# Direct Distillation between Different Domains

Jialiang Tang[1,2,3], Shuo Chen[4*], Gang Niu[4], Hongyuan Zhu[5,6], Joey Tianyi Zhou[6,7,8], Chen Gong[1,2,3*], and Masashi Sugiyama[4,9]

[1] School of Computer Science and Engineering, Nanjing University of Science and Technology, China
[2] Key Laboratory of Intelligent Perception and Systems for High-Dimensional Information of Ministry of Education, China
[3] Jiangsu Key Laboratory of Image and Video Understanding for Social Security, China
[4] Center for Advanced Intelligence Project, RIKEN, Japan
[5] Institute for Infocomm Research ($I^2R$), A*STAR, Singapore
[6] Centre for Frontier AI Research (CFAR), A*STAR, Singapore
[7] Institute of High Performance Computing (IHPC), A*STAR, Singapore
[8] The Centre for Advanced Technologies in Online Safety, Singapore
[9] The Graduate School of Frontier Sciences, The University of Tokyo, Japan
`shuo.chen.ya@riken.jp` `chen.gong@njust.edu.cn`

**Abstract.** Knowledge Distillation (KD) aims to learn a compact student network using knowledge from a large pre-trained teacher network, where both networks are trained on data from the same distribution. However, in practical applications, the student network may be required to perform in a new scenario (*i.e.*, the target domain), which usually exhibits significant differences from the known scenario of the teacher network (*i.e.*, the source domain). The traditional domain adaptation techniques can be integrated with KD in a two-stage process to bridge the domain gap, but the ultimate reliability of two-stage approaches tends to be limited due to the high computational consumption and the additional errors accumulated from both stages. To solve this problem, we propose a new one-stage method dubbed "Direct Distillation between Different Domains" (4Ds). We first design a learnable adapter based on the Fourier transform to separate the domain-invariant knowledge from the domain-specific knowledge. Then, we build a fusion-activation mechanism to transfer the valuable domain-invariant knowledge to the student network, while simultaneously encouraging the adapter within the teacher network to learn the domain-specific knowledge of the target data. As a result, the teacher network can effectively transfer categorical knowledge that aligns with the target domain of the student network. Intensive experiments on various benchmark datasets demonstrate that our proposed 4Ds method successfully produces reliable student networks and outperforms state-of-the-art approaches. Code is available at https://github.com/tangjialiang97/4Ds.

**Keywords:** Model compression · Knowledge distillation · Domain adaptation

---

*Corresponding authors
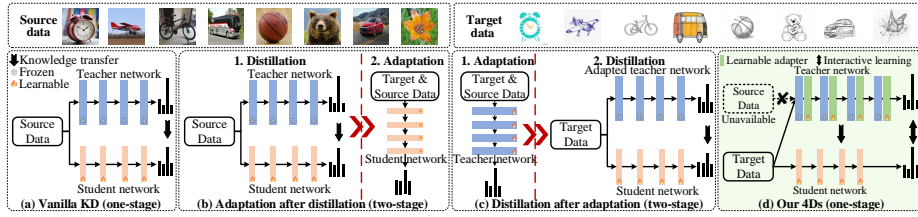
**Fig. 1:** Comparison between existing Knowledge Distillation (KD) methods and our Direct Distillation between Different Domains (4Ds). (a) Vanilla KD transfers knowledge from a fixed pre-trained teacher network to the student network where the data is identically distributed. "Adaptation after distillation" in (b) first learns a student network on the source data via KD and then generalizes the student network on the target data via Domain Adaptation (DA). "Distillation after adaptation" in (c) first adapts the source-data-trained teacher network to the target data via DA and then utilizes the adapted teacher network to guide the student network training on the target data via KD. (d) Our 4Ds trains the student network on the target data using the teacher network (with the learnable adapters) trained on the source data.

## 1   Introduction

In recent years, Deep Neural Networks (DNNs) have achieved great success in many computer vision areas, such as image recognition [21,45] and object detection [27,58]. However, these powerful DNNs are usually infeasible for resource-limited devices (*e.g.*, smartphones or automobiles) due to their enormous computational and storage demands. To solve this problem, various model compression approaches have been proposed, mainly including network pruning [50], model quantization [8], and Knowledge Distillation (KD) [18,22]. Among these methods, KD has gained popularity for enhancing the performance of small student networks by transferring knowledge from large pre-trained teacher networks.

However, in practice, it is quite often that the acquired small student network should be capable of adapting to a new environment that differs from the one in which the original teacher network was trained. For example, in autonomous driving, one popular way is to train a well-performed yet large perception model in a simulation environment first, and eventually deploy it in real-world autonomous vehicles with limited computational and memory resources [24]. Nevertheless, due to the domain gap between the simulation environment (*i.e.*, the source domain) and real-world driving scenarios (*i.e.*, the target domain), existing KD methods cannot always perform well. These methods usually assume that the underlying data distributions for both the teacher network and student network are the same, and the direct knowledge transfer from the teacher network (well-trained in the source domain) to the student network (in the target domain) sometimes leads to poor performance on the downstream tasks.

Intuitively, as shown in subfigures (b) and (c) of Fig. 1, there are two possible ways to mitigate such domain discrepancy in KD, namely: 1) First learning the student network in the source domain via KD, and then adapting it to the

target domain (*i.e.*, "adaptation after distillation"); and 2) First adapting the teacher network to the target domain, and then training the student network via KD (*i.e.*, "distillation after adaptation") [15, 26]. However, both aforementioned approaches have clear shortcomings in terms of high computational complexity, potential error accumulation, and dependence on the source data. First, the two-stage approaches repeatedly iterate the teacher or student network, significantly increasing the computational costs. Second, neither distillation nor adaptation is fully reliable, so the two-stage framework may inevitably accumulate additional errors that further impair the model performance. Third, adapting the teacher or student network to an unfamiliar domain usually still requires the source data, which may be inaccessible due to privacy concerns. Therefore, if we want to effectively distill and adapt the knowledge of pre-trained models to our practical cases-of-interests with limited computational and storage resources, it is desirable to overcome the above disadvantages of two-stage methods. This motivates us to develop a one-stage end-to-end method that directly distills the critical knowledge from the source domain to the target domain without using the source data.

In this paper, we propose a novel method called "**D**irect **D**istillation between **D**ifferent **D**omains" (4Ds), which straightforwardly trains a student network in the target domain by using only a well-trained teacher network in another different source domain. Specifically, as the teacher network contains both ***domain-invariant*** knowledge (*e.g.*, semantics) and ***domain-specific*** knowledge (*e.g.*, color and style), we design a novel Fourier transform based adapter and integrate such a learnable adapter into the teacher network to ***decouple*** these two types of knowledge. After that, we retain the useful domain-invariant knowledge and transfer it to the student network via a fusion-activation mechanism. For the domain-specific information learned from the source domain, it is difficult to effectively transfer to the student network due to the discrepancies between the source and target domains. Therefore, we encourage the learnable adapter imposed on the teacher network to grasp new domain-specific information from the target domain, thereby improving the teacher's performance as a reliable supervisor for the student network. Consequently, the teacher network can capture reliable category relations compatible with the target domain, and successfully transfer them to the student network. The introduced adapter constitutes only 2% of the teacher network parameters, making our method considerably more efficient than the two-stage approaches that repeatedly train the entire teacher or student network on source and target data (analyzed in the **Supplementary Materials**). Furthermore, we directly extract the domain-invariant knowledge contained in the teacher network without relying on the source data, therefore effectively avoiding potential data privacy issues.

The contributions of this paper are summarized as:

1. We investigate a novel learning scenario of model compression between different domains, and it motivates us to propose a one-stage KD framework termed 4Ds, which directly trains a compact student network in a new target domain using a pre-trained teacher network from the source domain.

2. We develop a new Fourier transform based adapter to decouple the domain-invariant knowledge and domain-specific knowledge contained in the teacher network, and then design a fusion-activation mechanism to transfer such valuable domain-invariant knowledge to the student network.
3. Intensive experiments on multiple benchmarks demonstrate that our proposed 4Ds can outperform the two-stage methods and state-of-the-art knowledge transfer based approaches.

