. However, in real world, those unseen samples or domains usually lie in different distributions from that of the training set. In such situation, the learnt model would perform poorly on those unseen samples.

(a) Exemplars of over-twisted common feature distribution

(b) Ideal well organized multi-domain composite feature distributed like petals of dandelions

Figure 1: The importance of the design of the proposed method. Conventional common feature constraint DG methods may result in feature space twist as in (a). While an ideal well generalizable feature space should make different domains pulled closer but still preserve their individuality respectively, i.e., slightly deviated from the universal center (the green star) as illustrated in (b).

The technique of domain generalization (DG) aims to deal with the fore-mentioned problem by learning a robust model that can generalize to unseen domains. In domain generalization, the source domains and unseen domains are assumed to share the same categories and domain labels are known, while the unseen domains are unavailable during training stage.

Previous DG methods mainly do **domain invariant feature learning** between multiple domains with domain alignment constraints [46, 18, 21, 45, 41, 40, 16, 24], where the

domain invariant features are expected to be shared among source domains as well as unseen domains. However, these methods only emphasize the common information between domains, but neglect those domain specific information, which may restrict their discriminative ability. Besides, the domain invariant feature cannot be perfectly obtained by just domain alignment constraint, so the domains would be squeezed together and even twisted severely, which leads to poor generalization on unseen samples, as shown in Figure 1. To recall more beneficial features, *domain-specific feature exploration* methods as in [11, 47, 7, 42] are proposed to preserve those domain specific feature by additionally disentangling it from the domain invariant feature. These approaches obtain more informative features leading to better performance. However, in these methods, the two parts of features are just considered in two different manners, i.e., the domain invariant features are tightly aligned as in those domain-invariant feature learning methods, while those domain specific features are free of any consideration of domain commonality, which is similar with freely learning domain specific models and may be with obvious domain discrepancy. But over-introduction of domain specific features may also lead to incorrect classification of unseen samples especially those differing largely from the source domains.

To get a better generalizable mode that can sufficiently exploit the complementary information among domains, in this work, we propose a method that neither tightly aligns the domains which would result in twisted mapping and loss of information, nor naturally considers them separately to preserve the favorable domain specific feature, but organizes the multiple domains together where all domains are pulled closer but still preserve their individuality respectively. To ensure that all domains are composite together rather than squeezed, each sample is classified via an instance adaptive classifier, which is slightly deviated from the universal classifier. This adaptive classifier deviation allows all instances of the same category but with different variations including intra-class and domain-related ones to be dispersed around the class centers rather than roughly squeezed together, leading to better generalization for unseen samples.

Briefly, the main contributions of this work lie in:

1. We propose a new method that integrates rather than tightly aligns the multiple domains for domain generalization. Specifically, the source domains are composite together by allowing each instance being classified by instance-adaptive classifier. The instance-adaptive classifier is slightly deviated from the universal classifier, in order to pull all domains closer but still preserve their individuality respectively.

2. To prevent the learnt instance-adaptive classifier deviation from being too large which has negative influence on correct classification, the deviation norm scaling hyper-parameter is introduced to help control the shifting degree.

3. The experimental analyses verify the effectiveness of this method. Besides, they also demonstrate that our predicted classifier deviation keeps category semantic and domain relation information, showing the rationality of the method and results.

2. Related Works

Previous DG methods mainly exploit multiple source domains and extract robust domain invariant features via **domain alignment constraints** [46, 18, 21, 45, 41, 40, 16, 24]. The earlier classical approach DICA [46] is motivated by a learning-theoretic analysis that reducing cross-domain dissimilarity improves the model generalization ability and a kernel-based optimization algorithm is proposed to minimize the dissimilarity across domains. But DICA simply considers marginal distribution gap without conditional distribution shift. Therefore the method in [40] is proposed to further take conditional distribution shift into account. Afterwards, as the development of deep learning and adversarial training, CIDDG [41] deploys a class-conditional adversarial network to align the conditional distributions of source domains.

Then to further improve the model robustness for more various inputs, **augmentation based approaches** are proposed [51, 57, 74, 75, 67, 26, 66, 39, 48, 62, 30]. The work in [57] generates hard adversarial examples to reinforce the model to handle more difficult scenarios. L2A-OT [74] maps source domain samples into pseudo-novel domains via optimal transport divergence maximization to break the limitation of training data diversity to some extent. Schemes from [39, 75, 47, 7, 62, 30] regard feature mean and variance as domain specific information and then augment domain samples via random selection and mixture of the feature statistics. There are also methods attempting domain specific or instance specific normalization to do individual feature mapping into shared feature space. Fourier analysis based methods [26, 66] factorize spatial images into multiple frequency components, re-weight these components in a random or exchange manner, and then recombine the components to augment the initial images. Besides augmentation, methods in [11, 47, 7, 42] endeavor to improve the model generalization capability for varied domains by preserving more **domain specific task-related feature** via feature disentanglement and meta learning. **Model ensembling** [14, 61, 9] is another way to distill knowledge from multiple different source domains. RMOE [14] is expected to merge knowledge from different perspectives. SWAD [9] ensembles models from sequential iterations to skip out from the local optima and find a better solution.

