

# DomainDrop: Suppressing Domain-Sensitive Channels for Domain Generalization

---

**ICCV2023**

- **Domain Adaptation(DA):**

In traditional machine learning, we assume that the training and test sets are independently and identically distributed (i.i.d.). However, this assumption does not hold in real-world scenarios. The core task of domain adaptation is to enable the model to extract consistent features between the source domain and the target domain.

- **Domain Generalization(DG):**

Domain generalization involves training a model using only one or multiple related domains, such that the trained model can directly generalize to unseen target domains.

- **Difference:**

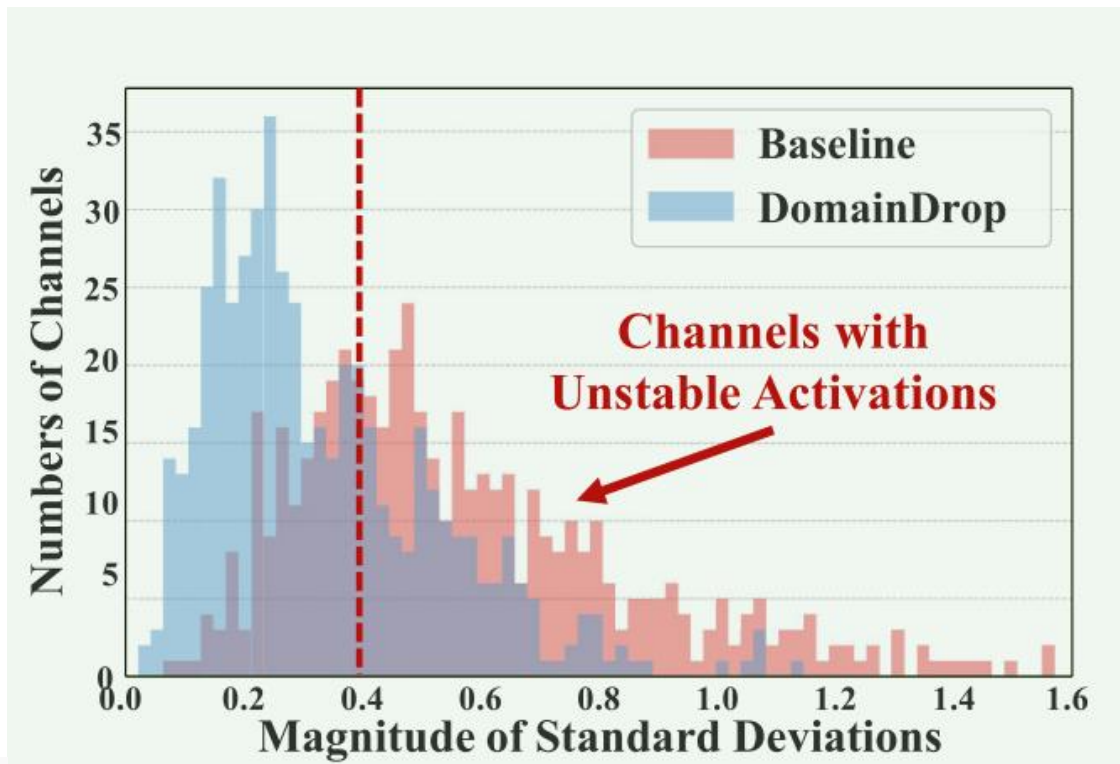
Domain adaptation uses both labeled source domain data and unlabeled target domain data for training, whereas domain generalization only employs labeled source domain data. In domain adaptation, target domain data is partially accessible during training, while in domain generalization, target domain data is completely unavailable.

# Background

Many models perform excellently on the source domain but experience a significant performance drop on the test domain. The core problem is that the model captures excessive domain-specific features rather than domain-invariant ones. Through experiments, we found unstable activations across different domains at the feature level.

