Order-preserving Consistency Regularization for Domain Adaptation and Generalization

Mengmeng Jing[1,2]*, Xiantong Zhen[2] †, Jingjing Li[1], Cees G. M. Snoek[2]
[1]University of Electronic Science and Technology of China
[2]University of Amsterdam
jingmeng1992@gmail.com zhenxt@gmail.com lijin117@yeah.net
c.g.m.snoek@uva.nl

Abstract

Deep learning models fail on cross-domain challenges if the model is oversensitive to domain-specific attributes, e.g., *lightning, background, camera angle,* etc. *To alleviate this problem, data augmentation coupled with consistency regularization are commonly adopted to make the model less sensitive to domain-specific attributes. Consistency regularization enforces the model to output the same representation or prediction for two views of one image. These constraints, however, are either too strict or not order-preserving for the classification probabilities. In this work, we propose the Order-preserving Consistency Regularization (OCR) for cross-domain tasks. The order-preserving property for the prediction makes the model robust to task-irrelevant transformations. As a result, the model becomes less sensitive to the domain-specific attributes. The comprehensive experiments show that our method achieves clear advantages on five different cross-domain tasks.*

1. Introduction

Deep neural networks have demonstrated their power in many computer vision tasks, especially when the training and test sets follow the same distribution. However, when we deploy a model in a real-world environment, we often encounter domain shifts between the training and test sets, which reduces the expected test-set performance and makes us unable to deploy with confidence [50]. For some safety-critical applications, *e.g.*, tumor recognition [22] and autonomous driving [19], a failing model is fatal.

Image data consists of a variety of attributes such as shape, color, background, texture, shooting angle, *etc.* We refer to one or more task-related attributes as *label attributes*, and the remaining irrelevant ones as *domain-specific attributes*. Wiles *et al.* [69] demonstrate the domain-specific attributes cause the distribution shifts, thus weakening the generalization of the model. Data augmentation coupled with consistency regularization is commonly employed to make a model invariant to the domain-specific attributes [68, 8, 27, 59, 4, 10, 11, 21]. Data augmentation perturbs the data so that the domain-specific information is incorporated into the perturbed image. By imposing a consistency regularization on the representations of the same image before and after perturbation, the model becomes less

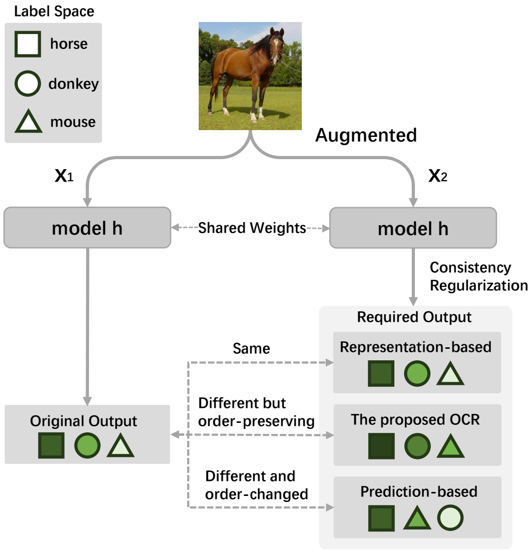


Figure 1. **The required output of three consistency regularizations.** Different shapes represent different categories. For different greens, the darker the color, the larger the classification probability. Representation-based method requires the output to be the same as the original. OCR only requires an order-preserving output and allows the output to vary. The prediction-based method is not order-preserving, which may cause the probability of the horse being classified to mouse is higher than that of donkey although donkeys are obviously more similar to horses than mice.

*This work was done when Mengmeng Jing was a visiting student at University of Amsterdam.

†Currently with United Imaging Healthcare, Co., Ltd., China.

sensitive to the domain-specific attributes.

The existing consistency regularization methods can be divided into two categories: representation-based methods [32, 61, 57] and prediction-based methods [4, 44, 71]. For the representation-based methods, usually the ℓ_1 or ℓ_2 loss is employed to enforce the model to output the same representation, even though two different views are fed into the model. This constraint, however, is too strict, which may bring difficulties to the training of the model. For example, different works on self-supervised learning [10, 11, 21] have reached a consensus that one of the representations needs to go through a non-linear prediction head before performing consistency regularization with the other. With the network model being a symmetric structure, directly imposing consistency regularization on the two representations will result in a model collapse.

Alternatively, the prediction-based methods [4, 44, 71] employ the cross-entropy loss to regularize the maximum classification probability of two representations to be the same. In other words, they ignore the order of the other classes, which would reduce the discriminability of the model. For example, consider a classification problem of three classes: *horse, donkey and mouse*. As illustrated in Fig. 1, for an image of a horse, the cross-entropy loss only regularizes the maximum classification probability of two representations to be horse, but it ignores the classification probability of donkey and mouse. If the order of classification probability is horse>donkey>mouse before augmentation, it may become horse>mouse>donkey after augmentation. Although the classification results have not changed, the discrimination of the model has reduced as donkeys are obviously more similar to horses than mice.

In view of these problems, we propose Order-preserving Consistency Regularization (OCR) for cross-domain tasks. OCR is able to enhance the model robustness to domain-specific attributes without the need of an asymmetric achitecture or a stop gradient. Specifically, we compute the residual component which is the variation in the augmented representation relative to the original representation. We postulate that if the model is robust to domain-specific attributes, the residual component should contain little or no task-related information. For example, in the classification task, when we classify the residual component, we regularize it to have the same probability to be classified into each category. In this way, the classification probabilities of the augmented representation are order-preserving compared to the original representation. As a result, the model becomes less sensitive to the domain-specific attributes. The core idea of OCR is that we allow the model to output different representations for two views of the same image, as long as the residual component contains as little task-related information as possible.

The contributions of this paper are threefold:

1. We propose Order-preserving Consistency Regularization (OCR) to enhance model robustness to domain-specific attributes. Compared with representation-based methods, OCR relaxes the constraints on model training, *i.e.*, it allows the model to output different representations for two views of the same image. Compared with prediction-based method, OCR maintains the order of the classification probabilities before and after augmentation, which helps the model to be less sensitive to the domain-specific attributes.

2. We provide a theoretical analysis for OCR. We prove that the representation-based method is a special case of OCR. Moreover, OCR can reduce the mutual information between the domain-specific attributes and the label attribute.

3. We test our method on five different cross-domain vision tasks to demonstrate the effectiveness of our method. In particular, OCR helps to enhance the robustness of the model against adversarial attacks.

2. Related Work

Consistency Regularization. Consistency regularization [2, 32, 57] is a common self-supervised learning method which enforces the model to output the same prediction even when the input is perturbed. Since it can enhance the robustness of the model to domain-variant styles, it has recently been used to address cross domain challenges [8, 68]. To generate the perturbed version of the image, some methods employ adversarial training [44] or dropout [32, 61], while others add perturbations by applying heuristic data augmentations [32, 57, 5, 71], such as color jitter, Gaussian blur, rotate, cutout, *etc*. To measure the consistency, the ℓ_1 or ℓ_2 norm [32, 61, 57] are adopted. Given the images of the original version and the perturbed version, some methods [71, 57, 8] employ the same model to extract representations for the two images, and then impose consistency regularization. We believe that this strategy is too strict, thereby increasing the difficulty of model training.