## 2   Related Works

### 2.1   Knowledge Distillation

Conventional KD promotes a lightweight student network to acquire knowledge from a cumbersome teacher network, which assumes that the data for the teacher network and student network follows the same distribution. According to the different knowledge being transmitted, existing KD approaches can be categorized into logit-based [22,33], feature-based [2,32], and relation-based [40,47] methods. Some feature-based methods [3,7] also perform multi-block feature fusion, they transfer knowledge from multiple teacher layers to a single student layer, while our 4Ds transfers global knowledge between all blocks of teacher and student.

Departing from the conventional KD, some recently proposed data-free KD methods assume that the original training data of the teacher network is usually unavailable due to privacy concerns, so they train the student networks by model-generated data [6,11] or web-collected real data [5,13,46]. However, these methods still fail to train a reliable student network in a new domain that significantly differs from the training domain of the teacher network.

### 2.2   Domain Adaptation

Domain Adaptation (DA) [31] leverages abundant labeled data or well-trained models from the source domain to enhance the model performance in the different target domains. In general, existing DA approaches bridge the distribution discrepancies between the source and target domains via instance reweighting [36], adversarial training [48], or knowledge transfer [26]. Among them, knowledge transfer methods are rather simple and have been successfully applied to many fields, such as semantic segmentation [15], federated learning [17], and few-shot learning [26]. To improve the model generalize ability in the target domain, these methods transfer and align a variety of knowledge between models in the source domain and target domain, such as soft targets [34], gradient information [57], sharpened predictions [26], and instance relations [12,15].

Note that although our approach shares some similar points with the DA methods that utilize knowledge transfer [15,23,26,57], there are fundamental differences between our 4Ds and the existing methods: 1) (*Research goal*) DA only adapts the model trained in the source domain to the target domain, whereas our approach further achieves model compression by training a small student

network in the target domain; 2) (*Employed strategies*) DA methods reduce the distribution discrepancies by aligning their distributions, while our approach improves the performance of the student network by extracting and transferring beneficial knowledge from the teacher network. To the best of our knowledge, we are the first to implement one-stage model compression under the learning scenario with different domains where the source data is entirely unavailable.

### 2.3   Fourier Feature

Lots of early works [28, 38, 43] reveal that the decoupled phase and amplitude of the Fourier feature can effectively capture the domain-invariant semantic and domain-specific statistic information, respectively. Recently, Xu et al. [51] encourage the model to produce consistent outputs across examples with varying amplitudes, thereby enhancing its resilience to domain shift. Lu et al. [37] directly train the model on the phase to learn domain-invariant information. Lee et al. [29] preserve the domain-invariant phase of the source-domain-trained model during the DA process, effectively generalizing the model to the target domain. In our case, we update the domain-specific knowledge of the teacher network trained in the source domain while transferring its domain-invariant knowledge to improve the performance of the small student network on the target domain.

## 3   Approach

In our problem setting, we aim to train a streamlined student network $\mathcal{N}_S$ in the new target domain using a pre-trained teacher network $\mathcal{N}_T$ from the source domain. Here, $\mathcal{N}_T$ is pre-trained on the source dataset $\mathcal{D}_s = \{(\mathbf{x}_i^s, y_i^s)\}_{i=1}^{|\mathcal{D}_s|}$ where $\mathbf{x}_i^s \in \mathcal{X}_s$, $y_i^s \in \mathcal{Y}_s$, and $\mathcal{N}_S$ is trained on the target dataset $\mathcal{D}_t = \{(\mathbf{x}_i^t, y_i^t)\}_{i=1}^{|\mathcal{D}_t|}$ where $\mathbf{x}_i^t \in \mathcal{X}_t$, $y_i^t \in \mathcal{Y}_t$, and "$|\cdot|$" symbolizes the data cardinality of the corresponding dataset. Although $\mathcal{D}_s$ and $\mathcal{D}_t$ share the same label space, their sample spaces differ from each other, *i.e.*, the label spaces $\mathcal{Y}_s = \mathcal{Y}_t$ but the sample spaces $\mathcal{X}_s \neq \mathcal{X}_t$. Ideally, $\mathcal{N}_S$ should deliver satisfactory performance on $\mathcal{D}_t$ by mimicking the knowledge acquired by $\mathcal{N}_T$ that trained on $\mathcal{D}_s$.

Specifically, $\mathcal{N}_T$ is initially pre-trained on $\mathcal{D}_s$, which embodies two distinct types of knowledge. The first one is **domain-invariant** knowledge, which pertains to the semantics of images across both $\mathcal{D}_s$ and $\mathcal{D}_t$. Such knowledge is shared between both two domains and is still beneficial for learning $\mathcal{N}_S$ on $\mathcal{D}_t$. The second one is **domain-specific** knowledge, which refers to the inherent color and brightness of images in $\mathcal{D}_s$. These attributes significantly diverge from those present in $\mathcal{D}_t$ and are harmful to the training of $\mathcal{N}_S$ on $\mathcal{D}_t$. Therefore, it is necessary to **decouple** the domain-invariant and domain-specific knowledge embedded within the teacher network.

To overcome the challenges of KD under domain discrepancies with the above issues, we propose a one-stage end-to-end method called 4Ds. As shown in Fig. 2, our 4Ds comprises three critical and closely collaborative elements, including: 1) Knowledge adapter incorporating Fourier transform, which decomposes the
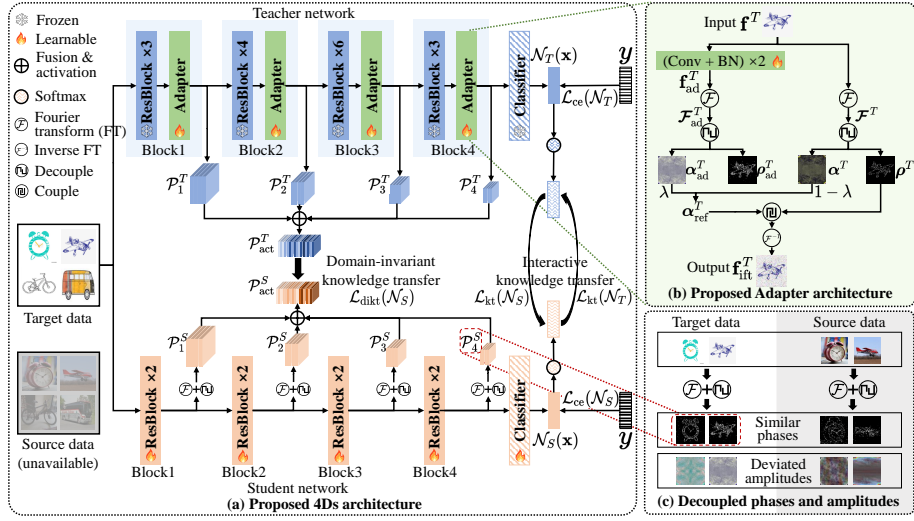
**Fig. 2:** The diagram of our proposed 4Ds. (a) During training, both the teacher network $\mathcal{N}_T$ (ResNet34) and student network $\mathcal{N}_S$ (ResNet18) interactively learn from the target data. $\mathcal{N}_T$ is encouraged to produce accurate and useful category relations for $\mathcal{N}_S$ by updating its imposed adapters. Meanwhile, $\mathcal{N}_S$ is promoted to learn the valuable domain-invariant features as well as the reliable category relations from $\mathcal{N}_T$. (b) In our designed adapter, the input feature $\mathbf{f}^T$ is first fed into two learnable convolution layers to grasp the target-domain-specific knowledge. Subsequently, the original domain-specific knowledge is refurbished by mixing the amplitudes $\boldsymbol{\alpha}^T$ and $\boldsymbol{\alpha}_{\text{ad}}^T$, which are decoupled from the original $\mathbf{f}^T$ and adapted $\mathbf{f}_{\text{ad}}^T$, respectively. Finally, the output feature $\mathbf{f}_{\text{ift}}^T$ is recovered from the remained phase $\boldsymbol{\rho}^T$ from $\mathbf{f}^T$ and refurbished amplitude $\boldsymbol{\alpha}_{\text{ref}}^T$. (c) The input source images and target images are decoupled into phases and amplitudes by the Fourier transform and decoupling operations.

spatial feature into domain-invariant phase and domain-specific amplitude via the Fourier transform; 2) Domain-invariant feature transfer, which distills the domain-invariant features from the teacher network to the student network via our designed fusion-activation mechanism; 3) Teacher-student interactive training, which alternatively trains the student network and learnable adapters in teacher network on the target data. Our 4Ds does not use any source data during the training of the student network. Therefore, for convenience in description, we symbolize the target data as $\mathcal{D} = \{(\mathbf{x}_i, y_i)\}_{i=1}^{|\mathcal{D}|}$ in the following discussions. Meanwhile, the notations with superscripts "$T$" and "$S$" denote that they are related to $\mathcal{N}_T$ and $\mathcal{N}_S$, respectively.