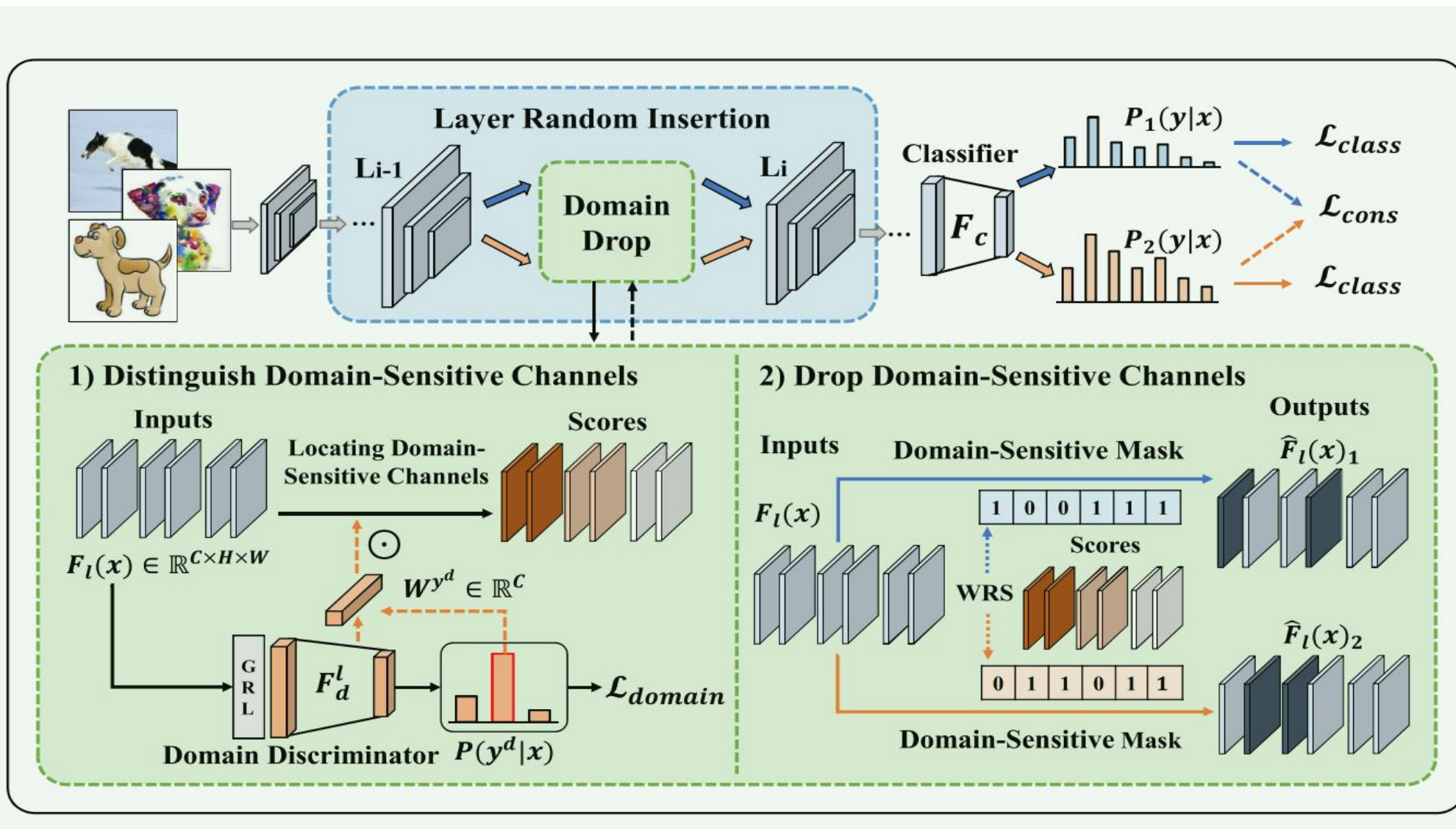
## Problem Statement

How to identify and suppress these domain-sensitive channels that lead to degraded generalization performance.



This paper proposes a simple yet effective framework, DomainDrop, which can explicitly silence domain-sensitive channels, i.e., unstable channels.