To solve this problem, some works [10, 11, 21] have designed an asymmetric architecture for the model, where one representation needs to go through an additional non-linear layer, which makes two images go through different paths in the same model. Although effective, these methods increase the complexity of the model architecture, and often require a large number of training data to unleash their power. Another line of work feeds one of the images (usually the original version) into the running average model or past model and then applies consistency regularization [32, 61]. These methods allow two versions of the images to pass through two similar yet different models, alleviating the problem of overly strict regularization to some extent. However, these methods require the storage of multiple copies of the model,

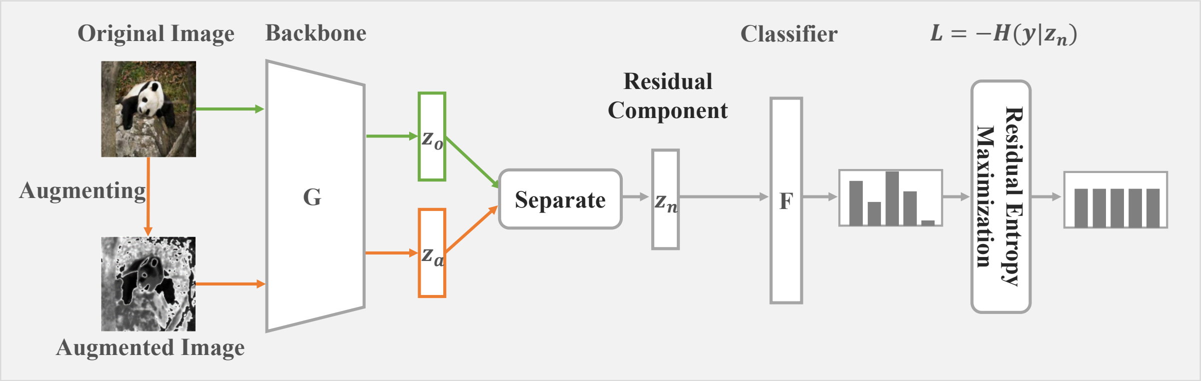


Figure 2. **Method overview**. Given the original image and its augmented counterpart, we feed them into the backbone model to obtain the original representation z_o and the augmented representation z_a. Then, we compute their residual component z_n and feed it into the classifier to get the classification results. Finally, we maximize the entropy of the classification probabilities for the residual component to reduce the task-related information in the residual component. As a result, the model become less sensitive to the domain-specific attributes.

thereby increasing the GPU memory consumption. Different from the above methods, our method does not require an asymmetric architecture, nor does it need to store multiple copies of the models. Our method allows the model to output different representations for two versions of the images, as long as the residual component obtained from these two representations does not contain task-related information.

Domain Adaptation and Generalization. Both domain adaptation (DA) [41, 56, 38, 15, 63, 79] and domain generalization (DG) [46, 18, 49, 45, 48] are cross-domain tasks, but their task settings are different. In the DA task, we are given a labeled source domain and an unlabeled target domain. We use the joint training of source and target samples to make the model adapt to the shifts between domains. The recent focus on privacy and copyright has given rise to a variant of the vanilla DA, *i.e.*, source-free domain adaptation (SFDA) [38, 72, 26, 28], where we are given a pre-trained source model but cannot directly access the source data. Based on SFDA, a new setting is proposed, namely Test-Time Adaptation (TTA) [67, 68, 8, 70]. TTA further requires that the model can adapt to the target domain in an online fashion, which is an even more challenging setting.

DG [46, 18, 49, 45, 48] trains on one or more labeled source domains to learn a model that is robust to changes in domain shifts, so that the trained model generalizes well to the (unseen) target domain. Compared to DA, DG is more difficult because during training it does not have access to (unlabeled) data from the target domain to adapt to changes in the distribution. The commonality between DA and DG is that they strive to learn the domain-invariant representations for better performance on the target domain. OCR can regularize the order of the predictions so that the model is insensitive to the domain-specific attributes, thus alleviating the domain shifts.

3. Methodology

Problem Formulation. In many computer vision challenges, be it image classification or semantic segmentation, we are given a dataset $\mathcal{D}_{train}$=$\{$x$\in\mathcal{X}_{train}$,y$\in$ $\mathcal{Y}_{train}\}$ where $\mathcal{X}_{train}$ and $\mathcal{Y}_{train}$ are the image set and label set for training and we need to establish the relationship between the data $\mathcal{X}_{train}$ and the ground-truth label $\mathcal{Y}_{train}$. In the classical Empirical Risk Minimization (ERM) [64], the training objective is to choose a hypothesis $h : \mathcal{X} \rightarrow \mathcal{Y}$ from a pre-defined hypothesis space $\mathcal{H}$ where the empirical risk is minimized w.r.t $\mathcal{D}_{train}$: $\inf_{h\in\mathcal{H}}$=$\mathbb{E}_{(x,y)\sim\mathcal{D}_{train}}[\mathcal{L}(h(x), y)]$. However, when deployed to the test set $\mathcal{D}_{test}$, the model would suffer from performance degradation since there may be domain shifts between the training set $\mathcal{D}_{train}$ and the test set $\mathcal{D}_{test}$, *i.e.*, $P(\mathcal{X}_{train}) \neq P(\mathcal{X}_{test})$. For example, samples of the same category in the training set and the test set often have varying appearance, caused by lightning conditions, camera angle, background, *etc*. These accidental attributes are irrelevant to our task, but will cause domain shifts. To generalize well, we need to train the model to be invariant to these domain-specific attributes.

Order-preserving Consistency Regularization. For a global understanding, we provide the overview of our method in Fig. 2. OCR consists of three steps, *i.e.*, data augmentation, residual component separation, and residual entropy maximization. Data augmentation [2, 57, 32, 44, 71] is a commonly used technology, which increases the diversity of samples and helps to improve the generalization of the model. Given a sample x_o, we obtain its augmentated version x_a=$\mathcal{N}(x_o)$ using transformations $\mathcal{N}$. For a clearer narration, we split the hypothesis h into two parts, *i.e.*, h=$F\circ G$, where G is the backbone model and F is the classifier. We feed x_o and x_a into G to get two different representations of the same sample: $z_o{=}G(x_o)$, $z_a{=}G(\mathcal{N}(x_o))$.

We define the residual component as the variation in the augmented representation relative to the original representation. To separate the residual component, an intuitive method is to subtract the original representation from the augmented representation. In order to control the proportion of the residual component more flexibly, here we consider the following linear relations:

$$z_a = \lambda z_o + (1-\lambda) z_n, \tag{1}$$

where z_n is the residual component, $\lambda \in (0,1)$ represents the proportion of z_o in z_a. From the perspective of Occam's razor, linearity is a good inductive bias, as also used in mixup [77]. Another reason we choose the relation in Eq. (1) is that it is an invertible operation so that we can easily infer z_n given z_a and z_o:

$$z_n = \frac{z_a - \lambda z_o}{1-\lambda}. \tag{2}$$

With the residual component z_n, we try to maximize the uncertainty of z_n's prediction so that it does not contain too much classification-related information. As the entropy can be regarded as the measure of the prediction uncertainty, we maximize the conditional entropy $\mathcal{H}(y|z_n)$ to enlarge the uncertainty of z_n's prediction. Therefore, our objective is as follows:

$$\mathcal{L}_{\text{OCR}} = -\mathcal{H}(y|z_n) = -\mathcal{H}[\text{Softmax}(F(z_n))], \tag{3}$$

where $F(z_n) \in \mathbb{R}^{B\times C}$ is the prediction of z_n, B is the batch size, C is the category number. $\mathcal{H}$ is the entropy. By minimizing Eq. (3), we regularize z_n to have equal probability of being classified into each category.