### 3.1   Knowledge Adapter Incorporating Fourier Transform

For the Fourier transform [9,28,38], as shown in Fig. 2 (c), the phase usually encodes high-level semantics, *i.e.*, the structures and relative positions of the objects in the frequency spectrum, which are consistent across domains. Conversely,

the amplitude effectively captures low-level statistics, *e.g.*, color and brightness, which are prone to domain gaps. Recent works [4, 29, 51] leverage the Fourier transform to segregate the intermediate spatial feature of DNN into phase and amplitude and independently address the phase or amplitude to enhance model generalizability. In this paper, we design an adapter incorporating the Fourier transform to decouple the phase and amplitude within the teacher network, rendering them domain-invariant and domain-specific features, respectively, and properly utilize both of them to train a reliable student network.

As depicted in Fig. 2, the teacher network comprises a total of $n$ blocks ($n$=4 for ResNet [21]), each following an efficient adapter consisting of only two $1\times1$ convolutional layers. During training, all original components of the teacher network are frozen and only the integrated adapters are updated, which ensures the preservation of the useful domain-invariant knowledge in the original teacher network. Conversely, the domain-specific knowledge that differs from the target domain is updated in the target domain by these learnable adapters.

For an input example $\mathbf{x} \in \mathcal{D}$ given to $\mathcal{N}_T$, the spatial feature $\mathbf{f}^T \in \mathbb{R}^{H \times W}$ ($H$ and $W$ denote the height and width of the feature map, respectively) produced by a specific block in $\mathcal{N}_T$ is first adapted by the learnable convolutional layers in its adjacency adapter as follows:

$$\mathbf{f}_{\mathrm{ad}}^T = \mathrm{BN}(\mathbf{W}_{\mathrm{ad}}^2 \max(0, \mathrm{BN}(\mathbf{W}_{\mathrm{ad}}^1 \mathbf{f}^T))). \tag{1}$$

Here, $\mathbf{W}_{\mathrm{ad}}^1 \in \mathbb{R}^{\frac{C}{r_{\mathrm{ad}}} \times C}$ and $\mathbf{W}_{\mathrm{ad}}^2 \in \mathbb{R}^{C \times \frac{C}{r_{\mathrm{ad}}}}$ are the weights for convolutional layers; the channel dimension $C$ of input feature is omitted to simplify the description; $r_{\mathrm{ad}} = 4$ is a scaling parameter used to reduce computationals; BN denotes the batch normalization. Then, the adapted feature $\mathbf{f}_{\mathrm{ad}}^T \in \mathbb{R}^{H \times W}$ is converted to frequency feature $\mathcal{F}_{\mathrm{ad}}^T = \mathrm{FT}(\mathbf{f}_{\mathrm{ad}}^T)$ by **F**ourier **t**ransform[*]. Subsequently, the adapted amplitude $\boldsymbol{\alpha}_{\mathrm{ad}}^T$ contains information specific to the target domain can be separated from $\mathcal{F}_{\mathrm{ad}}^T \in \mathbb{R}^{H \times W}$ leveraging the decoupling operation, namely:

$$(\boldsymbol{\alpha}_{\mathrm{ad}}^T, \boldsymbol{\rho}_{\mathrm{ad}}^T) = \mathrm{Decouple}(\mathcal{F}_{\mathrm{ad}}^T), \tag{2}$$

where

$$\begin{cases} \boldsymbol{\alpha}_{\mathrm{ad}}^T(u, v) = \sqrt{(\mathcal{F}_{\mathrm{ad\_real}}^T(u, v))^2 + (\mathcal{F}_{\mathrm{ad\_img}}^T(u, v))^2}, \\ \boldsymbol{\rho}_{\mathrm{ad}}^T(u, v) = \arctan \dfrac{\mathcal{F}_{\mathrm{ad\_img}}^T(u, v)}{\mathcal{F}_{\mathrm{ad\_real}}^T(u, v)}. \end{cases} \tag{3}$$

Here, $\boldsymbol{\alpha}_{\mathrm{ad}}^T \in \mathbb{R}^{H \times W}$ and $\boldsymbol{\rho}_{\mathrm{ad}}^T \in \mathbb{R}^{H \times W}$. Meanwhile, the original feature $\mathbf{f}^T$ is also decoupled to $\boldsymbol{\alpha}^T$ and $\boldsymbol{\rho}^T$ undergoes the same process.

Recent works [14, 29] have suggested that completely eliminating the source-domain-specific information in DNNs may harm their performance in the target domain. In light of this, we refurbish the domain-specific feature in the teacher

---

[*]Each Fourier coefficient $\mathcal{F}_{\mathrm{ad}}^T(u, v) = \frac{1}{HW} \sum_{h=1}^{H} \sum_{w=1}^{W} \mathbf{f}_{\mathrm{ad}}^T(h, w) e^{-\mathrm{i}2\pi \left(\frac{uh}{H} + \frac{vw}{W}\right)} = \mathcal{F}_{\mathrm{ad\_real}}^T(u, v) + \mathrm{i}\mathcal{F}_{\mathrm{ad\_img}}^T(u, v)$, where i is the imaginary unit, $\mathcal{F}_{\mathrm{ad\_real}}^T$ and $\mathcal{F}_{\mathrm{ad\_img}}^T$ are the real and imaginary parts, respectively.

network by assembling both $\boldsymbol{\alpha}_{\text{ad}}^T$ captured from the target domain and original $\boldsymbol{\alpha}^T$ inherited by the teacher network using a learnable parameter $\lambda \in (0, 1)$:

$$\boldsymbol{\alpha}_{\text{ref}}^T = \lambda\boldsymbol{\alpha}_{\text{ad}}^T + (1 - \lambda)\boldsymbol{\alpha}^T, \tag{4}$$

where the refurbished amplitude $\boldsymbol{\alpha}_{\text{ref}}^T \in \mathbb{R}^{H \times W}$, and $\lambda$ is analyzed in the **Supplementary Materials**. Then, the frequency feature $\boldsymbol{\mathcal{F}}_{\text{ref}}^T$ is coupled as:

$$\boldsymbol{\mathcal{F}}_{\text{ref}}^T = \text{Couple}(\boldsymbol{\alpha}_{\text{ref}}^T, \boldsymbol{\rho}^T) = \boldsymbol{\alpha}_{\text{ref}}^T \cos(\boldsymbol{\rho}^T) + \boldsymbol{\alpha}_{\text{ref}}^T \sin(\boldsymbol{\rho}^T), \tag{5}$$

where $\boldsymbol{\mathcal{F}}_{\text{ref}}^T \in \mathbb{R}^{H \times W}$ contains both valuable domain-invariant knowledge in the immutable $\boldsymbol{\rho}^T$ and domain-specific knowledge adaptable to the target domain in the refurbished $\boldsymbol{\alpha}_{\text{ref}}^T$. Finally, the frequency feature $\boldsymbol{\mathcal{F}}_{\text{ref}}^T$ is recovered as spatial feature $\mathbf{f}_{\text{ift}}^T = \text{IFT}(\boldsymbol{\mathcal{F}}_{\text{ref}}^T)$ via the **i**nverse **F**ourier **t**ransform[*]. Afterward, $\mathbf{f}_{\text{ift}}^T \in \mathbb{R}^{H \times W}$ is input into the subsequent blocks of the teacher network to produce accurate predictions.