Apart from the feature invariance and specificity exploration approaches, **meta learning based** methods like

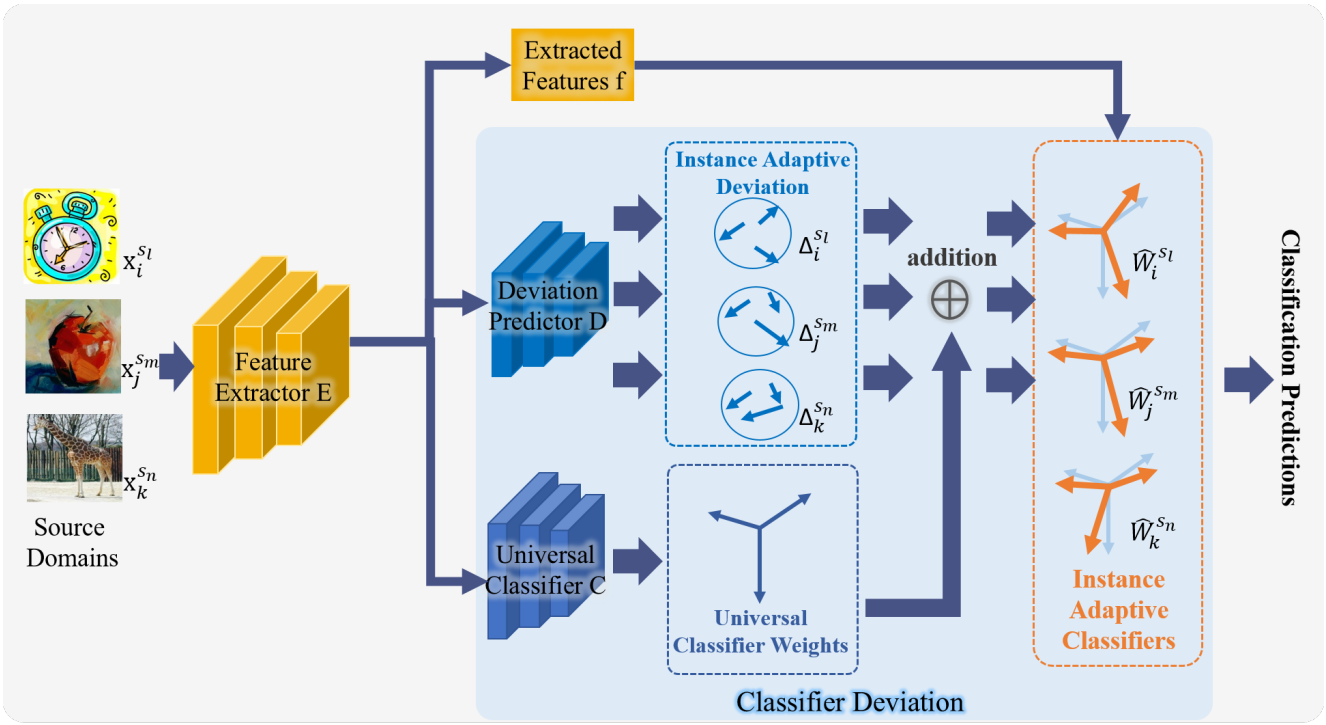

Figure 2: The overview of the proposed method. Our work consists of three modules including feature extractor E, a universal classifier C and an instance-adaptive classifier deviation predictor D. The predicted deviation is added to the universal classifier and then a slightly deviated classifier can be used for each specific input sample. Then the obtained feature can be composite rather than tightly aligned.

[4, 72, 38, 16, 48, 7] try to obtain better generalized models by enhancing optimization process. Specifically, MetaReg [4] is one of the typical meta learning based method, which splits the training source domains into meta-train and meta-test set and makes the model trained on meta-train to perform well on meta-test, thus enforcing the model to generalize decently to unseen domains. Meta learning based methods can be also regarded as adding a regularization term for optimizing target, and there are also a lot of general **regularization term based** methods [55, 2, 59, 50, 13, 44, 31, 52, 49]. IRM [2] minimizes invariant risk to regularize more robust invariant features for better generalization. In [52], the approach focuses on gradient consistency constraint to ensure more accurate common feature extraction.

As with the development of self-supervised pretraining technique, approaches applying **self-supervised training** are proposed to enhance the feature modeling ability for domain generalization [8, 60, 6, 35, 30, 31, 70]. JIGEN [8] is an earlier self-supervised learning DG method which designs a novel jigsaw task for improving comprehensive feature learning. PCL [70] is also a kind of self-supervised learning method from contrastive learning perspective, which finds that directly applying contrastive loss is not efficient enough and thus replacing the original sample-to-sample relations with proxy-to-sample relations.

Source free domain adaptation can be seen as a post-process of domain generalization, where the pretrained models with source domains are expected to adapt to target domains. A popular solution is using pseudo labeling [43, 58, 54, 68, 69] which exploit information entropy [43, 58], uncertainty [54] or neighbor structure [68, 69] to help more accurate pseudo label deduction. There also feature distribution alignment methods [71, 64, 17, 34] which try to map new target domain to the learnt source domain feature space. Method proposed in [34] improves both the source training and target adapting process by using an augmentation invariance constraint to ensure generalizable feature mapping. As the self-supervised learning shows powerful capability of modeling comprehensive features, self-supervised training including adaptive contrastive learning are also proposed to improve adaptation accuracy [27, 12, 73].

3. Method

In the problem of domain generalization, a single or multiple labeled source domains are denoted as $\{S_i\}_{i=1}^{N_S}$, where each domain S_i contains n_i samples with corresponding cat-

egory labels, i.e., $S_i = \{(x_k^{s_i}, y_k^{s_i})\}_{k=1}^{n_i}$. All source domains share the same task labels, i.e., $y_k^{s_i} \in \{1, 2, \cdots, c\}$. The goal is to train a model by using samples from all source domains, which can generalize well to unseen target domains $\{T_j\}_{j=1}^{N_T}$. Here, the target domains also share the same task labels with the source domains.

As analyzed in introduction, the model for domain generalization is expected to not only be shared by multiple domains, but also integrate the complementary information from multiple domains. To this end, we propose a domain composition method that can simultaneously pull instances of multiple domains closer but still preserve their individuality. Particularly, a universal classifier is shared by samples from all domains to carry the common information among domains, while an instance-adaptive classifier is used to classify each sample. Each instance-adaptive classifier is slightly deviated from the universal classifier, and calculated as the sum of universal classifier and a small deviation. Thus, the instance-adaptive classifier allows the individuality (i.e., those complementary information) of all samples including those domain specific ones to be preserved. As a result, all domains are flexibly composite together rather than tightly aligned together, inducing a robust as well as informative model for domain generalization. The details of each module are given below.