During training, we use λ to control the proportion of the residual component and the original representation in the augmented representation. λ should change dynamically to match the process of model training. At the beginning of training, the model would be sensitive to the domain-specific attributes, so the difference between z_o and z_a would be large. Then, λ should be a small value so that the proportion of z_o in z_a is lower. As the training goes on, the model gradually becomes less sensitive to the domain-specific attributes, at this time, z_o and z_a would be similar and λ should increase to a larger value accordingly. Inspired by Ganin *et al.* [16], we adopt an annealing strategy for λ:

$$\lambda = \lambda_0[1-(1+\alpha\frac{t}{T})^{-\beta}], \tag{4}$$

where α=10, β=0.75, t is the current iteration number and T is the total number of iterations. λ_0 is the initial value of λ. In this way, λ is more likely to be sampled to a small value at the beginning of training and then gradually becomes larger as the training goes on. In the ablations we illustrate that this strategy could achieve better performance than that of a fixed λ value.

Now we prove three properties of the proposed OCR:

(1) OCR is order-preserving. In previous methods with consistency regularization, *e.g.*, [8, 57, 71], the similarity between the representations and the prototype feature of a class in classifier F is computed as:

$$\hat{y}_o^i = \text{sim}(P_i, z_o), \hat{y}_a^i = \text{sim}(P_i, z_a), \tag{5}$$

where $\text{sim}(\cdot)$ is the similarity function, *i.e.*, the inner-product, P_i is the prototype feature of class i, $\hat{y}_o^i$ and $\hat{y}_a^i$ are probabilities of representations z_o and z_a belonging to class i, respectively. When substituting Eq. (1) into Eq. (5), we get:

$$\begin{aligned}\hat{y}_a^i &= \text{sim}(P_i, \lambda z_o + (1-\lambda)z_n)\\ &= \lambda\text{sim}(P_i, z_o) + (1-\lambda)\text{sim}(P_i, z_n)\\ &= \lambda\hat{y}_o^i + (1-\lambda)\hat{y}_n^i.\end{aligned} \tag{6}$$

In Eq. (3), when the conditional entropy $\mathcal{H}(y|z_n)$ is maximized, the residual component will have equal probability of being classified into each category, *i.e.*, $\hat{y}_n^1 = \hat{y}_n^2 = \cdots = \hat{y}_n^C = K$. Therefore, the relation between $\hat{y}_a^i$ and $\hat{y}_o^i$ is:

$$\hat{y}_a^i = \lambda\hat{y}_o^i + (1-\lambda)K = f(\hat{y}_o^i; \lambda, K). \tag{7}$$

Within the same iteration, K and λ are two constants. Then, $f(\hat{y}_o^i; \lambda, K)$ is an order-preserving mapping, which guarantees that if $\hat{y}_o^j > \hat{y}_o^k$, then $\hat{y}_a^j > \hat{y}_a^k$. Therefore, OCR is order-preserving.

(2) Representation-based consistency regularization is a special case of OCR. Previous cross-domain methods, *e.g.*, [71, 57, 8], optimize the ℓ_1 or ℓ_2 loss to impose consistency regularization between z_o and z_a. We use $\hat{z}_n{=}z_a - z_o$ to represent the unnormalized residual component, Δy^i to represent the prediction of $\hat{z}_n$ belonging to class i. Therefore, the objective of representation-based consistency regularization is to make $\hat{z}_n$ close to the zero vector:

$$\hat{z}_n = \mathbf{0} \Rightarrow \text{sim}(P_i, \hat{z}_n) = 0 \Rightarrow \Delta y^1 = \cdots = \Delta y^C = 0. \tag{8}$$

We believe this regularization is too strict and may increase the training difficulty. It is very reasonable for the model to output different representations for different images. The goal of our method is *not* to enforce $\hat{z}_n$ to be close to the zero vector, but to make $\hat{z}_n$ contain no task-related information. Our method relaxes the constraint in Eq. (8) as:

$$\Delta y^1 = \Delta y^2 = \cdots = \Delta y^C. \tag{9}$$

Obviously, $\hat{z}_n$ of the zero vector in Eq. (8) can also match the condition in Eq. (9), making representation-based consistency regularization a special case of our method.

Table 1. **Overview** of tasks, datasets, backbones and evaluation metrics.

Task	Dataset	Backbone	Evaluation metric
Domain Adapatation Classification	Office-Home	ResNet-50	Accuracy
Test-Time Adaptation	CIFAR100-C	ResNeXt-29	Accuracy
Domain Generalization Classification	PACS	ResNet-18	Accuracy
Domain Generalization Segmentation	GTAV, SYNTHIA, Cityscapes BDD100K, Mapillary	DeepLabV3+	mIoU
Domain Adaptation Object Detection	Cityscapes, FoggyCityscapes	ResNet-50	mAP

(3) OCR can make the model less sensitive to the domain-specific attributes. The mutual information between the residual component and the label attribute is:

$$I(Z_n;Y) = KL[p(z_n,y)||p(z_n)p(y)] \tag{10}$$

$$= \int dz_n\, dy\, p(z_n,y) \log \frac{p(z_n,y)}{p(z_n)p(y)} \tag{11}$$

$$= \int dz_n\, dy\, p(z_n,y) \log \frac{p(y|z_n)}{p(y)} \tag{12}$$

$$= \mathcal{H}(Y) - \mathcal{H}(Y|Z_n), \tag{13}$$

where $\mathcal{H}$ denotes the entropy, Y is the label set and Z_n is the residual component set, $z_n \in Z_n$, $y \in Y$. Note that $\mathcal{H}(Y)$ is independent of our optimization procedure and so can be ignored. Then, we have:

$$\min_{z_n} I(Z_n;Y) = \min_{z_n} -\mathcal{H}(Y|Z_n) = \max_{z_n} \mathcal{H}(Y|Z_n). \tag{14}$$

Therefore, by minimizing Eq. (3), we are just minimizing the mutual information between the residual component and the label attribute. As data augmentation imposes various task-irrelevant transformations to introduce domain-specific attributes for the original representation and correspondingly generates the residual component, the residual component can be regarded as the proxy of domain-specific attributes. Minimizing the mutual information in Eq. (14) can decorrelate the domain-specific attributes and the label attribute. As a result, the problem of sensitivity to domain-specific attributes is alleviated.

According to the Information Bottleneck principle [62], an optimal representation z of input x should satisfy two properties: sufficiency and minimality. Achille and Soato [1] have demonstrated that being invariant to domain-specific attributes is helpful to guarantee the minimality. Therefore, the proposed OCR is helpful to learn a better representation, which could improve the performance of the model on cross-domain tasks.

4. Experiments

4.1. Tasks, Datasets and Setup

To evaluate our method, we consider five different cross-domain tasks: domain adaptation, test-time adapatation, domain generalization classification, domain generalization detection, and domain generalization semantic segmentation. Different tasks involve different datasets and setups, which we summarize in Table 1.

Domain Adaptation Classification. For domain adaptation classification we report on *Office-Home* [65]. It consists of four domains: Art, Clipart, Product and Real-world. There are about 15,500 images categorized into 65 classes. We consider two different settings, *i.e*., source-dependent [41, 56] and source-free [38, 72]. For the source-dependent setting, we use all labeled source samples and all unlabeled target samples for training. For the source-free setting, only the model trained in the source domain and the unlabeled target samples are given. Upon evaluation, we test the models in the unlabeled target samples. For the hyper-parameter, we set λ_0=0.7.