During the 4Ds training process, the features produced by each teacher's block are separated into domain-invariant and domain-specific features by their adjacent adapter. As a result, we can transfer useful domain-invariant features to improve the performance of the student network (as detailed in Section 3.2) and also promote a teacher network easily accessible by the student network to facilitate knowledge transfer (as explained in Section 3.3). The structure analysis and pseudo-code for the adapter are provided in the **Supplementary Materials**.

### 3.2   Domain-Invariant Feature Transfer

The domain-invariant features decoupled from the teacher network contain useful information similar to the target data, which benefits DNN training on the target domain [29,51]. Therefore, we develop the fusion-activation strategy to facilitate the transfer of domain-invariant features from the teacher network to the student network. Initially, the fusion operation integrates the features extracted from various blocks. Subsequently, the activation operation maps these fused features as attention weights to control the feature transfer.

In the fusion operation, an input example $\mathbf{x} \in \mathcal{D}$ is passed into the teacher network to extract the domain-invariant features $\{\boldsymbol{\mathcal{P}}_i^T\}_{i=1}^n$ from $n$ adapters, where $\boldsymbol{\mathcal{P}}_i^T \in \mathbb{R}^{C_i \times H_i \times W_i}$ and $\boldsymbol{\mathcal{P}}_i^T[j, :, :] = \boldsymbol{\rho}_{i,j}^T \in \mathbb{R}^{H_i \times W_i}$. In DNNs, the size of features generally decreases as the network deepens, which results in a size mismatch between the features from different adapters. To fuse these features, all features in $\{\boldsymbol{\mathcal{P}}_i^T\}_{i=1}^n$ are scaled to match those in $\boldsymbol{\mathcal{P}}_n^T$ with the smallest size using average pooling and then concatenated as:

$$\boldsymbol{\mathcal{P}}_{\text{fuse}}^T = \text{Concatenate}(\text{AvgPool}(\{\boldsymbol{\mathcal{P}}_i^T\}_{i=1}^n)), \tag{6}$$

where $\boldsymbol{\mathcal{P}}_{\text{fuse}}^T \in \mathbb{R}^{M \times H_n \times W_n}$ and $M = \sum_{i=1}^n C_i$.

---

[*]Each $\mathbf{f}_{\text{ift}}^T(h, w)$ is computed as: $\mathbf{f}_{\text{ift}}^T(h, w) = \frac{1}{UV} \sum_{u=1}^U \sum_{v=1}^V \boldsymbol{\mathcal{F}}_{\text{ref}}^T(u, v) e^{\text{i}2\pi\left(\frac{uh}{U} + \frac{vw}{V}\right)}$.

After the feature fusion, the activation operation is utilized to capture the global dependencies within the fused features $\mathcal{P}_{\text{fuse}}^T$ as the channel-wise attention weights. The features in $\mathcal{P}_{\text{fuse}}^T$ are first average pooled to the channel-wise statistics with the size of $1\times1$ as follows:

$$z_m = \frac{1}{H_n \times W_n} \sum_{h=1}^{H_n} \sum_{w=1}^{W_n} \mathcal{P}_{\text{fuse}}^T[m, h, w], \tag{7}$$

where $\mathcal{P}_{\text{fuse}}^T[m, :, :] \in \mathbb{R}^{H_n \times W_n}$ represents the $m$-th feature in $\mathcal{P}_{\text{fuse}}^T$. Then, $\mathbf{Z} = [z_1, \cdots, z_m, \cdots, z_M]^\top$ ("$\top$" denotes the transpose operation) is passed through two Fully-Connected (FC) layers to capture the relation between features in $\mathcal{P}_{\text{fuse}}^T$ as the attention weights:

$$\mathbf{S} = \sigma(\mathbf{W}_{\text{act}}^2 \max(0, \mathbf{W}_{\text{act}}^1 \mathbf{Z})), \tag{8}$$

where $\sigma(\cdot)$ denotes the sigmoid activation function, $\mathbf{W}_{\text{act}}^1 \in \mathbb{R}^{\frac{M}{r_{\text{act}}} \times M}$ and $\mathbf{W}_{\text{act}}^2 \in \mathbb{R}^{M \times \frac{M}{r_{\text{act}}}}$ are the weights of the two FC layers, and $r_{\text{act}} = 4$. Based on the attention weights in $\mathbf{S} = [s_1, \cdots, s_m, \cdots, s_M]^\top$, $\forall s_m \in [0, 1]$, the features in $\mathcal{P}_{\text{fuse}}^T$ can be activated as:

$$\mathcal{P}_{\text{act}}^T = \mathcal{P}_{\text{fuse}}^T \odot \text{Expand}(\mathbf{S}, (M, H_n, W_n)), \tag{9}$$

where $\text{Expand}(\mathbf{S}, (M, H_n, W_n))$ extends the dimension of $\mathbf{S}$ to $\mathbb{R}^{M \times H_n \times W_n}$. Concurrently, the features from each block of the student network without the adapter are directly converted to frequency features by the Fourier transform and decoupled into domain-invariant features as $\{\mathcal{P}_i^S\}_{i=1}^n$. Then, $\{\mathcal{P}_i^S\}_{i=1}^n$ is fused as $\mathcal{P}_{\text{fuse}}^S$ via Eq. (6) and subsequently activated by $\mathbf{S}$:

$$\mathcal{P}_{\text{act}}^S = \mathcal{P}_{\text{fuse}}^S \odot \text{Expand}(\mathbf{S}, (M, H_n, W_n)). \tag{10}$$

Based on $\{\mathcal{P}_{\text{act},i}^S\}_{i=1}^{|\mathcal{D}|}$ of $\mathcal{N}_S$ and $\{\mathcal{P}_{\text{act},i}^T\}_{i=1}^{|\mathcal{D}|}$ of $\mathcal{N}_T$, we can distill the domain-invariant knowledge of the teacher network to the student network using the following **d**omain-**i**nvariant **k**nowledge **t**ransfer loss function $\mathcal{L}_{\text{dikt}}(\mathcal{N}_S)$:

$$\mathcal{L}_{\text{dikt}}(\mathcal{N}_S) = \frac{1}{|\mathcal{D}|} \sum_{i=1}^{|\mathcal{D}|} ||\mathcal{P}_{\text{act},i}^S - \mathcal{P}_{\text{act},i}^T||_2. \tag{11}$$

During distillation, the student network is promoted to learn expressive channels among domain-invariant features contained by the teacher network under the guidance of the attention weights, thereby enhancing its performance in the target domain. In some cases, this feature transfer process might be hindered by the feature dimension mismatched between the $\mathcal{N}_T$ and $\mathcal{N}_S$. In the **Supplementary Materials**, we address this problem through a feature mapping operation and provide a pseudo-code of the fusion-activation mechanism.

### 3.3   Interactive Training of 4Ds

We propose an interactive framework to concurrently train both the teacher and student networks on the target data. During training, the teacher network, which possesses useful domain-invariant knowledge, can quickly generalize well in the target domain. Moreover, we allow the teacher network to train alongside the student network, which can stimulate a student-friendly teacher network [39]. As a result, the teacher network can provide accurate and easily digestible knowledge to the student network, therefore improving the distillation performance.