3.1. DandelionNet Training

The framework of the proposed DandelionNet is composed of three modules, including a shared feature extractor E, a universal classifier C and a classifier-deviation predictor D, illustrated in Figure 2. Specifically, the feature extractor is expected to obtain shared comprehensive representations. The universal classifier is used as the base classifier share by samples from all source domains. The classifier-deviation predictor is the newly introduced module, which are used to predict the deviation of the classifier of each instance from the universal classifier. Based on the universal classifier and classifier-deviation of each sample, the instance-adaptive classifier is finally obtained, which is used to classify the sample. The whole network is simply optimized by the cross-entropy classification loss for all training domains.

3.1.1 Feature Extractor E

Concretely, the feature extractor E is expected to transform instances from all domains into shared and comprehensive feature representations, formulated as below:

$$f_k^{s_i} = E(x_k^{s_i}), x_k^{s_i} \in S_i, i = 1, 2, \cdots, N_S. \tag{1}$$

Here, the extractor can be any types of architecture, such as the commonly used ResNet [23].

3.1.2 Universal Classifier C

The universal classifier C is shared by samples from all domains, which are expected to generalize well on unseen domains. For easy adapting to each instance, the universal classifier is designed as a fully-connected layer without bias, parametrized as W^C.

In the process of training, the classification is conducted not directly by the universal classifier C but the adaptively updated instance-adaptive classifier $\hat{C}$. After training, the universal classifier C contains shared as well as complementary favorable information, which inducing better generalization on unseen domains. Therefore, during testing, the universal classifier C is used to classify the unseen samples:

$$p_k^{t_j} = W^C \cdot f_k^{t_j}. \tag{2}$$

3.1.3 Instance-Adaptive Classifier $C_k^{s_i}$

If the universal classifier is directly used to classify samples from all domains, the samples would have to squeeze together in order to reduce the domain discrepancy. To avoid this tight squeezing, in this work an instance-adaptive classifier $C_k^{s_i}$ for each sample $x_k^{s_i}$ is introduced to. Besides, to ensure the instances from different domains but the same category still lie closer for accurate classification, the instance-adaptive classifier $C_k^{s_i}$ are enforced to surround closely w.r.t. the universal classifier. Therefore, the instance-adaptive classifier $C_k^{s_i}$ is designed as the sum of universal classifier and a small deviation.

Specifically, the deviation of each instance's classifier to the universal classifier is adaptively predicted by a predictor D, formulated as follows:

$$\Delta_k^{s_i} = D(f_k^{s_i}). \tag{3}$$

Subsequently, the parameters of instance-adaptive classifier C_i^k for instance x_i^k is calculated as:

$$\hat{W}_k^{s_i} = W^C + r * \Delta_k^{s_i}. \tag{4}$$

Finally, with the adaptive classifier, the classification result of each sample is reformulated as below:

$$\hat{p}_k^{s_i} = \hat{W}_k^{s_i} \cdot f_k^{s_i}. \tag{5}$$

As mentioned, the deviation is enforced to be small in order to ensure less classification result bias. To this end, both W^C and $\Delta_k^{s_i}$ are normalized with the L2Norm equals to 1.0, and the parameter r is set small. In this work, r is set 0.1 unless specified.

To optimize the whole network including the E, C and D, cross entropy loss of samples from all domains are used to update the whole network in end-to-end manner. The overall objective is written as below:

Algorithm 1 Optimization procedure of our method.

Input: The training set including labeled source domain training set S_i, total number of training iterations end_iter, the hyper-parameter r.

Output: The trained feature extractor E and universal classifier C.

1: **for** $i = 1$ to end_epoch **do**
2: Extract features via Equation (1).
3: Optimize instance specific classifier deviation via Equation (3).
4: Update instance specific classifier weights according to the learnt deviation (Equation (4)).
5: Do classification via the updated classifier (Equation (5)) and optimize by cross entropy loss (Equation (6)).
6: **end for**

$$L^S = \min_{\hat{W}_k^{s_i}, W^E} H\Big(\hat{W}_k^{s_i} \cdot \big(E(x_k^{s_i})\big), y_k^{s_i}\Big) = \min_{W^D, W^C, W^E} H\Big((W^C + r * \Delta_k^{s_i}) \cdot \big(E(x_k^{s_i})\big), y_k^{s_i}\Big), \tag{6}$$

where each training sample is trained according the adjusted specific semantic center, i.e., classifier weight, achieving instance-adaptive feature mapping, which facilitates favorable information preservation and over-twist reduction.

3.1.4 Why Instance-Adaptive Classifier Allows More Favorable Complementary Feature

Intuitively, in conventional classification methods that directly using the universal classifier with cross entropy loss for classification of samples from all domains during training stage, the optimal feature representation $f_k^{s_i}$ for a sample $f_k^{s_i}$ tends to be close to the classifier weight as much as possible, i.e., $f_k^{s_i} \to W^C$. This will implicitly make all samples such as $f_k^{s_i}$ and $f_l^{s_j}$ lean close to each other, leading to squeezing. Whilst with instance-adaptive classifiers, the optimal feature representation $f_k^{s_i}$ tends to be close to the adaptive classifier $\hat{W}_k^{s_i} = W^C + r * \Delta_k^{s_i}$, i.e., $f_k^{s_i} \to W^C + r * \Delta_k^{s_i}$. This equals roughly to $f_k^{s_i} + r * \Delta_k^{s_i} \to\to W^C$. This is roughly equivalent to that, any two samples $f_k^{s_i} + r * \Delta_k^{s_i}$ and $f_l^{s_j} + r * \Delta_l^{s_j}$ lean close to each other. Since $\Delta_k^{s_i}$ and $\Delta_l^{s_j}$ are adaptively determined and different, so $f_k^{s_i}$ and $f_l^{s_j}$ are also not necessary to be close, thus avoiding over-squeezing.