Test-Time Adaptation. For test-time adaptation we report on *CIFAR100-C* [24]. This dataset includes 15 different corruption types with five levels of severity categorized into 100 classes. These corruptions were added to clean images from CIFAR100 [30]. There are 10,000 images for each corruption type. We used the ResNeXt-29 model pre-trained in the clean CIFAR100 dataset from [25]. This task involves two settings: online [67] and continual online [68]. In both settings, we conduct the experiments on CIFAR100-C in an online fashion without the need for labels. The difference between the two settings is that the online setting will initialize the model to the state of pre-training on the clean dataset before adapting to each corruption type, while the continual online setting will continuously adapt data of different corruption types. In this task, we evaluate our method on images with the highest severity, *i.e*., level 5. For the hyper-parameter, we set λ_0=0.8.

Domain Generalization Classification. For this task we report on *PACS* [34]. A commonly used domain generalization benchmark which includes four domains: Art Painting, Cartoon, Photo and Sketch. There are 9,991 images categorized into seven classes. We train the model on 3 of 4 domains and evaluate it on the remaining one. In this task, we set λ_0=0.5.

Domain Generalization Semantic Segmentation. For this task we follow the *Semantic segmentation benchmark* [12], which includes five datasets. *GTAV* [54] is a large-scale synthetic dataset consisting of 24,966

Table 2. **Domain Adaptation.** Accuracy (%) on Office-Home with ResNet-50 backbone. All per-domain results are in the supplementary material. R- and P-Cons. Reg. means representation-based and prediction-based consistency regulariztion. Results with are implemented by us.

Method	Mean
Source-Use	
MCD [56]	64.1
w/ OCR [56]	66.6
CDAN [41]	65.8
w/ OCR	68.0
Source-Free	
ResNet-50 [23]	46.1
Source-only	60.2
NRC [72]	71.9
w/ R-Cons. Reg.	71.5
w/ P-Cons. Reg.	72.1
w/ OCR	**72.6**
SHOT [38]	71.8
w/ R-Cons. Reg.	71.4
w/ P-Cons. Reg.	72.0
w/ OCR	**72.8**
SHOT++ [39]	72.8
w/ OCR	**73.2**

Table 3. **Test-Time Adaptation.** Test error (%) for CIFAR100-to-CIFAR100C adaptation. The backbone model is ResNeXt-29. The corruption severity is 5. OCR can improve the baselines on both online setting and continual online setting.

	Online	Continual Online
Source	46.4	46.4
BN Adapt [37]	35.8	35.4
TENT [67]	34.4	35.6
w/ OCR	**31.3**	**32.4**
CoTTA [68]	36.8	32.5
w/ OCR	**34.6**	**31.6**

driving-scene images generated from the Grand Theft Auto V game. There are 19 objects in the images. *SYNTHIA* [55] is another synthetic dataset containing 9,400 photo-realistic synthetic images with a resolution of 960×720. *Cityscapes* [13] is a large-scale real-world dataset consisting of 3,450 finely-annotated images and 20,000 coarsely-annotated images collected from urban scenes of 50 cities in Germany. We use the finely-annotated set for training and testing. *BDD-100K* [75] is also a real-world dataset which consists of urban driving scene images collected from the US. We use 7,000 images for training and 1,000 images for evaluation. *Mapillary* [47] is the last real-world dataset containing 25,000 images collected from locations around the world. For this task, we follow the protocol in [12]. Specifically, the model is trained in GTAV for 40K iterations and evaluated on the remaining datasets. In this task, we set λ_0=0.1.

Table 4. **Domain Generalization Classification.** Accuracies (%) on PACS. Results are based on the leave-one-domain-out protocol [81], where for each task we use 3 of the 4 domains as the source and the remaining 1 as the target, *e.g.*, ”Art” means ”Cartoon, Photo, Sketch→Art”. R- and P-Cons. Reg. means representation-based and prediction-based consistency regulariztion.

	PACS				
	Art	Cartoon	Photo	Sketch	Mean
MMD-AAE [35]	75.2	72.7	96.0	64.2	77.0
CCSA [45]	80.5	76.9	93.6	66.8	79.4
JiGen [6]	79.4	75.3	96.0	71.6	80.5
Metareg [3]	83.7	77.2	95.5	70.3	81.7
L2A-OT [80]	83.3	78.2	96.2	73.6	82.8
ResNet-18 [23]	77.5	77.9	96.1	70.7	80.6
w/ Manifold Mixup [66]	75.6	70.1	93.5	65.4	76.2
w/ Cutout [14]	74.9	74.9	95.9	67.7	78.3
w/ Cutmix [76]	74.6	71.8	95.6	65.3	76.8
w/ Mixup [77]	76.8	74.9	95.8	66.6	78.5
w/ DropBlock [17]	76.4	75.4	95.9	69.0	79.2
w/ MixStyle [81]	82.3	79.0	**96.3**	73.8	82.8
w/ R-Cons. Reg.	77.9	78.6	93.5	78.6	82.2
w/ P-Cons. Reg.	79.2	80.2	95.9	79.3	83.7
w/ OCR	84.4	80.7	95.9	80.8	**85.5**
IIB [33]	79.5	80.3	96.0	79.8	83.9
w/ OCR	85.1	80.9	96.2	81.8	**86.0**
SFA [36]	81.2	77.8	93.9	73.7	81.7
w/ OCR	84.5	80.5	96.1	81.2	**85.6**
SelfReg [29]	82.3	78.4	96.2	77.5	83.6
w/ OCR	85.5	80.9	96.2	81.4	**86.0**
CIRL [42]	86.1	81.0	95.9	**82.7**	86.3
w/ OCR	**86.3**	**81.5**	96.1	82.4	**86.6**

Domain Adaptation Object Detection. In this task, we report on *Cityscapes* [13] and *FoggyCityscapes* [58]. FoggyCityscapes [58] is a synthetic foggy dataset based on Cityscapes. Each image is rendered with a clear Cityscapes image and the depth map. There are 8 categories in both domains. For the hyper-parameter, we set $\lambda = 0.5$.

For the data augmentations used in our method, we apply `RandomCrop` and `RandomHorizontalFlip` for the original image. For the augmented images, we further apply `ColorJitter`, `RandomGrayscale` and `GaussianBlur`. The detailed parameters for these augmentations are in the supplementary material.

To test the effectiveness of our method, in all the cross-domain tasks, our method is inserted into the existing method as a plug-and-play module. We choose the weight of OCR through importance-weighted cross validation [60]. Our method is implemented with PyTorch [52] and MindSpore[1]. Code is available at https://github.com/mmjing/OCR.

[1] https://www.mindspore.cn

Table 5. **Domain Generalization Semantic Segmentation.** All models are trained on GTAV and evaluated on BDD100K, Cityscapes, SYNTHIA and Mapillary. We use ResNet-50 with output stride 16. Results with † are from [12]. Best mIoU (%) results highlighted in bold. OCR can improve all the baseline methods.

Source	GTAV				Mean	Boost
Target	BDD100K	Cityscapes	SYNTHIA	Mapillary		
†DeepLabv3+ [9]	25.1	29.0	26.2	28.2	27.1	
w/ OCR	34.7	34.8	25.1	39.8	33.6	6.5 ↑
†IBN-Net [51]	32.3	33.9	27.9	37.8	33.0	
w/ OCR	34.9	**41.7**	27.6	38.7	**35.7**	2.7 ↑
†RobustNet [12]	35.2	36.6	**28.3**	**40.3**	35.1	
w/ OCR	**37.2**	38.9	27.0	39.7	**35.7**	0.6 ↑

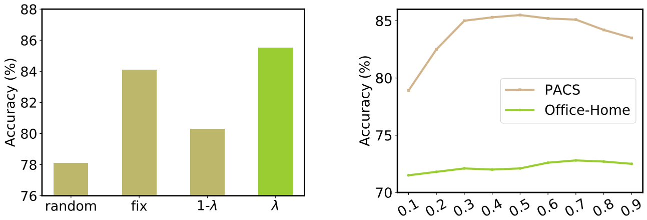
Table 6. **Domain Adaptation Object Detection.** mAP (%) on Cityscapes → FoggyCityscapes.