Specifically, the input target data $\mathcal{D} = \{(\mathbf{x}_i, y_i)\}_{i=1}^{|\mathcal{D}|}$ is first processed by both $\mathcal{N}_T$ and $\mathcal{N}_S$, producing the corresponding outputs $\{\mathcal{N}_T(\mathbf{x}_i)\}_{i=1}^{|\mathcal{D}|}$ and $\{\mathcal{N}_S(\mathbf{x}_i)\}_{i=1}^{|\mathcal{D}|}$. Then, a well-performed and student-friendly teacher network is achieved by minimizing the following integrated function:

$$\mathcal{L}_{\text{total}}(\mathcal{N}_T) = \mathcal{L}_{\text{ce}}(\mathcal{N}_T) + \beta \mathcal{L}_{\text{kt}}(\mathcal{N}_T), \tag{12}$$

where $\mathcal{L}_{\text{ce}}(\mathcal{N}_T) = \frac{1}{|\mathcal{D}|} \sum_{i=1}^{|\mathcal{D}|} \mathcal{H}_{\text{ce}}(\mathcal{N}_T(\mathbf{x}_i), y_i)$ is the **c**ross **e**ntropy loss function. $\mathcal{L}_{\text{kt}}(\mathcal{N}_T)$ is the **k**nowledge **t**ransfer loss function and it is defined as follows:

$$\mathcal{L}_{\text{kt}}(\mathcal{N}_T) = \frac{1}{|\mathcal{D}|} \sum_{i=1}^{|\mathcal{D}|} \text{KL}\left(\frac{\mathcal{N}_T(\mathbf{x}_i)}{\tau} \mid \frac{\mathcal{N}_S(\mathbf{x}_i)}{\tau}\right), \tag{13}$$

where $\tau = 4$ is a default temperature parameter controlling the softening degree of soft targets $\frac{\mathcal{N}_T(\mathbf{x}_i)}{\tau}$ and $\frac{\mathcal{N}_S(\mathbf{x}_i)}{\tau}$, which contain the category relations between the interested categories in $\mathcal{D}$, and $\text{KL}(\cdot|\cdot)$ represents the **K**ullback-**L**eibler divergence that measures the distribution discrepancies between two inputs. $\mathcal{L}_{\text{ce}}(\mathcal{N}_T)$ and $\mathcal{L}_{\text{kt}}(\mathcal{N}_T)$ motivate $\mathcal{N}_T$ to learn the knowledge specific to the target data and imitate the soft targets of $\mathcal{N}_S$, respectively, and their importance is tuned by a trade-off parameter $\beta > 0$.

Subsequently, the student network is trained on the target data while learning from the teacher network by the following total objective loss function:

$$\mathcal{L}_{\text{total}}(\mathcal{N}_S) = \mathcal{L}_{\text{ce}}(\mathcal{N}_S) + \beta \mathcal{L}_{\text{kt}}(\mathcal{N}_S) + \gamma \mathcal{L}_{\text{dikt}}(\mathcal{N}_S), \tag{14}$$

where $\gamma > 0$ serves as another trade-off parameter, $\mathcal{L}_{\text{kt}}(\mathcal{N}_S)$ encourages $\mathcal{N}_S$ to learn the readily learnable category relations from $\mathcal{N}_T$, and $\mathcal{L}_{\text{dikt}}(\mathcal{N}_S)$ (as in Eq. (11)) promotes $\mathcal{N}_S$ to learn the valuable domain-invariant knowledge inherited by $\mathcal{N}_T$. Noticeably, $\mathcal{L}_{\text{dikt}}(\mathcal{N}_T)$ will compel $\mathcal{N}_T$ to mimic the domain-invariant features produced by $\mathcal{N}_S$, leading to dramatic disturbance of the domain-invariant features inherited by $\mathcal{N}_T$. Therefore, we reasonably exclude $\mathcal{L}_{\text{dikt}}(\mathcal{N}_T)$ from $\mathcal{L}_{\text{total}}(\mathcal{N}_T)$ to preserve the valuable domain-invariant knowledge in $\mathcal{N}_T$. The training algorithm is summarized in the **Supplementary Materials**.

## 4   Experiments

In this section, we demonstrate the effectiveness of our proposed 4Ds on multiple popular benchmark datasets.

**Table 1:** Classification accuracies (in %) of the experiments on four DA datasets. The column "Avg. (4Ds)" provides the mean accuracy achieved by student networks trained via our 4Ds. The column "Avg.$^T$"/"Avg.$^S$" shows the average accuracy of four subsets among the corresponding dataset produced by the teacher/student networks trained by standard Back-Propagation (BP). The performance improvement of our 4Ds compared with "Avg.$^S$" is indicated in green font in the brackets. For the DA task "A2C", the teacher network is pre-trained in the Art domain and the student network is trained in the Clipart domain, and the same applies to other items.

| Dataset | Teacher | Student | A2C | A2P | A2R | C2A | C2P | C2R | P2A | P2C | P2R | R2A | R2C | R2P | Avg. (4Ds) | Avg.$^T$ | Avg.$^S$ |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Office-Home [49] | ResNet34 | ResNet18 | 81.41 | 92.12 | 84.86 | 71.81 | 92.11 | 85.20 | 72.42 | 81.33 | 85.44 | 72.63 | 81.67 | 92.22 | **82.77** (5.03) | 82.63 | 77.74 |
| | ResNeXt101 | ResNeXt50 | 84.10 | 95.49 | 89.44 | 83.74 | 95.60 | 89.22 | 84.74 | 84.87 | 89.22 | 84.36 | 84.76 | 95.49 | **88.42** (3.71) | 87.99 | 84.71 |
| | VGGNet16 | VGGNet13 | 80.52 | 90.76 | 80.61 | 64.99 | 89.63 | 81.88 | 65.63 | 80.42 | 81.42 | 65.02 | 79.83 | 90.76 | **79.29** (2.44) | 80.51 | 76.85 |
| | ResNet34 | VGGNet13 | 81.56 | 93.13 | 84.63 | 68.93 | 93.13 | 84.51 | 69.13 | 81.55 | 84.51 | 69.96 | 81.55 | 93.47 | **82.17** (5.32) | 82.63 | 76.85 |

| Dataset | Teacher | Student | A2C | A2D | A2W | C2A | C2D | C2W | D2A | D2C | D2W | W2A | W2C | W2D | Avg. (4Ds) | Avg.$^T$ | Avg.$^S$ |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Office-Caltech [16] | ResNet34 | ResNet18 | 96.44 | 100.00 | 100.00 | 97.39 | 100.00 | 98.31 | 97.40 | 96.88 | 100.00 | 97.15 | 97.38 | 100.00 | **98.33** (2.14) | 98.64 | 96.19 |
| | ResNeXt101 | ResNeXt50 | 99.55 | 100.00 | 100.00 | 97.39 | 100.00 | 100.00 | 97.91 | 99.56 | 100.00 | 97.35 | 99.55 | 100.00 | **99.27** (1.56) | 99.35 | 97.71 |
| | VGGNet16 | VGGNet13 | 95.55 | 96.87 | 98.30 | 97.91 | 100.00 | 98.30 | 97.39 | 96.44 | 100.00 | 97.35 | 95.55 | 100.00 | **97.80** (1.64) | 98.16 | 96.16 |
| | ResNet34 | VGGNet13 | 96.44 | 100.00 | 100.00 | 96.91 | 100.00 | 98.30 | 97.91 | 96.00 | 100.00 | 97.11 | 96.44 | 100.00 | **98.25** (2.09) | 98.64 | 96.16 |

| Dataset | Teacher | Student | A2C | A2P | A2S | C2A | C2P | C2S | P2A | P2C | P2S | S2A | S2C | S2P | Avg. (4Ds) | Avg.$^T$ | Avg.$^S$ |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| PACS [62] | ResNet34 | ResNet18 | 98.09 | 99.40 | 97.07 | 97.12 | 99.10 | 96.56 | 96.87 | 98.30 | 96.69 | 95.87 | 97.73 | 99.10 | **97.65** (2.98) | 96.55 | 94.67 |
| | ResNeXt101 | ResNeXt50 | 98.57 | 99.70 | 96.94 | 98.29 | 99.70 | 97.45 | 97.56 | 98.93 | 97.33 | 97.32 | 99.14 | 99.70 | **98.38** (3.31) | 97.64 | 95.07 |
| | VGGNet16 | VGGNet13 | 96.80 | 98.80 | 96.05 | 95.12 | 98.80 | 96.69 | 95.85 | 97.23 | 95.80 | 94.88 | 97.01 | 98.80 | **96.81** (3.31) | 95.29 | 93.50 |
| | ResNet34 | VGGNet13 | 94.45 | 99.40 | 96.43 | 93.90 | 99.40 | 96.43 | 95.41 | 97.45 | 96.56 | 94.90 | 96.45 | 99.40 | **96.68** (3.18) | 96.55 | 93.50 |