3.1.5 Overall Optimization

Overally speaking, our method has three parts to optimize, i.e., feature extractor E, universal classifier C and classifier deviation predictor D parameterized by W^E, W^C and W^D, respectively. The training process is presented in Algorithm 1.

Specifically, when optimizing C, the gradient is:

$$g(W^C) = \frac{\partial L^S}{\partial \hat{W}_k^{s_i}} \times \frac{\partial \hat{W}_k^{s_i}}{\partial W^C}.$$

Meanwhile, the back-propagation gradient for optimizing D should be:

$$g(W^D) = \frac{\partial L^S}{\partial \hat{W}_k^{s_i}} \times \frac{\partial \hat{W}_k^{s_i}}{\partial W^D}.$$

Then W^E is also updated by chain rule and the gradient is written as follows:

$$g(W^E) = g(W^C) \times \frac{\partial W^C}{\partial W^E} + g(W^D) \times \frac{\partial W^D}{\partial W^E}.$$

In result, the whole model is optimized to achieve a better generalized and adaptive classification network molding and well organizing more favorable features. The overall optimization process can be detailed in Algorithm 1.

3.2. DandelionNet Testing

As analyzed in previous subsection, the instance-adaptive classifier makes the universal classifier more generalizable on unseen domains. Moreover, we experimentally find that the trials using universal classifier and instance-adaptive classifier show similar accuracies on unseen domains. So, in testing phase, feature extractor E and universal classifier C are just exploited for simplicity, i.e., Equation (2).

Following [43], our method can be also applicable for source-free domain adaption via direct finetuning. Specifically, the universal classifier C and feature extractor E obtained from Equation (6) are further finetuned by using the unlabeled target domain data, with entropy minimization and information maximization loss as in [32, 53, 25, 43].

4. Experiments

To verify the effectivity of the proposed method, extensive comparisons and experimental analysis are conducted on domain generalization benchmarks. Furthermore, to verify the versatility of the proposed method to other tasks, the proposed method are also evaluated on source free domain adaptation benchmarks.

4.1. Evaluation on Domain Generalization

Following [22], we evaluate our method compared with others on various benchmarks for domain generalization: PACS [37] (9,991 images, 7 classes, and 4 domains), VLCS [19] (10,729 images, 5 classes, and 4 domains), OfficeHome [56] (15,588 images, 65 classes, and 4 domains), TerraIncognita [5] (24,788 images, 10 classes, and 4 domains), and DomainNet [63] (586,575 images, 345 classes, and 6 domains). For a fair comparison, the training and evaluation

Table 1: Comparison with domain generalization methods (ResNet-50) in terms of out-of-domain accuracies on five domain generalization benchmarks. Best and second best accuracies are higlighted with **bold** and underline.

Method	PACS	VLCS	OfficeHome	TerraInc	DomainNet	Avg
ERM [55]	84.2	77.3	67.6	47.8	44.0	64.2
IRM[2]	83.5	78.6	64.3	47.6	33.9	61.6
GroupDRO [50]	84.4	76.7	66.0	43.2	33.3	60.7
DANN [20]	83.7	78.6	65.9	46.7	38.3	62.6
CDANN [40]	82.6	77.5	65.7	45.8	38.3	62.0
VREx [13]	84.9	78.3	66.4	46.4	33.6	61.9
RSC [28]	85.2	77.1	65.5	46.6	38.9	62.7
Mixstyle [75]	85.2	77.9	60.4	44.0	34.0	60.3
I-Mixup[65]	84.6	77.4	68.1	47.9	39.2	63.4
MLDG [36]	84.9	77.2	66.8	47.8	41.2	63.6
mDISI [7]	86.2	79.0	69.2	48.1	42.8	65.1
SWAD[9]	88.1	79.1	70.6	50.0	46.5	66.9
MIRO [10]	85.4	79.0	70.5	50.4	46.0	66.5
PCL [70]	88.7	-	71.6	52.1	47.7	-
EoA [3]	88.6	79.1	72.5	52.3	47.4	68.0
Ours (r=0.1)	**89.2**	**81.6**	70.4	**54.5**	48.1	**68.8**
Ours (r=0.1)+SWAD	88.6	79.6	**73.2**	51.6	**49.0**	68.4

Table 2: Comparison on source free domain adaptation benchmarks Office-Home and DomainNet (ResNet50). Best and second best accuracies are higlighted with **bold** and underline.

Method	w/o domain labels	DomainNet							OfficeHome				
		→ C	→ I	→ P	→ Q	→ R	→ S	Avg	→ Ar	→ Cl	→ Pr	→ Rw	Avg
SHOT [43]	✗	58.6	25.2	55.3	15.3	70.5	52.4	46.2	72.2	59.3	82.8	82.9	74.3
DECISION [1]	✗	61.5	21.6	54.6	18.9	67.5	51.0	45.9	74.5	59.4	84.4	83.6	75.5
NRC [69]	✓	65.8	24.1	56.0	16.0	69.2	53.4	47.4	70.6	60.0	84.6	83.5	74.7
BDT [34]+NRC	✓	**75.4**	24.6	57.8	23.6	65.8	58.5	51.0	72.6	67.4	85.9	83.6	77.4
CSS [33]	✓	70.3	25.7	57.3	17.1	**69.9**	57.1	49.6	**75.1**	64.1	86.6	84.4	77.6
Ours	✓	71.1	**27.0**	**60.0**	**25.6**	69.6	**60.7**	**52.3**	73.1	**69.9**	**87.1**	**85.6**	**78.9**

protocol including the dataset splits in [22] are used for all methods. In all comparison with existing methods, our method exploits the ResNet-50 network as the backbone of feature extractor E and the hyper-parameter r is set as 0.1. Our method is trained with 15,000 iterations on DomainNet and trained with 5,000 iterations on other datasets. All models are optimized by using Adam optimizer with initial learning rate $5*10^{-5}$. The implementation is modified based on DomainBed [22] and fixed random seeds are used in all experiments for easier reproduction.