Method	mAP	Boost
SDAT [53]	37.5	
w/ OCR	**39.1**	1.6 ↑
SUDA [78]	42.8	
w/ OCR	**44.2**	1.4 ↑

4.2. Results

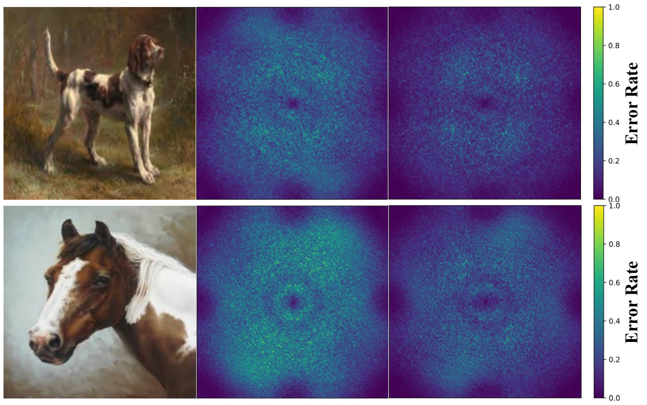
Domain Adaptation. In Table 2, OCR is inserted as a plug-and-play module in each of the compared methods. In the source-dependent setting, OCR improves MCD [56] by 2.5% and CDAN [41] by 2.2%. In the source-free setting, OCR is still effective, it improves NRC [72] by 0.7% and SHOT [38] by 1.0%. In addition, as a comparison, OCR outperforms the prediction-based consistency regularization. We observe that the representation-based method does not offer clear advantage over the baseline NRC [72] and SHOT [38], which may be due to the strict regularization that increases the difficulty of model training. OCR is feature-based and independent of specific architectures, so it can be applied to transformer-based methods as well. We test SDAT [53] (ViT-B/16) and SDAT with OCR on Office-Home and achieve results of 84.3% and 85.0%, respectively. OCR can also achieve performance improvements on transformer-based architectures.

Test-Time Adaptation. In Table 3, for the online setting, OCR achieves 3.1% and 2.2% lower test errors than TENT [67] and CoTTA [68], respectively. For the continual online setting, TENT [67] and CoTTA [68] are also improved by 3.2% and 0.9% after adding OCR. This shows that OCR can enhance the robustness of the model against various types of corruptions. In fact, the augmented data can be regarded as data with corruption applied. Our OCR can effectively reduce the task-related information residing in the residual component in the augmented representations, thus enhancing the robustness of the model.



(a) Different strategies for λ (b) Initial value λ_0

Figure 3. **Parameter λ Analysis on PACS.** (a) The strategy in Eq. (4) achieves best performance. (b) Different tasks require different initial values.



(a) Input Image **(b) Trained with ERM** **(c) Trained with OCR**

Figure 4. **Fourier Perspective.** Model sensitivity to additive noise aligned with different Fourier basis vectors on PACS (Art). The pixels closer to the center in the heat map represent the impact of low frequency noise, while the pixels outward represents the impact of high frequency noise. The model trained with OCR is more robust compared with the model learned by ERM.

Domain Generalization Classification. In Table 4, OCR outperforms the vanilla ResNet-18 with a large margin. Note that Mixup and Manifold Mixup do not improve the vanilla ResNet-18. The reason why Mixup is ineffective here is because Mixup mainly encourages the model to be robust to the combination of the existing patterns, but does not enhance the ability to handle the unseen styles. MixStyle regularizes the model to be robust to the unseen

Table 7. **Analysis on Other Layers.** Accuracies (%) on Office-Home, where "Input" means we apply OCR on pixel-level and "BT*" is the bottleneck block of ResNet-50. Generally, the deeper the layer, the more effective OCR will be.

ResNet-50 layers						
Input	Conv1	BT1	BT2	BT3	BT4	FC
68.5	69.0	70.4	70.6	71.2	72.2	72.8

styles, however, it does not explicitly minimize the domain-specific information in the representation, leading to its worse performance than OCR. We also test our method on PACS based on ResNet-50 and SWAD [7]. OCR improves SWAD from 87.8% to 88.5%. OCR achieves consistent performance advantages on both ResNet-18 and ResNet-50.

Domain Generalization Semantic Segmentation. In Table 5, the Baseline is DeepLabV3+ [9]. OCR improves the Baseline by 6.5%. For IBN-Net [51], the improvement is 2.7%, which is also impressive. For RobustNet [12], we observe that OCR has a small improvement of 0.6%, this may be because RobustNet also enhances the generalization of the model by eliminating domain specific information. OCR, however, is different from RobustNet since RobustNet disentangles the domain-specific and domain-invariant part in the feature covariance while OCR does this based on the assumption of linear combination.

Cross-domain Object Detection. In Table 6, we report results of OCR and the compared methods on the object detection task of City [13]→ FoggyCity [58]. OCR achieves improvements of 1.6% and 1.4% compared to SDAT [53] and SUDA [78], respectively. Therefore, OCR is effective on object detection tasks.

4.3. Ablations

Parameter λ Analysis. In Fig. 3, we provide an analysis for parameter λ. In Fig. 3 (a), we illustrate the impact of different choices for λ on PACS, where "random" indicates we choose a random value from range (0,1) as λ during each iteration, "fix" means we fix λ as 0.5, "λ" represents the strategy in Eq. (4), while "$1 - \lambda$" is the opposite strategy of "λ". As can be seen from Fig. 3 (a), using the strategy in Eq. (4) helps to train an optimal model. This result is in line with our hypothesis, *i.e.*, at the beginning of training, there is a small domain-specific ratio in the representation, so a small λ is required, as OCR continuously minimizes the domain-specific information, the domain-invariant part gradually increases, so a larger λ is required. In addition, we test the performance of OCR with the simple formulation, i.e., $z_n{=}z_a{-}z_o$, on PACS and achieve an accuracy of 84.0%, which is close to that of the fixed proportion setting in Fig. 3 (a), but lower than our formulation which obtains 85.5%. In Fig. 3 (b), we report the impact of different initial values λ_0 on performance. From Fig. 3 (b), we observe that the results of Office-Home do not fluctuate much with different intial values, its best λ_0 is around 0.7. The results of PACS, are more sensitive to different initial values, its best λ_0 is around 0.5. Therefore, different tasks need different initial values.

Analysis on Other Layers. By default we apply OCR to representations of the penultimate layer of the model. OCR can be applied to the representations of other layers as well. We show results with a ResNet-50 in Table 7, we observe that: (1) In the image level, OCR cannot achieve ideal results, which may be because some attributes of the sample, *e.g.*, lighting and shooting angle, cannot be separated in the image level; (2) In general, the deeper the layer, the more effective OCR will be. Prior works [40, 74] have found that representations extracted from the shallow layers are more generalized, while the representations extracted from the deep layers show strong task relevance. Therefore, shallow representations are not suitable for applying OCR, while deep representations can eliminate domain-specific information through OCR.

Fourier Perspective. Following [73], we investigate the sensitivity of our models to high and low frequency corruptions via a perturbation analysis in the Fourier domain. We plot the Fourier heat map in Fig. 4. The pixels closer to the center in the heat map represent the impact of low frequency noise, while the pixels outward represent the impact of high frequency noise. We observe that the model trained with OCR is more robust compared with the model learned by ERM, especially in the high frequency domain. High frequency information is often introduced by styles that vary significant across domains. Therefore, OCR can effectively eliminates the domain-specific style information.