| Dataset | Teacher | Student | C2P | C2R | C2S | P2C | P2R | P2S | R2C | R2P | R2S | S2C | S2P | S2R | Avg. (4Ds) | Avg.$^T$ | Avg.$^S$ |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| DomainNet [41] | ResNet34 | ResNet18 | 67.61 | 80.63 | 66.35 | 74.49 | 80.50 | 66.05 | 74.77 | 68.53 | 66.25 | 75.18 | 68.73 | 80.46 | **72.46** (2.64) | 72.82 | 69.82 |
| | ResNeXt101 | ResNeXt50 | 73.99 | 84.89 | 71.23 | 78.63 | 84.77 | 71.40 | 78.84 | 74.22 | 71.89 | 78.75 | 73.89 | 84.84 | **77.27** (3.62) | 77.11 | 73.65 |
| | VGGNet16 | VGGNet13 | 67.82 | 79.79 | 64.68 | 73.52 | 79.98 | 64.76 | 73.91 | 67.67 | 64.85 | 74.11 | 67.71 | 79.81 | **71.55** (3.40) | 70.54 | 68.15 |
| | ResNet34 | VGGNet13 | 67.54 | 80.27 | 66.18 | 73.91 | 80.3 | 66.01 | 74.3 | 67.81 | 66.04 | 74.54 | 67.52 | 80.02 | **72.03** (3.88) | 72.82 | 68.15 |

**Benchmark Datasets**: We perform intensive experiments on four widely used benchmark datasets, including: **Office-Caltech** [16], composed of 10 classes with approximately 2,500 images across four distinct domains (Amazon, Caltech, Dslr, Webcam); **Office-Home** [49], contains 65 categories with 15,500 images belonging to four distinct domains (Art, Clipart, Product, Real); **PACS** [62], consists of 7 classes with about 10,000 images from four domains (Art, Cartoon, Photo, Sketch); and **DomainNet** [41], currently the largest dataset for DA, which contains six domains with 345 categories, we follow [30, 59] to select four domains (Clipart, Painting, Real, Sketch) for our experiments.

**Implementation Details**: All student networks in 4Ds employ Stochastic Gradient Descent (SGD) with weight decay of $5\times10^{-4}$ and momentum of 0.9 for optimization. Meanwhile, the adapters in the teacher network utilize Adam as the optimizer. All networks are trained over 120 epochs, the initial learning rates of the student network and adapters are 0.05 and 0.01, respectively, which are divided by 10 at the 40-th, 70-th, and 100-th epochs. Additionally, the trade-off parameters in Eq. (12) and Eq. (14) are set to $\beta$=1.0 and $\gamma$=0.1, respectively, and their parametric sensitivities are analyzed in the **Supplementary Materials**.

## 4.1 Experimental Results on Benchmark Datasets

In this section, we conduct intensive experiments across four DA datasets to evaluate the effectiveness of our proposed 4Ds. Specifically, we consider four widespread teacher-student pairs, including "ResNet34-ResNet18", "ResNeXt101-ResNeXt50", "VGGNet16-VGGNet13", and "ResNet34-VGGNet13". The teacher

**Table 2:** Average classification accuracies (in %) of the student networks trained by various methods on 12 tasks for each dataset. Enhancements (or reductions) in performance attributed to applied methods are denoted in green (or red) font in brackets.

| Type | Algorithm | Required data | Office-Home | Office-Caltech | PACS | DomainNet | Avg. |
|---|---|---|---|---|---|---|---|
| Standard | Teacher | Target | 82.63 | 98.64 | 96.55 | 72.82 | 87.66 |
| BP | Student | Target | 77.74 | 96.19 | 94.67 | 69.82 | 84.60 |
| Two stage | DAA-NRC | Target | 78.31 (0.57) | 95.17 (-1.02) | 91.82 (-2.85) | 65.81 (-4.01) | 82.77 (-1.83) |
| | DAA-SHOT | Target | 81.17 (3.43) | 95.75 (-0.44) | 95.67 (1.00) | 69.86 (0.04) | 85.61 (1.01) |
| | DAA-ProMix | Target | 80.47 (2.73) | 95.27 (-0.92) | 95.92 (1.25) | 70.24 (0.42) | 85.48 (0.88) |
| | DAA-BAIT | Target | 80.64 (2.90) | 95.71 (-0.48) | 95.16 (0.49) | 70.37 (0.55) | 85.47 (0.87) |
| | AAD-NRC | Target & source | 78.73 (0.99) | 96.10 (-0.09) | 91.89 (-2.78) | 66.68 (-3.14) | 83.35 (-1.25) |
| | AAD-SHOT | Target & source | 79.57 (1.83) | 96.67 (0.48) | 95.61 (0.94) | 70.22 (0.40) | 85.51 (0.91) |
| | AAD-ProMix | Target & source | 79.62 (1.88) | 95.61 (-0.58) | 95.28 (0.61) | 70.39 (0.57) | 85.23 (0.63) |
| | AAD-BAIT | Target & source | 80.21 (2.47) | 96.44 (0.25) | 95.89 (1.22) | 71.05 (1.23) | 85.90 (1.30) |
| Domain adaptation | SP [26] | Target & source | 71.59 (-6.15) | 88.33 (-7.86) | 81.33 (-13.34) | 55.81 (-14.01) | 74.26 (-10.34) |
| | CCDistill [15] | Target & source | 79.38 (1.64) | 95.75 (-0.44) | 93.17 (-1.50) | 66.67 (-3.15) | 83.74 (-0.86) |
| | HRKD [12] | Target & source | 79.11 (1.37) | 96.12 (-0.07) | 95.18 (0.51) | 68.37 (-1.45) | 84.69 (0.09) |
| | CDKD [34] | Target | 79.29 (1.55) | 93.07 (-3.12) | 89.53 (-5.14) | 66.36 (-3.46) | 82.06 (-2.54) |
| Traditional knowledge distillation | VKD [22] | Target | 78.99 (1.25) | 94.37 (-1.82) | 90.02 (-4.65) | 68.90 (-0.92) | 83.07 (-1.53) |
| | Fitnets [44] | Target | 78.52 (0.78) | 93.57 (-2.62) | 88.78 (-5.89) | 68.16 (-1.66) | 82.25 (-2.35) |
| | VID [1] | Target | 79.24 (1.50) | 94.37 (-1.82) | 90.05 (-4.62) | 68.74 (-1.08) | 83.10 (-1.50) |
| | SPKD [61] | Target | 78.53 (0.79) | 93.72 (-2.47) | 87.61 (-7.06) | 67.38 (-2.44) | 81.81 (-2.79) |
| | SemCKD [3] | Target | 75.55 (-2.19) | 93.79 (-2.40) | 85.91 (-8.76) | 66.70 (-3.12) | 80.48 (-4.12) |
| | SRRL [52] | Target | 79.84 (2.10) | 93.76 (-2.43) | 90.02 (-4.65) | 67.82 (-2.00) | 82.86 (-1.74) |
| | CATKD [19] | Target | 80.09 (2.35) | 92.48 (-3.71) | 88.64 (-6.03) | 69.92 (0.10) | 82.78 (-1.82) |
| | DIST [25] | Target | 80.55 (2.81) | 93.35 (-2.84) | 88.88 (-5.79) | 69.98 (0.16) | 83.19 (-1.41) |
| | DKD [60] | Target | 79.77 (2.03) | 95.97 (-0.22) | 92.71 (-1.96) | 70.77 (0.95) | 84.81 (0.21) |
| | USKD [56] | Target | 80.18 (2.44) | 96.87 (0.68) | 92.54 (-2.13) | 70.33 (-0.51) | 84.98 (0.38) |
| | OFAKD [20] | Target | 80.28 (2.54) | 95.99 (-0.20) | 86.56 (-8.11) | 66.02 (-3.80) | 82.21 (-2.39) |
| | FAMKD [42] | Target | 80.30 (2.56) | 93.83 (-2.36) | 88.95 (-5.72) | 71.02 (1.20) | 83.53 (-1.07) |
| Ours | 4Ds | Target | **82.77 (5.03)** | **98.33 (2.14)** | **97.65 (2.98)** | **72.46 (2.64)** | **87.80 (3.20)** |

networks are pre-trained on a specific subset (*i.e.*, the source data) among a DA dataset and guide the student network training on the remaining subsets (*i.e.*, the target data) of this DA dataset.