We report out-of-domain accuracies and their averages. As seen in Table 1, compared with the typical regularization based methods ERM [55], IRM [2], GroupDRO [50] and VREx [13], adversarial based domain invariant feature methods DANN [20] and CDANN [40] achieve better performance. Approaches leveraging mixup like [75, 65] endeavor to diversify training samples and thus improves the accuracy of complicated scenarios. However, these methods only focus on domain invariant feature even if with more complicated data augmentation. The domain specific feature mining method [7] achieves further improvement benefitted from the additional domain specific information. Compared to these existing methods, our method with the hyper-parameter r simply set as 0.1 outperforms the best method EoA [3] with an improvement of 0.8% on average of five datasets. Especially, on VLCS and TerraIncognita datasets, our method outperforms EoA [3] with an improvement of about 2%, which clearly demonstrates the superiority of our method. Considering that the domain discrepancy is different in different dataset, r can be actually set differently on dataset, so if r is tuned more finely as the best value specified for each sub-dataset and even each class, our method can achieves better performance. Besides, sensitivity of r is analyzed in the following analysis section.

SWAD [9] is a recently proposed pluggable model-assembling method which can be easily combined with other

Table 3: Hyper-parameter analysis on r for domain generalization. Best and second best accuracies are higlighted with **bold** and underline. Besides, the best choice of hyper-parameter is marked with yellow.

Method	PACS	VLCS	OfficeHome	TerraInc	DomainNet	Avg
Ours(r=0.5)	86.4	79.5	66.2	50.4	48.0	66.1
Ours(r=0.2)	88.3	80.7	69.0	52.9	48.0	67.8
Ours(r=0.1)	**89.2**	**81.6**	70.4	**54.5**	**48.1**	**68.8**
Ours(r=0.05)	88.3	**81.6**	**70.5**	**54.7**	47.9	68.6
Ours(r=0.01)	88.2	79.5	**70.5**	54.4	47.8	68.1

works. So, we also investigate to combine our method with SWAD method. As can be seen in Table 1, our method degrades when combining with SWAD on PACS, VLCS and TerraIncognita, which may be due to the smaller number of categories, while on Office-Home and DomainNet, SWAD further improves the accuracy of our method.

4.2. Evaluation on Source Free Domain Adaptation

Following SHOT [43], we conduct multi-source free domain adaptation experiments on OfficeHome [56] and DomainNet [63], to investigate the generalization ability of the learnt model for easier domain adaptation. For a fair comparison, following the same protocol in SHOT [43], all methods are firstly pretrained on multiple source domains, then are finetuned on training split of target domain, and final are tested on the testing split. The backbone of all methods is ResNet-50, and the hyper-parameter r of our method is also simply set as 0.1. The pretrained models of all methods are trained with 15,000 iterations on DomainNet and trained with 5,000 iterations on other ones. They are finetuned on target domain with 5,000 iterations on DomainNet and 1,000 iterations on others. All the pretraining as well as finetuning process are optimized by Adam with learning rate $5 * 10^{-5}$.

As shown in Table 2, our method achieves the best average performance via simple finetuning with entropy loss and information maximization loss, indicating that the pretraining model from our DandelionNet has better generalization which can be easily adapted to distinct target domains. Moreover, same as [69, 34, 33], our work does not need domain labels, which is flexible for real-world application.

4.3. Experimental Analysis

In the following, we investigate the visualization of obtained feature, performance improvement under different domain discrepancy, and sensitivity to the hyper-parameter r, to deeply analyze how our method works.

Visualization of learnt feature and deviation We firstly show the optimized features from our method to see how the multiple domains distribute in learnt feature space. For clearer presentation, RotatedMNIST [15, 29] is used, where the domain shift simply corresponds to the image rotation an-

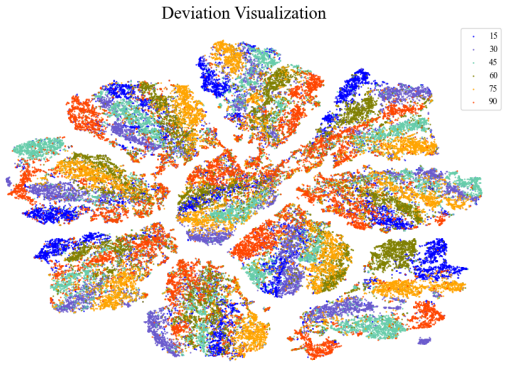

Figure 3: Visualization of learnt deviation in RotatedMNIST dataset. Different colors stand for different domains including target domain 15° (in blue). Best viewed in color.

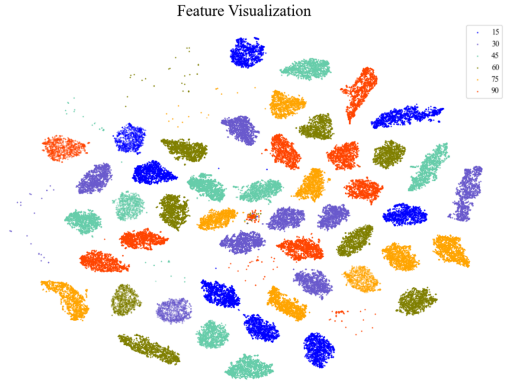

Figure 4: Visualization of learnt latent feature in RotatedMNIST dataset. Different colors stand for different domains including target domain 15° (in blue) and different shapes are for different classes. Best viewed in color.