Robustness to Adversarial Attack. In Table 8 we report the adversarial robustness of our method against various white-box attacks, including FGSM [20], BIM [31] and PGD [43]. We impose the adversarial attacks through the Adversarial Robust Tool box[2]. For fair comparison, we set the iteration number as 10, adversarial strength as 0.01 and step size as 0.01, all other parameters remain at their default values. Compared with ERM and prediction-based consistency regularization, OCR achieves the best robustness to all the three adversarial attacks. Especially for the iterative-based methods with more powerful attacks, OCR achieves accuracies of 61.6% and 61.7% against PGD and BIM, which is remarkably higher than ERM and prediction-based consistency regularization. The superior robustness of OCR against the adversarial attack derives from explicitly eliminating the negative effects of the domain-specific attributes which causes the domain shifts.

Effect of Order-preserving Property. In Table 9, we report the top 1 to top 5 accuracies on Office-Home. Com-

[2] https://github.com/advboxes/AdvBox

Table 8. **Robustness to Adversarial Attack.** Accuracy (%) on PACS after different adversarial attacks. The results are all based on the leave-one-domain-out protocol [81]. Our method is effective to enhance the model robustness to the adversarial attacks.

	PACS					
	Art	Cartoon	Photo	Sketch	Mean	Boost
No attack						
ERM Baseline	78.3	76.0	95.0	72.7	80.5	
w/ P-Cons. Reg.	79.2	80.2	95.9	79.3	83.7	3.2 ↑
w/ OCR	**85.4**	**81.1**	**96.2**	**81.2**	**86.0**	**5.5** ↑
FGSM attack [20]						
ERM Baseline	19.4	55.8	51.5	48.7	43.9	
w/ P-Cons. Reg.	32.0	63.8	66.2	66.6	57.2	13.3 ↑
w/ OCR	**45.5**	**69.5**	**73.3**	**73.2**	**65.4**	**21.5** ↑
BIM attack [31]						
ERM Baseline	16.6	55.0	40.5	41.6	38.4	
w/ P-Cons. Reg.	26.5	63.3	59.9	60.6	52.6	14.2 ↑
w/ OCR	**38.7**	**69.1**	**68.4**	**70.4**	**61.7**	**23.3** ↑
PGD attack [43]						
ERM Baseline	16.8	54.9	40.4	41.5	38.4	
w/ P-Cons. Reg.	26.1	63.4	60.1	60.8	52.6	14.2 ↑
w/ OCR	**38.5**	**69.0**	**68.2**	**70.6**	**61.6**	**23.2** ↑

Table 9. **Top 1 to top 5 accuracies (%) on Office-Home. Baseline method is SHOT.**

Acc.(%)	Baseline	Baseline+P-Cons. Reg.	Baseline+OCR
Top 1	71.8	72.0	72.8
Top 3	85.5	86.1	87.6
Top 5	89.4	90.2	92.2

Table 10. **Effect of data augmentations.**

		Removing one augmentation		
Tasks	All	ColorJitter	RandomGray.	GaussianBlur
PACS (Ours)	85.5	83.4	84.6	83.9
PACS (P-Cons. Reg.)	83.7	82.0	83.1	82.7
+PGD (Ours)	61.6	52.9	55.3	47.1
+PGD (P-Cons. Reg.)	52.6	45.2	47.6	41.5

pared with prediction-based method, OCR has more significant advantages in top 3 and top 5 accuracies, which proves that the order-preserving property in consistency regularization guarantees that even though the maximum probability category does not hit the ground-truth label, it is very likely that the label appears in the top 3 or top 5 categories.

Effect of Data Augmentations. Following the setting in Table 8, we remove ColorJitter, RandomGrayscale and GaussianBlur, respectively. The results are reported in Table 10. We observe that the combination of three augmentations can achieve the best performance. According to the practice in self-supervised learning [10], not all the combinations help improve the generalization of the model. Exploring the best combination would be a promising future work.

5. Conclusion and Future work

In this paper, we propose Order-preserving Consistency Regularization (OCR) to enhance model robustness to domain-specific attributes for cross-domain tasks. We first separate the residual component from the augmented representation. Then, we maximize the entropy of the residual component to enlarge the uncertainty of its prediction. As a result, the residual component contains little information about the task of interest, *i.e.*, the model is less sensitive to the domain-specific attributes. Throughout the experiments, we have shown that OCR enhances the generalization of the model and provides better robustness to adversarial attacks. OCR is easy to implement and can be applied to any cross-domain task to improve the performance. Like any data-augmentation based method, our proposal fails when the augmentations are completely independent of the domain gaps. Therefore, exploring the most related data augmentations for specific cross-domain tasks would be a suitable future work.

Acknowledgment

This work was supported in part by the National Natural Science Foundation of China under Grant 62250061, 62176042, 62276054, and in part by the Sichuan Science and Technology Program under Grant 2023YFG0156, and in part by CAAI-Huawei MindSpore Open Fund.

References

[1] Alessandro Achille and Stefano Soatto. Emergence of invariance and disentanglement in deep representations. *Journal of Machine Learning Research*, 19(1):1947–1980, 2018. 5

[2] Philip Bachman, Ouais Alsharif, and Doina Precup. Learning with pseudo-ensembles. In *Advances in Neural Information Processing Systems*, volume 27, 2014. 2, 3

[3] Yogesh Balaji, Swami Sankaranarayanan, and Rama Chellappa. Metareg: Towards domain generalization using meta-regularization. In *Advances in Neural Information Processing Systems*, volume 31, 2018. 6

[4] David Berthelot, Nicholas Carlini, Ekin D Cubuk, Alex Kurakin, Kihyuk Sohn, Han Zhang, and Colin Raffel. Remixmatch: Semi-supervised learning with distribution matching and augmentation anchoring. In *International Conference on Learning Representations*, 2019. 1, 2

[5] David Berthelot, Nicholas Carlini, Ian Goodfellow, Nicolas Papernot, Avital Oliver, and Colin A Raffel. Mixmatch: A holistic approach to semi-supervised learning. In *Advances in Neural Information Processing Systems*, volume 32, 2019. 2

[6] Fabio M Carlucci, Antonio D'Innocente, Silvia Bucci, Barbara Caputo, and Tatiana Tommasi. Domain generalization by solving jigsaw puzzles. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2229–2238, 2019. 6

[7] Junbum Cha, Sanghyuk Chun, Kyungjae Lee, Han-Cheol Cho, Seunghyun Park, Yunsung Lee, and Sungrae Park. Swad: Domain generalization by seeking flat minima. In *Advances in Neural Information Processing Systems*, volume 34, pages 22405–22418, 2021. 8

[8] Dian Chen, Dequan Wang, Trevor Darrell, and Sayna Ebrahimi. Contrastive test-time adaptation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 295–305, 2022. 1, 2, 3, 4

[9] Liang-Chieh Chen, Yukun Zhu, George Papandreou, Florian Schroff, and Hartwig Adam. Encoder-decoder with atrous separable convolution for semantic image segmentation. In *European Conference on Computer Vision*, pages 801–818, 2018. 7, 8

[10] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *International Conference on Machine Learning*, pages 1597–1607. PMLR, 2020. 1, 2, 9

[11] Xinlei Chen and Kaiming He. Exploring simple siamese representation learning. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15750–15758, 2021. 1, 2