Table 1 reports the corresponding classification results. We can observe that the student networks trained by our 4Ds consistently outperform ones trained on the target data by standard back-propagation. Moreover, the performance of compact student networks trained by our 4Ds is comparable to that of the complicated teacher network trained on the target data. These experimental results demonstrate that our 4Ds can effectively address the distribution discrepancies between the source domain and target domain without using any source data, leading to a superior student network.

## 4.2    Comparison with Domain Adaptation Methods

In this subsection, we compare our 4Ds with two-stage methods discussed in Section 1, namely "Distillation After Adaptation" ("DAA") and "Adaptation After Distillation" ("AAD"). During the adaptation stage, we employ Neighborhood Reciprocity Clustering (NRC) [54], Source HypOthesis Transfer (SHOT) [35], Proxy Mixup (ProMix) [10], and BAIT [53] to generalize the teacher or student network from the source domain to the target domain. In the distillation stage, we follow VKD [22] to transfer softened category probabilities produced by the

teacher network to the student network. Moreover, we also compare our 4Ds with these DA methods using knowledge transfer, including Sharpened Predictions (SP) [26], Cross-domain Correlation Distillation (CCDistill) [15], Hierarchical Relational KD (HRKD) [12], and Cross Domain KD (CDKD) [34]; and traditional KD methods, including VKD [22], Fitnets [44], Variational Information Distillation (VID) [1], Similarity-Preserving KD (SPKD) [47], Semantic Calibration KD (SemCKD) [3], Softmax Regression Representation Learning (SRRL) [52], Class Attention Transfer based KD (CATKD) [19], DIstill from Stronger Teacher (DIST) [25], Decoupled KD (DKD) [60], Universal Self-KD (USKD) [56], One-For-All KD (OFAKD) [20], and Frequency Attention Module KD (FAMKD) [42]. All codes and parameter settings are obtained from their official GitHub pages, and all methods use the teacher-student pair ResNet34-ResNet18 to train on four DA datasets.

Table 2 presents the classification accuracies of the student networks trained by various compared methods and our 4Ds. Firstly, the student network trained by our one-stage 4Ds performs better than those trained by two-stage methods because our 4Ds can effectively circumvent the error accumulation problem inherent in two-stage methods. Meanwhile, our 4Ds also outperforms those DA methods utilizing knowledge transfer and traditional KD methods, which indicates that our 4Ds can effectively transfer the desired knowledge for the student network, and thus enhancing its performance.

### 4.3    Ablation Studies

We select the teacher-student pair ResNet34-ResNet18 to estimate the key operations in 4Ds on the Office-Home dataset, and the results are shown in Table 3. The contributions of these key operations are analyzed as follows:

**1) Transferred knowledge**. We train a student network that only mimics the categorical knowledge or domain-invariant features from the teacher network. It can be observed that the performance of the student network without mimicking the categorical knowledge or domain-invariant features dramatically decreases, especially for domain-invariant features. Furthermore, the student network learns from $\mathbf{f}_{\mathrm{ad}}^T$ with both domain-invariant and domain-specific information incurs an obvious performance degradation. This indicates that decoupling domain-invariant features in the teacher network is indeed necessary to train reliable student net-

**Table 3:** Average classification accuracies (in %) of the ablation experiments. The performance drop of each item compared with the complete 4Ds is indicated in red font in column "Avg.↓".

| Type | Algorithm | Avg. | Avg.↓ |
|---|---|---|---|
| Transferred knowledge | w/o $\mathcal{L}_{\mathrm{kt}}(\mathcal{N}_S)$ | 81.80 | 0.97 |
| | w/o $\mathcal{L}_{\mathrm{dikt}}(\mathcal{N}_S)$ | 80.25 | 2.52 |
| | w/o $\mathcal{L}_{\mathrm{kt}}(\mathcal{N}_S)$&$\mathcal{L}_{\mathrm{dikt}}(\mathcal{N}_S)$ | 77.74 | 5.03 |
| | $\mathcal{L}_{\mathrm{kt}}(\mathcal{N}_S)$&$\mathbf{f}_{\mathrm{ad}}^T$ | 81.73 | 1.04 |
| Transfering method | CKD [19] | 82.26 | 0.51 |
| | SPKD [47] | 81.75 | 1.02 |
| | MGD [55] | 81.95 | 0.82 |
| | SemcKD [3] | 81.38 | 1.39 |
| | ReviewerKD [7] | 82.23 | 0.54 |
| Weighting strategy | No weights | 80.07 | 2.70 |
| | Block-wise | 81.39 | 1.38 |
| | Channel-wise | 81.44 | 1.33 |
| | Block&Channel-wise | 81.83 | 0.94 |
| Training strategy | Fixed teacher | 79.64 | 3.13 |
| | Learnable teacher | 78.55 | 4.22 |
| | w/o $\mathcal{L}_{\mathrm{ce}}(\mathcal{N}_T)$ | 81.22 | 1.55 |
| | w/o $\mathcal{L}_{\mathrm{kt}}(\mathcal{N}_T)$ | 81.86 | 0.91 |
| 4Ds | $\mathcal{L}_{\mathrm{total}}(\mathcal{N}_S)$ | **82.77** | 0.00 |

works.

**2) Transferring method** for domain-invariant features. We replace the feature transfer method proposed in our 4Ds with advanced feature-based KD methods [3, 7, 19, 47, 55] to transfer domain-invariant features. We can observe that the original 4Ds outperforms those with advanced feature-based KD methods. This shows the effectiveness of our method in domain-invariant feature transfer.

**3) Weighting strategy** for domain-invariant features. We calculate "Block-wise" and "Channel-wise" weights (detailed in the **Supplementary Materials**) to transfer features from the teacher to the student. We can observe that 4Ds performs better than other strategies. This indicates that our fusion-activation mechanism, which captures the global dependencies of features from all blocks, can effectively transfer the domain-invariant features.

**4) Training strategy** for our 4Ds. We do not use adapters, instead, the student networks are learned from a fixed teacher network and a learnable teacher network pre-trained in the source domain. The former cannot grasp the suitable domain-specific knowledge from the target domain, and the latter cannot retain the valuable domain-invariant knowledge learned from the source domain. We can observe that both of them show suboptimal performance compared with our 4Ds. These results demonstrate that both preserving domain-invariant knowledge and updating domain-specific knowledge in our method are crucial for the student network to achieve satisfactory performance. Moreover, we also guide the training of the student network through a teacher network that uses adapters but without $\mathcal{L}_{\mathrm{ce}}(\mathcal{N}_T)$ or $\mathcal{L}_{\mathrm{kt}}(\mathcal{N}_T)$, and their performance is lower compared to the complete 4Ds. This indicates that a precise and student-friendly teacher network in the target domain is beneficial for training the student network.

## 5    Conclusion

This paper proposes a new KD method called 4Ds to learn a compact student network in a new target domain from the teacher network trained within the familiar source domain. To our best knowledge, we are the first to perform one-stage model compression between different domains. Our proposed 4Ds adopts three key operations to address the domain discrepancy between source and target domains, namely: 1) Decoupling of domain-invariant and domain-specific knowledge in the teacher network; 2) Distilling the useful domain-invariant knowledge inherited by the teacher network to the student network; 3) Adapting domain-specific knowledge of the teacher network in the target domain to provide reliable knowledge to the student network. The main advantage of our method is that 4Ds is a one-stage end-to-end framework, so it successfully alleviates the error accumulation from the two stages of traditional approaches and avoids time-consuming iterations between these stages. Therefore, 4Ds can effectively solve the domain gap to train reliable student networks in the target domain by only using teacher networks in the source domain.