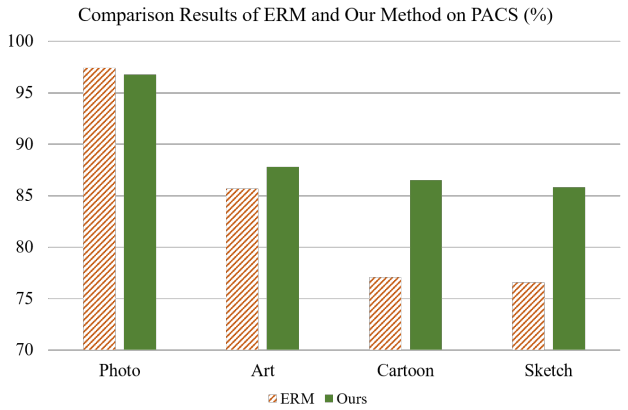

Figure 5: Classification accuracies of ERM and our method on distinct unseen domains with different domain discrepancy with training source domains.

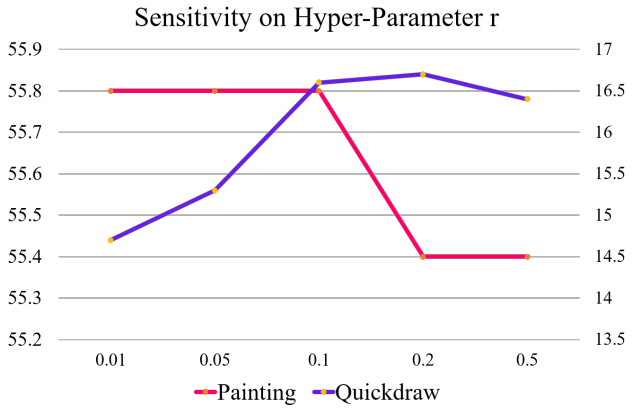

Figure 6: Accuracies with different deviation scaling hyper-parameter r on painting and quickdraw from DomainNet.

gle, i.e., consisting of 6 domains corresponding to different angles $15°, 30°, 45°, 60°, 75°, 90°$. Taking an example that $30°, 45°, 60°, 75°, 90°$ are used as source domains, while $15°$ is used as the unseen target domain. The learned deviation and feature of the source as well as the unseen target domains are visualized via TSNE in Figure 3 and 4. It can be clearly seen that the multiple domains including the unseen domain (in blue color) are pulled close but still with their individuality. Besides, the learnt deviation also shows domain composition centered on the category center, indicating that feature and our deviation are optimized as expected.

Performance improvement on domains with large discrepancy. In the available benchmarks, there are several domains with large gap which are quite challenging, such as *Cartoon* in PACS, *Sketch* in PACS and DomainNet, *Clipart* in Office-Home and DomainNet, as well as *Quickdraw* in DomainNet are quite different from the rest domains. Their appearance differs from the other domains which are mostly collected from real world photos or realistic art pictures. Our method is evaluated on PACS for an example to show how our method and the conventional baseline ERM perform. As seen, when the optimized model generalize to the *Art* and *Photo* which are visually similar to the source domains, both ERM and our method performs similarly on average. But when generalizing to *Cartoon* and *Sketch* which are quite different from the source domains, our method performs much better, even with improvement around 9%, convincingly indicating that our model contains more favorable and accurate information for better generalization.

Hyper-parameter sensitivity. The only hyper-parameter of our method is the deviation scaling parameter r. The effect of r, i.e., the ratio of norm of delta and classifier weight in Equation (3) is shown in Table 3 on the five *domain generalization* benchmarks in terms of average accuracies of multiple unseen domains. As can be seen, smaller hyper-parameter r within certain scope is favorable, e.g., our method with $r <= 0.1$ performs better than that with $r > 0.1$, which is also consistent with our motivation. Moreover, we show the detailed performance changing trend in Figure 6 on distinct unseen domains w.r.t. different r on *painting* and *quickdraw* in DomainNet dataset. As can be seen, when the unseen domain has large discrepancy with source domains (e.g., *quickdraw*), larger r performs better. While when the unseen domain has smaller discrepancy with source domains (e.g., *painting*), smaller r performs better.

5. Conclusion and future work

Aiming for better domain generalization, this work proposes a new perspective that the multiple source domains should be composite rather than tightly aligned. Specifically, a novel method called DandelionNet is proposed, which introduces an instance-adaptive classifier specified for distinct sample. The instance-adaptive classifier is slightly deviated from a universal classifier, thus allowing samples from the same category but different domains to be dispersed around the class center rather than squeezed together. As a result, the multiple domains can be harmoniously composite, organized like petals of **dandelions**. Thus, the obtained model can be more powerful for better domain generalization.

In future, we will explore more elaborate methods to get better adaptive deviation to achieve more flexible and exhaustive feature excavation for better generalization on unseen domains.

ACKNOWLEDGEMENT

This work was partially supported by the National Key R&D Program of China (2021ZD0111901), and the Natural Science Foundation of China (No. 62122074).

References

[1] S. M. Ahmed, D. S. Raychaudhuri, S. Paul, S. Oymak, and A. K. Roy-Chowdhury. Unsupervised multi-source domain adaptation without access to source data. In *CVPR*, 2021. 6

[2] M. Arjovsky, L. Bottou, I. Gulrajani, and D. Lopez-Paz. Invariant risk minimization. *arXiv preprint arXiv:1907.02893*, 2019. 3, 6

[3] D. Arpit, H. Wang, Y. Zhou, and C. Xiong. Ensemble of averages: Improving model selection and boosting performance in domain generalization. In *NeurIPS*, 2022. 6

[4] Y. Balaji, S. Sankaranarayanan, and R. Chellappa. Metareg: Towards domain generalization using meta-regularization. In *NeurIPS*, 2018. 3

[5] S. Beery, G. V. Horn, and P. Perona. Recognition in terra incognita. In *ECCV*, 2018. 5