[12] Sungha Choi, Sanghun Jung, Huiwon Yun, Joanne T Kim, Seungryong Kim, and Jaegul Choo. Robustnet: Improving domain generalization in urban-scene segmentation via instance selective whitening. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11580–11590, 2021. 5, 6, 7, 8

[13] Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. The cityscapes dataset for semantic urban scene understanding. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3213–3223, 2016. 6, 8

[14] Terrance DeVries and Graham W Taylor. Improved regularization of convolutional neural networks with cutout. *arXiv preprint arXiv:1708.04552*, 2017. 6

[15] Yaroslav Ganin and Victor Lempitsky. Unsupervised domain adaptation by backpropagation. In *International Conference on Machine Learning*, pages 1180–1189. PMLR, 2015. 3

[16] Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, Mario Marchand, and Victor Lempitsky. Domain-adversarial training of neural networks. *Journal of Machine Learning Research*, 17(1):2096–2030, 2016. 4

[17] Golnaz Ghiasi, Tsung-Yi Lin, and Quoc V Le. Dropblock: A regularization method for convolutional networks. In *Advances in Neural Information Processing Systems*, volume 31, 2018. 6

[18] Muhammad Ghifary, W Bastiaan Kleijn, Mengjie Zhang, and David Balduzzi. Domain generalization for object recognition with multi-task autoencoders. In *IEEE International Conference on Computer Vision*, pages 2551–2559, 2015. 3

[19] Rui Gong, Wen Li, Yuhua Chen, and Luc Van Gool. Dlow: Domain flow for adaptation and generalization. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2477–2486, 2019. 1

[20] Ian Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. In *International Conference on Learning Representations*, 2015. 8, 9

[21] Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Avila Pires, Zhaohan Guo, and Mohammad Gheshlaghi Azar. Bootstrap your own latent-a new approach to self-supervised learning. In *Advances in Neural Information Processing Systems*, volume 33, pages 21271–21284, 2020. 1, 2

[22] Hao Guan and Mingxia Liu. Domain adaptation for medical image analysis: a survey. *IEEE Transactions on Biomedical Engineering*, 69(3):1173–1185, 2021. 1

[23] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 770–778, 2016. 6

[24] Dan Hendrycks and Thomas Dietterich. Benchmarking neural network robustness to common corruptions and perturbations. In *International Conference on Learning Representations*, 2019. 5

[25] Dan Hendrycks, Norman Mu, Ekin Dogus Cubuk, Barret Zoph, Justin Gilmer, and Balaji Lakshminarayanan. Augmix: A simple method to improve robustness and uncertainty under data shift. In *International Conference on Learning Representations*, 2020. 5

[26] Jiaxing Huang, Dayan Guan, Aoran Xiao, and Shijian Lu. Model adaptation: Historical contrastive learning for unsupervised domain adaptation without source data. In *Advances in Neural Information Processing Systems*, volume 34, 2021. 3

[27] Ashraful Islam, Chun-Fu Richard Chen, Rameswar Panda, Leonid Karlinsky, Rogerio Feris, and Richard J Radke. Dynamic distillation network for cross-domain few-shot recognition with unlabeled data. In *Advances in Neural Information Processing Systems*, volume 34, pages 3584–3595, 2021. 1

[28] Mengmeng Jing, Xiantong Zhen, Jingjing Li, and Cees G. M. Snoek. Variational model perturbation for source-free domain adaptation. In *Advances in Neural Information Processing Systems*, 2022. 3

[29] Daehee Kim, Youngjun Yoo, Seunghyun Park, Jinkyu Kim, and Jaekoo Lee. Selfreg: Self-supervised contrastive regularization for domain generalization. In *IEEE/CVF International Conference on Computer Vision*, pages 9619–9628, 2021. 6

[30] Alex Krizhevsky and Geoffrey Hinton. Learning multiple layers of features from tiny images. *Technical report*, 2009. 5

[31] Alexey Kurakin, Ian J. Goodfellow, and Samy Bengio. Adversarial machine learning at scale. In *International Conference on Learning Representations*, 2017. 8, 9

[32] Samuli Laine and Timo Aila. Temporal ensembling for semi-supervised learning. In *International Conference on Learning Representations*, 2017. 2, 3

[33] Bo Li, Yifei Shen, Yezhen Wang, Wenzhen Zhu, Dongsheng Li, Kurt Keutzer, and Han Zhao. Invariant information bottleneck for domain generalization. In *AAAI Conference on Artificial Intelligence*, volume 36, pages 7399–7407, 2022. 6

[34] Da Li, Yongxin Yang, Yi-Zhe Song, and Timothy M Hospedales. Deeper, broader and artier domain generalization. In *IEEE International Conference on Computer Vision*, pages 5542–5550, 2017. 5

[35] Haoliang Li, Sinno Jialin Pan, Shiqi Wang, and Alex C Kot. Domain generalization with adversarial feature learning. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5400–5409, 2018. 6

[36] Pan Li, Da Li, Wei Li, Shaogang Gong, Yanwei Fu, and Timothy M Hospedales. A simple feature augmentation for domain generalization. In *IEEE/CVF International Conference on Computer Vision*, pages 8886–8895, 2021. 6

[37] Yanghao Li, Naiyan Wang, Jianping Shi, Jiaying Liu, and Xiaodi Hou. Revisiting batch normalization for practical domain adaptation. In *International Conference on Learning Representations Workshop*, 2017. 6

[38] Jian Liang, Dapeng Hu, and Jiashi Feng. Do we really need to access the source data? source hypothesis transfer for unsupervised domain adaptation. In *International Conference on Machine Learning*, pages 6028–6039. PMLR, 2020. 3, 5, 6, 7

[39] Jian Liang, Dapeng Hu, Yunbo Wang, Ran He, and Jiashi Feng. Source data-absent unsupervised domain adaptation through hypothesis transfer and labeling transfer. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(11):8602–8617, 2021. 6

[40] Mingsheng Long, Yue Cao, Zhangjie Cao, Jianmin Wang, and Michael I Jordan. Transferable representation learning with deep adaptation networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 41(12):3071–3085, 2018. 8

[41] Mingsheng Long, Zhangjie Cao, Jianmin Wang, and Michael I Jordan. Conditional adversarial domain adaptation. In *Advances in Neural Information Processing Systems*, pages 1640–1650, 2018. 3, 5, 6, 7

[42] Fangrui Lv, Jian Liang, Shuang Li, Bin Zang, Chi Harold Liu, Ziteng Wang, and Di Liu. Causality inspired representation learning for domain generalization. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8046–8056, 2022. 6

[43] Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks. In *International Conference on Learning Representations*, 2018. 8, 9

[44] Takeru Miyato, Shin-ichi Maeda, Masanori Koyama, and Shin Ishii. Virtual adversarial training: a regularization method for supervised and semi-supervised learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 41(8):1979–1993, 2018. 2, 3

[45] Saeid Motiian, Marco Piccirilli, Donald A Adjeroh, and Gianfranco Doretto. Unified deep supervised domain adaptation and generalization. In *IEEE International Conference on Computer Vision*, pages 5715–5725, 2017. 3, 6

[46] Krikamol Muandet, David Balduzzi, and Bernhard Schölkopf. Domain generalization via invariant feature representation. In *International Conference on Machine Learning*, pages 10–18. PMLR, 2013. 3

[47] Gerhard Neuhold, Tobias Ollmann, Samuel Rota Bulo, and Peter Kontschieder. The mapillary vistas dataset for semantic understanding of street scenes. In *IEEE International Conference on Computer Vision*, pages 4990–4999, 2017. 6