This paper aims to remove domain-specific features from representations during the forward propagation process. In addition, a domain discriminator is introduced to assign a specific dropout rate to each channel based on its effectiveness in the domain recognition task.



### Suppressing Domain-Sensitive Channels

To clearly understand how the model removes domain-specific features during training, domain discriminators are inserted into multiple intermediate layers to locate domain-sensitive channels. Each intermediate branch consists of a global average pooling (GAP) layer and a fully connected (FC) layer.

$$\mathcal{L}_{domain}^l = -\frac{1}{K} \sum_{k=1}^K \left[ \frac{1}{n_k} \sum_{i=1}^{n_k} \sum_{j=1}^K \mathbb{1}_{[j=k]} \log F_d(F_l(x_i^k)) \right]. \quad (1)$$

### Distinguish Domain-Sensitive Channels

To determine which channels contain domain-specific information, a domain discriminator is used as an indicator. Specifically, we assume that channels contributing the most to domain prediction contain domain-specific information, and quantify the correlation between them. We define the score of the  $j$ -th channel of the feature map as:

$$s_j = W_j^{y^d} \cdot \text{GAP}(F_l(x_i^k))_j, \quad (2)$$

## Dropping Domain-Sensitive Channels

To explicitly reduce domain-specific information in the feature map, we propose selecting and dropping the most domain-sensitive channels during training. Specifically, using the score  $S_j$ , we first compute the probability  $P_j$  of dropping the  $j$ -th channel:

$$p_j = s_j / \sum_{c=1}^C s_c. \quad (3)$$

## Layer-wise Training Scheme

Through experiments, we observe that domain-specific patterns commonly exist across network layers. Therefore, we adopt a layer-wise training strategy: we randomly select an intermediate layer in the network and apply DomainDrop to the feature map at each iteration. This scheme effectively reduces the channel sensitivity to domain shift across multiple network layers.

## Enhancing Domain-Invariant Channels

During the training stage, DomainDrop is used to remove domain-sensitive channels probabilistically, which is equivalent to applying Gaussian noise to domain-specific features. To alleviate the output inconsistency caused by such perturbations introduced by DomainDrop, a dual consistency loss is proposed to strengthen domain-invariant channels.

$$\begin{aligned} \mathcal{L}_{cons} = & \frac{1}{2} (\text{KL}[\sigma(\hat{F}(x)_1/T) || \sigma(\hat{F}(x)_2/T)] \\ & + \text{KL}[\sigma(\hat{F}(x)_2/T) || \sigma(\hat{F}(x)_1/T)]), \end{aligned} \quad (5)$$

## dataset

Experiments are conducted on four traditional domain generalization (DG) datasets:

**PACS** consists of images from four domains: photo, art painting, cartoon, and sketch, covering 7 object categories and 9,991 images.

**Office-Home** contains 65 categories across four domains with approximately 15,500 images.

**VLCS** includes five categories from four domains, and the dataset is split into training and validation sets with a 7:3 ratio.

**DomainNet** comprises 345 categories from six domains.

Subsequently, 80% of the source domain data is used for training and 20% for validation.

## Experiments Setting up

The batch size is set to **128**, weight decay to **0.0005**, and the initial learning rate to **0.002**. A decay rate of **0.1%** is applied to the learning rate for **80%** of the total epochs (with the total number of epochs set to **50**).

Additionally, an alternative configuration uses an initial learning rate of **5e-5** and a batch size of **64**. For DomainDrop, the dropout rate is set to **0.33**; the weight of the Gradient Reversal Layer (GRL) is set to **0.25**; and the temperature parameter  $T$  for the dual consistency loss is set to **1.5**.