[6] S. Bucci, A. D'Innocente, Y. Liao, and et al. Self-supervised learning across domains. *PAMI*, 44(9):5516–5528, 2021. 3

[7] M. Bui, T. Tran, A. T. Tran, and D. Phung. Exploiting domain-specific features to enhance domain generalization. In *NeurIPS*, 2021. 2, 3, 6

[8] F. M. Carlucci, A. D'Innocente, S. Bucci, and et al. Domain generalization by solving jigsaw puzzles. In *CVPR*, 2019. 3

[9] J. Cha, S. Chun, Kyungjae Lee, and et al. Swad: Domain generalization by seeking flat minima. In *NeurIPS*, 2021. 2, 6

[10] J. Cha, K. Lee, S. Park, and S. Chun. Domain generalization by mutual-information regularization with pre-trained models. In *ECCV*, 2022. 6

[11] P. Chattopadhyay, Y. Balaji, and J. Hoffman. Learning to balance specificity and invariance for in and out of domain generalization. In *ECCV*, 2020. 2

[12] D. Chen, D. Wang, T. Darrell, and et al. Contrastive test-time adaptation. In *CVPR*, 2022. 3

[13] J.-H. Jacobsen D. Krueger, E. Caballero and et al. Out-of-distribution generalization via risk extrapolation. In *ICML*, 2021. 3, 6

[14] Y. Dai, X. Li, J. Liu, and et al. Generalizable person re-identification with relevance-aware mixture of experts. In *CVPR*, 2021. 2

[15] L. Deng. The mnist database of handwritten digit images for machine learning research. *SPM*, 29(6):141–142, 2012. 7

[16] Q. Dou, D. C. Castro, K. Kamnitsas, and B. Glocker. Domain generalization via model-agnostic learning of semantic features. In *NeurIPS*, 2019. 1, 2, 3

[17] C. Eastwood, I. Mason, C. K. I. Williams, and B. Schölkopf. Source-free adaptation to measurement shift via bottom-up feature restoration. In *ICLR*, 2022. 3

[18] S. M. Erfani, M. Baktashmotlagh, M. Moshtaghi, and et al. Robust domain generalisation by enforcing distribution invariance. In *IJCAI*, 2016. 1, 2

[19] C. Fang, Y. Xu, and D. N. Rockmore. Unbiased metric learning: On the utilization of multiple datasets and web images for softening bias. In *ICCV*, 2013. 5

[20] Y. Ganin, E. Ustinova, H. Ajakan, P. Germain, and et al. Domain-adversarial training of neural networks. *JMLR*, 17(59):1–35, 2016. 6

[21] M. Ghifary, D. Balduzzi, W. B. Kleijn, and M. Zhang. Scatter component analysis: A unified framework for domain adaptation and domain generalization. *IEEE TPAMI*, 39(7):1414–1430, 2017. 1, 2

[22] I. Gulrajani and D. Lopez-Paz. In search of lost domain generalization. In *ICLR*, 2021. 5, 6

[23] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *CVPR*, 2016. 4

[24] S. Hu, K. Zhang, Z. Chen, and L. Chan. Domain generalization via multidomain discriminant analysis. In *UAI*, 2020. 1, 2

[25] W. Hu, T. Miyato, S. Tokui, and et al. Learning discrete representations via information maximizing self-augmented training. In *ICML*, 2017. 5

[26] J. Huang, D. Guan, A. Xiao, and S. Lu. Fsdr: Frequency space domain randomization for domain generalization. In *CVPR*, 2021. 2

[27] J. Huang, D. Guan, A. Xiao, and S. Lu. Model adaptation: Historical contrastive learning for unsupervised domain adaptation without source data. In *NeurIPS*, 2021. 3

[28] Z. Huang, H. Wang, E. P Xing, and D. Huang. Self-challenging improves cross-domain generalization. In *ECCV*, 2020. 6

[29] M. Jaderberg, K. Simonyan, A. Zisserman, and K. Kavukcuoglu. Spatial transformer networks. In *NeurIPS*, 2015. 7

[30] S. Jeon, K. Hong, P. Lee, and et al. Feature stylization and domain-aware contrastive learning for domain generalization. In *MM*, 2021. 2, 3

[31] D. Kim, Y. Yoo, S. Park, J. Kim, and Jaekoo Lee. Selfreg: Self-supervised contrastive regularization for domain generalization. In *ICCV*, 2021. 3

[32] A. Krause, P. Perona, and R. G. Gomes. Discriminative clustering by regularized information maximization. In *NeurIPS*, 2010. 5

[33] J. N. Kundu, S. Bhambri, and A. Kulkarni. Concurrent subsidiary supervision for unsupervised source-free domain adaptation. In *ECCV*, 2022. 6, 7

[34] J. N. Kundu, A. Kulkarni, S. Bhambr, and et al. Balancing discriminability and transferability for source-free domain adaptation. In *ICML*, 2022. 3, 6, 7

[35] H. S. Le, R. Akmeliawati, and G. Carneiro. Domain generalisation with domain augmented supervised contrastive learning. In *AAAI Student Abstract*, 2021. 3

[36] D. Li, Y. Yang, Y. Song, and T. Hospedales. Learning to generalize: Meta-learning for domain generalization. In *AAAI*, 2018. 6

[37] D. Li, Y. Yang, Y. Song, and T. M Hospedales. Deeper, broader and artier domain generalization. In *ICCV*, 2017. 5

[38] D. Li, J. Zhang, Y. Yang, and et al. Episodic training for domain generalization. In *ICCV*, 2019. 3

[39] P. Li, D. Li, W. Li, and et al. A simple feature augmentation for domain generalization. In *ICCV*, 2021. 2

[40] Y. Li, M. Gong, X. Tian, and et al. Domain generalization via conditional invariant representations. In *AAAI*, 2018. 1, 2, 6