[48] Li Niu, Wen Li, and Dong Xu. Multi-view domain generalization for visual recognition. In *IEEE International Conference on Computer Vision*, pages 4193–4201, 2015. 3

[49] Li Niu, Wen Li, and Dong Xu. Visual recognition by learning from web data: A weakly supervised domain generalization approach. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2774–2783, 2015. 3

[50] Sinno Jialin Pan and Qiang Yang. A survey on transfer learning. *IEEE Transactions on knowledge and data engineering*, 22(10):1345–1359, 2009. 1

[51] Xingang Pan, Ping Luo, Jianping Shi, and Xiaoou Tang. Two at once: Enhancing learning and generalization capacities via ibn-net. In *European Conference on Computer Vision*, pages 464–479, 2018. 7, 8

[52] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems*, volume 32, 2019. 6

[53] Harsh Rangwani, Sumukh K Aithal, Mayank Mishra, Arihant Jain, and Venkatesh Babu Radhakrishnan. A closer look at smoothness in domain adversarial training. In *International Conference on Machine Learning*, pages 18378–18399. PMLR, 2022. 7, 8

[54] Stephan R Richter, Vibhav Vineet, Stefan Roth, and Vladlen Koltun. Playing for data: Ground truth from computer games. In *European Conference on Computer Vision*, pages 102–118. Springer, 2016. 5

[55] German Ros, Laura Sellart, Joanna Materzynska, David Vazquez, and Antonio M Lopez. The synthia dataset: A large collection of synthetic images for semantic segmentation of urban scenes. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3234–3243, 2016. 6

[56] Kuniaki Saito, Kohei Watanabe, Yoshitaka Ushiku, and Tatsuya Harada. Maximum classifier discrepancy for unsupervised domain adaptation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3723–3732, 2018. 3, 5, 6, 7

[57] Mehdi Sajjadi, Mehran Javanmardi, and Tolga Tasdizen. Regularization with stochastic transformations and perturbations for deep semi-supervised learning. In *Advances in Neural Information Processing Systems*, volume 29, 2016. 2, 3, 4

[58] Christos Sakaridis, Dengxin Dai, and Luc Van Gool. Semantic foggy scene understanding with synthetic data. *International Journal of Computer Vision*, 126:973–992, 2018. 6, 8

[59] Kihyuk Sohn, David Berthelot, Nicholas Carlini, Zizhao Zhang, Han Zhang, Colin A Raffel, Ekin Dogus Cubuk,

Alexey Kurakin, and Chun-Liang Li. Fixmatch: Simplifying semi-supervised learning with consistency and confidence. In *Advances in Neural Information Processing Systems*, volume 33, pages 596–608, 2020. 1

[60] Masashi Sugiyama, Matthias Krauledat, and Klaus-Robert MÃžller. Covariate shift adaptation by importance weighted cross validation. *Journal of Machine Learning Research*, 8(May):985–1005, 2007. 6

[61] Antti Tarvainen and Harri Valpola. Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results. In *Advances in Neural Information Processing Systems*, volume 30, 2017. 2

[62] Naftali Tishby, Fernando C. Pereira, and William Bialek. The information bottleneck method. In *Annual Allerton Conference on Communications, Control and Computing*, pages 368–377, 1999. 5

[63] Eric Tzeng, Judy Hoffman, Kate Saenko, and Trevor Darrell. Adversarial discriminative domain adaptation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7167–7176, 2017. 3

[64] Vladimir N. Vapnik. *Statistical Learning Theory*. Wiley-Interscience, 1998. 3

[65] Hemanth Venkateswara, Jose Eusebio, Shayok Chakraborty, and Sethuraman Panchanathan. Deep hashing network for unsupervised domain adaptation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5018–5027, 2017. 5

[66] Vikas Verma, Alex Lamb, Christopher Beckham, Amir Najafi, Ioannis Mitliagkas, David Lopez-Paz, and Yoshua Bengio. Manifold mixup: Better representations by interpolating hidden states. In *International Conference on Machine Learning*, pages 6438–6447. PMLR, 2019. 6

[67] Dequan Wang, Evan Shelhamer, Shaoteng Liu, Bruno Olshausen, and Trevor Darrell. Tent: Fully test-time adaptation by entropy minimization. In *International Conference on Learning Representations*, 2020. 3, 5, 6, 7

[68] Qin Wang, Olga Fink, Luc Van Gool, and Dengxin Dai. Continual test-time domain adaptation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022. 1, 2, 3, 5, 6, 7

[69] Olivia Wiles, Sven Gowal, Florian Stimberg, Sylvestre-Alvise Rebuffi, Ira Ktena, Krishnamurthy Dj Dvijotham, and Ali Taylan Cemgil. A fine-grained analysis on distribution shift. In *International Conference on Learning Representations*, 2021. 1

[70] Zehao Xiao, Xiantong Zhen, Ling Shao, and Cees GM Snoek. Learning to generalize across domains on single test samples. In *International Conference on Learning Representations*, 2021. 3

[71] Qizhe Xie, Zihang Dai, Eduard Hovy, Thang Luong, and Quoc Le. Unsupervised data augmentation for consistency training. In *Advances in Neural Information Processing Systems*, volume 33, pages 6256–6268, 2020. 2, 3, 4

[72] Shiqi Yang, Yaxing Wang, Joost van de Weijer, Luis Herranz, and Shangling Jui. Exploiting the intrinsic neighborhood structure for source-free domain adaptation. In *Advances in Neural Information Processing Systems*, volume 34, 2021. 3, 5, 6, 7

[73] Dong Yin, Raphael Gontijo Lopes, Jon Shlens, Ekin Dogus Cubuk, and Justin Gilmer. A fourier perspective on model robustness in computer vision. In *Advances in Neural Information Processing Systems*, volume 32, 2019. 8

[74] Jason Yosinski, Jeff Clune, Yoshua Bengio, and Hod Lipson. How transferable are features in deep neural networks? In *Advances in Neural Information Processing Systems*, pages 3320–3328, 2014. 8

[75] Fisher Yu, Haofeng Chen, Xin Wang, Wenqi Xian, Yingying Chen, Fangchen Liu, Vashisht Madhavan, and Trevor Darrell. Bdd100k: A diverse driving dataset for heterogeneous multitask learning. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2636–2645, 2020. 6

[76] Sangdoo Yun, Dongyoon Han, Seong Joon Oh, Sanghyuk Chun, Junsuk Choe, and Youngjoon Yoo. Cutmix: Regularization strategy to train strong classifiers with localizable features. In *IEEE/CVF International Conference on Computer Vision*, pages 6023–6032, 2019. 6

[77] Hongyi Zhang, Moustapha Cisse, Yann N Dauphin, and David Lopez-Paz. mixup: Beyond empirical risk minimization. In *International Conference on Learning Representations*, 2018. 4, 6

[78] Jingyi Zhang, Jiaxing Huang, Zichen Tian, and Shijian Lu. Spectral unsupervised domain adaptation for visual recognition. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9829–9840, 2022. 7, 8

[79] Weichen Zhang, Wanli Ouyang, Wen Li, and Dong Xu. Collaborative and adversarial network for unsupervised domain adaptation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3801–3809, 2018. 3

[80] Kaiyang Zhou, Yongxin Yang, Timothy Hospedales, and Tao Xiang. Learning to generate novel domains for domain generalization. In *European Conference on Computer Vision*, pages 561–578. Springer, 2020. 6

[81] Kaiyang Zhou, Yongxin Yang, Yu Qiao, and Tao Xiang. Domain generalization with mixstyle. In *International Conference on Learning Representations*, 2020. 6, 9