Methods	Art	Cartoon	Photo	Sketch	Avg.
ResNet-18					
DeepAll [72] (AAAI'20)	80.31	76.65	95.38	71.67	81.00
I <sup>2</sup> -Drop [46] (ICML'20)	80.27	76.54	96.11	76.38	82.33
DMG [5] (ECCV'20)	80.38	76.90	93.35	75.21	81.46
RSC [23] (ECCV'20)	83.43	80.31	95.99	80.85	85.15
NAS-OoD [1] (ICCV'21)	83.74	79.69	96.23	77.27	84.23
MixStyle [73] (ICLR'21)	84.10	78.80	96.10	75.90	83.70
LDSDG [57] (ICCV'21)	81.44	79.56	95.51	80.58	84.27
PLACE [19] (arXiv'21)	82.60	78.33	95.65	81.47	84.51
FACT [61] (CVPR'21)	<b>85.37</b>	78.38	95.15	79.15	84.51
StableNet [68] (CVPR'21)	81.74	79.91	96.53	80.50	84.69
EFDMix [69] (CVPR'22)	83.90	79.40	96.80	75.00	83.90
StyleNeophile [24] (CVPR'22)	84.41	79.25	94.93	83.27	85.47
COMEN [6] (CVPR'22)	82.60	<b>81.00</b>	94.60	84.50	85.70
I <sup>2</sup> -ADR [34] (ECCV'22)	82.90	80.80	95.00	83.50	85.60
CDG [12] (ICV'22)	83.50	80.10	95.60	83.80	85.80
ALOFT [20] (CVPR'23)	84.81	79.05	96.11	80.55	85.13
<b>DomainDrop (Ours)</b>	<b>84.47</b> ± 0.77	<b>80.50</b> ± 0.56	<b>96.83</b> ± 0.21	<b>84.83</b> ± 0.67	<b>86.66</b>
ResNet-50					
DeepAll [72] (AAAI'20)	81.31	78.54	94.97	69.76	81.15
RSC [23] (ECCV'20)	87.89	82.16	97.92	83.85	87.83
FACT [61] (CVPR'21)	89.63	81.77	96.75	84.46	88.15
PLACE [19] (arXiv'21)	87.55	83.11	97.19	83.48	87.83
COPA [59] (ICCV'21)	83.30	79.80	94.60	82.50	85.10
SWAD [3] (NeurIPS'21)	89.30	83.40	97.30	82.50	88.10
EFDMix [69] (CVPR'22)	<b>90.60</b>	82.50	98.10	76.40	86.90
StyleNeophile [24] (CVPR'22)	90.35	84.20	96.73	85.18	89.11
I <sup>2</sup> -ADR [34] (ECCV'22)	88.50	83.20	95.20	85.80	88.20
PTE [35] (ECCV'22)	87.90	78.40	<b>98.20</b>	75.70	85.10
CDG [12] (ICV'22)	88.90	83.50	97.60	84.90	88.70
ALOFT [20] (CVPR'23)	89.26	83.11	97.96	84.04	88.59
<b>DomainDrop (Ours)</b>	<b>89.82</b> ± 0.44	<b>84.22</b> ± 0.37	<b>98.02</b> ± 0.24	<b>85.98</b> ± 1.14	<b>89.51</b>

Methods	Caltech	LabelMe	Pascal	Sun	Avg.
DeepAll [72] (AAAI'20)	96.98	62.00	73.83	68.66	75.37
JiGen [2] (CVPR'19)	96.17	62.06	70.93	71.40	75.14
MMLD [33] (AAAI'20)	97.01	62.20	73.01	72.49	76.18
RSC [23] (ECCV'20)	95.83	63.74	71.86	72.12	75.89
StableNet [68] (CVPR'21)	96.67	<b>65.36</b>	73.59	<b>74.97</b>	77.65
<b>DomainDrop (Ours)</b>	<b>98.94</b> ± 0.19	63.97 ± 1.33	<b>76.36</b> ± 0.93	73.74 ± 1.17	<b>78.25</b>