[41] Y. Li, X. Tian, M. Gong, and et al. Deep domain generalization via conditional invariant adversarial networks. In *ECCV*, 2018. 1, 2

[42] Y. Li, D. Zhang, M. Keuper, and A. Khoreva. Intra-source style augmentation for improved domain generalization. In *WACV*, 2022. 2

[43] J. Liang, D. Hu, and J. Feng. Do we really need to access the source data? source hypothesis transfer for unsupervised domain adaptation. In *ICML*, 2020. 3, 5, 6, 7

[44] L. Mansilla, R. Echeveste, D. H. Milone, and E. Ferrante. Domain generalization via gradient surgery. In *ICCV*, 2021. 3

[45] S. Motiian, M. Piccirilli, D. A. Adjeroh, and G. Doretto. Unified deep supervised domain adaptation and generalization. In *ICCV*, 2017. 1, 2

[46] K. Muandet, D. Balduzzi, , and B. Scholkopf. Domain generalization via invariant feature representation. In *ICML*, 2013. 1, 2

[47] H. Nam, H. Lee, J. Park, and et al. Reducing domain gap by reducing style bias. In *CVPR*, 2021. 2

[48] F. Qiao, L. Zhao, and X. Peng. Learning to learn single domain generalization. In *CVPR*, 2020. 2, 3

[49] A. Rame, C. Dancette, and M. Cord. Fishr: Invariant gradient variances for out-of-distribution generalization. In *ICML*, 2022. 3

[50] S. Sagawa, P. W. Koh, and P. Liang T. B. Hashimoto. Distributionally robust neural networks for group shifts: On the importance of regularization for worst-case generalization. In *ICLR*, 2020. 3, 6

[51] S. Shankar, V. Piratla, S. Chakrabarti, S. Chaudhuri, P. Jyothi, and S. Sarawagi. Generalizing across domains via cross-gradient training. In *ICLR*, 2018. 2

[52] Y. Shi, J. Seely, P. H.S. Torr, and et al. Gradient matching for domain generalization. In *ICLR*, 2022. 3

[53] Y. Shi and F. Sha. Information-theoretical learning of discriminative clusters for unsupervised domain adaptation. In *ICML*, 2012. 5

[54] S. Prabhu Teja and F. François. Uncertainty reduction for model adaptation in semantic segmentation. In *CVPR*, 2021. 3

[55] R. Vedantam, D. Lopez-Paz, and D. J. Schwab. An empirical investigation of domain generalization with empirical risk minimizers. In *NeurIPS*, 2021. 3, 6

[56] H. Venkateswara, J. Eusebio, S. Chakraborty, and S. Panchanathan. Deep hashing network for unsupervised domain adaptation. In *CVPR*, 2017. 5, 7

[57] R. Volpi, I. I. Tecnologia, H. Namkoong, O. Sener, and et al. Generalizing to unseen domains via adversarial data augmentation. In *NeurIPS*, 2018. 2

[58] D. Wang, E. Shelhamer, S. Liu, and et al. Tent: Fully test-time adaptation by entropy minimization. In *ICLR*, 2021. 3

[59] H. Wang, Z. He, Z. C. Lipton, and E. P. Xing. Learning robust representations by projecting superficial statistics out. In *ICLR*, 2019. 3

[60] S. Wang, L. Yu, C. Li, and et al. Learning from extrinsic and intrinsic supervisions for domain generalization. In *ECCV*, 2020. 3

[61] Y. Wang, H. Li, L. Chau, and A. C. Kot. Embracing the dark knowledge: Domain generalization using regularized knowledge distillation. In *MM*, 2021. 2

[62] Z. Wang, Y. Luo, R. Qiu, and et al. Learning to diversify for single domain generalization. In *ICCV*, 2021. 2

[63] X. Xia X. Peng, Q. Bai and et al. Moment matching for multi-source domain adaptation. In *ICCV*, 2019. 5, 7

[64] H. Xia, H. Zhao, and Z. Ding. Adaptive adversarial network for source-free domain adaptation. In *ICCV*, 2021. 3

[65] M. Xu, J. Zhang, B. Ni, and et al. Adversarial domain adaptation with domain mixup. In *AAAI*, 2020. 6

[66] Q. Xu, R. Zhang, Ya Zhang, and et al. A fourier-based framework for domain generalization. In *CVPR*, 2021. 2

[67] Z. Xu, D. Liu, J. Yang, and et al. Robust and generalizable visual representation learning via random convolutions. In *ICLR*, 2021. 2

[68] S. Yang, Y. Wang, J. van de Weijer, and et al. Generalized source-free domain adaptation. In *ICCV*, 2021. 3

[69] S. Yang, Y. Y. Wang, J. Weijer, and et al. Exploiting the intrinsic neighborhood structure for source-free domain adaptation. In *NeurIPS*, 2021. 3, 6, 7

[70] X. Yao, Y. Bai, X. Zhang, and et al. Pcl: Proxy-based contrastive learning for domain generalization. In *CVPR*, 2022. 3, 6

[71] H. Yeh, B. Yang, P. C. Yuen, and et al. Sofa: Source-data-free feature alignment for unsupervised domain adaptation. In *WACV*, 2021. 3

[72] M. Zhang, H. Marklund, A. Gupta, S. Levine, and C. Finn. Adaptive risk minimization: A meta-learning approach for tackling group shift. *arXiv preprint arXiv:2007.02931*, 2020. 3

[73] Z. Zhang, W. Chen, H. Cheng, and et al. Divide and contrast: Source-free domain adaptation via adaptive contrastive learning. In *NeurIPS*, 2022. 3

[74] K. Zhou, Y. Yang, T. Hospedales, and T. Xiang. Learning to generate novel domains for domain generalization. In *ECCV*, 2020. 2

[75] K. Zhou, Y. Yang, Y. Qiao, and T. Xiang. Domain generalization with mixstyle. In *ICLR*, 2021. 2, 6