Methods	Art	Clipart	Product	Real	Avg.
ResNet-18					
DeepAll [72] (AAAI'20)	56.98	50.14	72.85	74.39	63.59
RSC [23] (ECCV'20)	57.70	48.58	72.59	74.17	63.26
MixStyle [73] (ICLR'21)	58.70	53.40	74.20	75.90	65.50
SagNet [37] (CVPR'21)	60.20	45.38	70.42	73.38	62.34
COPA [59] (ICCV'21)	59.40	55.10	74.80	75.00	66.10
FACT [61] (CVPR'21)	<b>60.34</b>	54.85	74.48	76.55	66.56
StyleNeophile [24] (CVPR'22)	59.55	55.01	73.57	75.52	65.89
CDG [12] (ICV'22)	59.20	54.30	<b>74.90</b>	75.70	66.00
<b>DomainDrop (Ours)</b>	<b>59.62</b> ± 0.40	<b>55.60</b> ± 0.31	<b>74.50</b> ± 0.53	<b>76.64</b> ± 0.35	<b>66.59</b>
ResNet-50					
DeepAll [72] (AAAI'20)	61.30	52.40	75.80	76.60	66.50
RSC [23] (ECCV'20)	57.70	51.40	74.80	75.10	65.50
SelfReg [26] (ICCV'21)	63.60	53.10	76.90	78.10	67.90
SagNet [37] (CVPR'21)	63.40	54.80	75.80	78.30	68.10
SWAD [3] (NeurIPS'21)	66.10	57.70	78.40	80.20	70.60
Fishr [44] (ICML'22)	63.40	54.20	76.40	78.50	68.20
PTE [35] (ECCV'22)	66.30	55.80	78.20	<b>80.40</b>	70.10
<b>DomainDrop (Ours)</b>	<b>67.33</b> ± 0.45	<b>60.39</b> ± 0.48	<b>79.05</b> ± 0.29	80.22 ± 0.22	<b>71.75</b>

Methods	Clipart	Infograph	Painting	Quickdraw	Real	Sketch	Avg.
DeepAll [72] (AAAI'20)	62.20	19.90	45.50	13.80	57.50	44.40	40.50
RSC [23] (ECCV'20)	55.00	18.30	44.40	12.20	55.70	47.80	38.90
SagNet [37] (CVPR'21)	57.70	19.00	45.30	12.70	58.10	48.80	40.30
SelfReg [26] (ICCV'21)	60.70	21.60	49.40	12.70	60.70	51.70	42.80
I <sup>2</sup> -ADR [34] (ECCV'22)	<b>64.40</b>	20.20	49.20	<b>15.00</b>	61.60	53.30	44.00
PTE [35] (ECCV'22)	62.40	21.00	50.50	13.80	<b>64.60</b>	52.40	44.10
<b>DomainDrop (Ours)</b>	62.93 ± 0.25	<b>21.63</b> ± 0.07	<b>50.67</b> ± 0.15	14.83 ± 0.30	62.70 ± 0.08	<b>53.46</b> ± 0.59	<b>44.37</b>

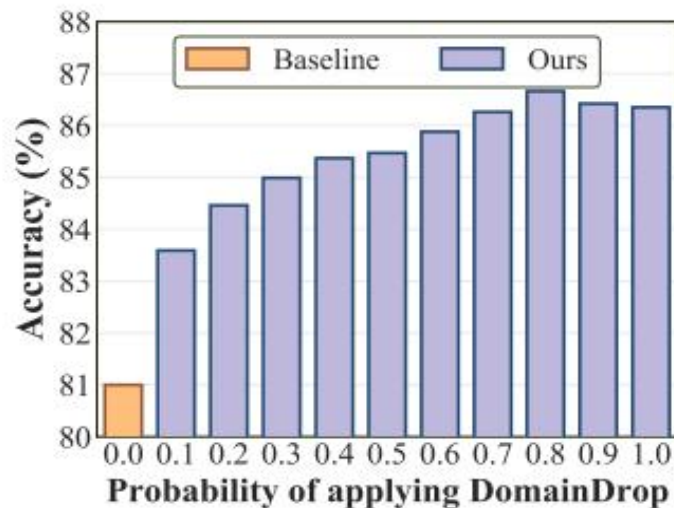
## Ablation study:

- The first setting applies DomainDrop to all layers, activated with a probability of 0.5 per layer, and comparisons are conducted on the PACS and Office-Home datasets.
- The second setting applies DomainDrop only to a single fixed layer, rather than to multiple random layers.
- The third setting removes the dual consistency loss that is used to enhance domain-invariant features.

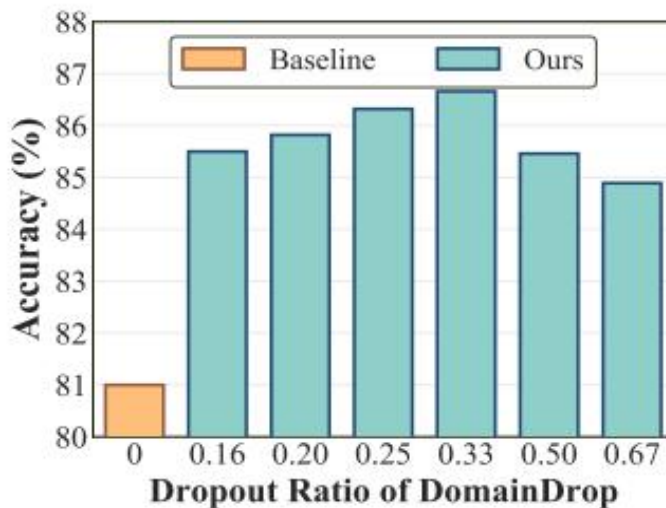
PACS								
Methods	DD	LT	CL	Art	Cartoon	Photo	Sketch	Avg.
Baseline	-	-	-	80.31 $\pm$ 1.54	76.65 $\pm$ 0.48	95.38 $\pm$ 0.12	71.67 $\pm$ 1.49	81.00
Variant 1	✓	-	-	82.37 $\pm$ 0.78	79.27 $\pm$ 0.26	95.45 $\pm$ 0.27	83.15 $\pm$ 0.73	85.06
Variant 2	✓	✓	-	83.84 $\pm$ 0.12	79.69 $\pm$ 0.29	96.47 $\pm$ 0.36	83.74 $\pm$ 0.39	85.94
Variant 3	✓	-	✓	83.32 $\pm$ 0.76	80.09 $\pm$ 0.91	95.85 $\pm$ 0.33	83.71 $\pm$ 0.61	85.74
DomainDrop	✓	✓	✓	<b>84.47</b> $\pm$ 0.77	<b>80.50</b> $\pm$ 0.56	<b>96.83</b> $\pm$ 0.21	<b>84.83</b> $\pm$ 0.67	<b>86.66</b>

OfficeHome								
Methods	DD	LT	CL	Art	Clipart	Product	Real	Avg.
Baseline	-	-	-	56.98 $\pm$ 0.47	50.14 $\pm$ 0.88	72.85 $\pm$ 0.35	74.39 $\pm$ 0.32	63.59
Variant 1	✓	-	-	57.49 $\pm$ 0.37	53.69 $\pm$ 0.58	73.82 $\pm$ 0.14	76.18 $\pm$ 0.20	65.30
Variant 2	✓	✓	-	59.46 $\pm$ 0.28	53.86 $\pm$ 0.57	74.23 $\pm$ 0.29	76.31 $\pm$ 0.24	65.96
Variant 3	✓	-	✓	58.17 $\pm$ 0.29	54.96 $\pm$ 0.40	73.95 $\pm$ 0.35	76.42 $\pm$ 0.49	65.88
DomainDrop	✓	✓	✓	<b>59.62</b> $\pm$ 0.40	<b>55.60</b> $\pm$ 0.31	<b>74.50</b> $\pm$ 0.53	<b>76.64</b> $\pm$ 0.35	<b>66.59</b>



(a) Effect of dropout probability.



(b) Effects of dropout ratio.

Figure 4. Effects of hyper-parameters including the probability of applying DomainDrop and the dropout ratio in DomainDrop. The experiments are conducted on PACS with ResNet-18 backbone.

***Thanks***