## Acknowledgment

## References

1. Ahn, S., Hu, S.X., Damianou, A., Lawrence, N.D., Dai, Z.: Variational information distillation for knowledge transfer. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 9163–9171 (2019) 12, 13

2. Chen, D., Mei, J.P., Zhang, H., Wang, C., Feng, Y., Chen, C.: Knowledge distillation with the reused teacher classifier. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 11933–11942 (2022) 4

3. Chen, D., Mei, J.P., Zhang, Y., Wang, C., Wang, Z., Feng, Y., Chen, C.: Cross-layer distillation with semantic calibration. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 35, pp. 7028–7036 (2021) 4, 12, 13, 14

4. Chen, G., Peng, P., Ma, L., Li, J., Du, L., Tian, Y.: Amplitude-phase recombination: Rethinking robustness of convolutional neural networks in frequency domain. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV). pp. 458–467 (2021) 7

5. Chen, H., Guo, T., Xu, C., Li, W., Xu, C., Xu, C., Wang, Y.: Learning student networks in the wild. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 6428–6437 (2021) 4

6. Chen, H., Wang, Y., Xu, C., Yang, Z., Liu, C., Shi, B., Xu, C., Xu, C., Tian, Q.: Data-free learning of student networks. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV). pp. 3514–3522 (2019) 4

7. Chen, P., Liu, S., Zhao, H., Jia, J.: Distilling knowledge via knowledge review. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 5008–5017 (2021) 4, 13, 14

8. Choi, Y., Choi, J., El-Khamy, M., Lee, J.: Data-free network quantization with adversarial knowledge distillation. In: Proceedings of the IEEE/CVF Conference

on Computer Vision and Pattern Recognition Workshops (CVPR). pp. 710–711 (2020) 2

9. Cooley, J.W., Lewis, P.A., Welch, P.D.: The fast fourier transform and its applications. IEEE Transactions on Education (ToE) **12**(1), 27–34 (1969) 6

10. Ding, Y., Sheng, L., Liang, J., Zheng, A., He, R.: Proxymix: Proxy-based mixup training with label refinery for source-free domain adaptation. Neural Networks (NN) **167**, 92–103 (2023) 12

11. Do, K., Le, H., Nguyen, D., Nguyen, D., Harikumar, H., Tran, T., Rana, S., Venkatesh, S.: Momentum adversarial distillation: Handling large distribution shifts in data-free knowledge distillation. In: Advances in Neural Information Processing Systems (NeurIPS) (2022) 4

12. Dong, C., Li, Y., Shen, Y., Qiu, M.: Hrkd: hierarchical relational knowledge distillation for cross-domain language model compression. arXiv preprint arXiv:2110.08551 (2021) 4, 12, 13

13. Fang, G., Bao, Y., Song, J., Wang, X., Xie, D., Shen, C., Song, M.: Mosaicking to distill: Knowledge distillation from out-of-domain data. In: Advances in Neural Information Processing Systems (NeurIPS). vol. 34, pp. 11920–11932 (2021) 4

14. Feng, W., Ju, L., Wang, L., Song, K., Zhao, X., Ge, Z.: Unsupervised domain adaptation for medical image segmentation by selective entropy constraints and adaptive semantic alignment. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 37, pp. 623–631 (2023) 7

15. Gao, H., Guo, J., Wang, G., Zhang, Q.: Cross-domain correlation distillation for unsupervised domain adaptation in nighttime semantic segmentation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 9913–9923 (2022) 3, 4, 12, 13

16. Gong, B., Shi, Y., Sha, F., Grauman, K.: Geodesic flow kernel for unsupervised domain adaptation. In: 2012 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 2066–2073 (2012) 11

17. Gong, X., Sharma, A., Karanam, S., Wu, Z., Chen, T., Doermann, D., Innanje, A.: Preserving privacy in federated learning with ensemble cross-domain knowledge distillation. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 36, pp. 11891–11899 (2022) 4

18. Gou, J., Yu, B., Maybank, S.J., Tao, D.: Knowledge distillation: A survey. International Journal of Computer Vision (IJCV) **129**(6), 1789–1819 (2021) 2

19. Guo, Z., Yan, H., Li, H., Lin, X.: Class attention transfer based knowledge distillation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 11868–11877 (2023) 12, 13, 14

20. Hao, Z., Guo, J., Han, K., Tang, Y., Hu, H., Wang, Y., Xu, C.: One-for-all: Bridge the gap between heterogeneous architectures in knowledge distillation. Advances in Neural Information Processing Systems (NeurIPs) **36** (2024) 12, 13

21. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 770–778 (2016) 2, 7

22. Hinton, G., Vinyals, O., Dean, J.: Distilling the knowledge in a neural network. arXiv:1503.02531 (2015) 2, 4, 12, 13

23. Huang, J., Guan, D., Xiao, A., Lu, S.: Fsdr: Frequency space domain randomization for domain generalization. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 6891–6902 (2021) 4

24. Huang, J., Guan, D., Xiao, A., Lu, S.: Model adaptation: Historical contrastive learning for unsupervised domain adaptation without source data. In: Advances in

Neural Information Processing Systems (NeurIPS). vol. 34, pp. 3635–3649 (2021) 2

25. Huang, T., You, S., Wang, F., Qian, C., Xu, C.: Knowledge distillation from a stronger teacher. Advances in Neural Information Processing Systems (NeurIPS) **35**, 33716–33727 (2022) 12, 13

26. Islam, A., Chen, C.F.R., Panda, R., Karlinsky, L., Feris, R., Radke, R.J.: Dynamic distillation network for cross-domain few-shot recognition with unlabeled data. In: Advances in Neural Information Processing Systems (NeurIPS). vol. 34, pp. 3584–3595 (2021) 3, 4, 12, 13

27. Jiande, F., Weixin, X., Zongxiang, L.: A low complexity distributed multitarget detection and tracking algorithm. Chinese Journal of Electronics (CJE) **32**(3), 429–437 (2023) 2

28. LeCun, Y., Bengio, Y., Hinton, G.: Deep learning. nature **521**(7553), 436–444 (2015) 5, 6

29. Lee, S., Bae, J., Kim, H.Y.: Decompose, adjust, compose: Effective normalization by playing with frequency for domain generalization. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 11776–11785 (2023) 5, 7, 8

30. Li, J., Li, G., Shi, Y., Yu, Y.: Cross-domain adaptive clustering for semi-supervised domain adaptation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 2505–2514 (2021) 11

31. Li, J., Yu, Z., Du, Z., Zhu, L., Shen, H.T.: A comprehensive survey on source-free domain adaptation. IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI) (2024) 4

32. Li, L.: Self-regulated feature learning via teacher-free feature distillation. In: ECCV (2022) 4

33. Li, L., Dong, P., Wei, Z., Yang, Y.: Automated knowledge distillation via monte carlo tree search. In: ICCV (2023) 4

34. Li, W., Fan, K., Yang, H.: Teacher-student mutual learning for efficient source-free unsupervised domain adaptation. Knowledge-Based Systems **261**, 110204 (2023) 4, 12, 13

35. Liang, J., Hu, D., Wang, Y., He, R., Feng, J.: Source data-absent unsupervised domain adaptation through hypothesis transfer and labeling transfer. IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI) **44**(11), 8602–8617 (2021) 12

36. Long, M., Wang, J., Ding, G., Sun, J., Yu, P.S.: Transfer joint matching for unsupervised domain adaptation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 1410–1417 (2014) 4

37. Lu, W., Wang, J., Li, H., Chen, Y., Xie, X.: Domain-invariant feature exploration for domain generalization. arXiv preprint arXiv:2207.12020 (2022) 5

38. Oppenheim, A.V., Lim, J.S.: The importance of phase in signals. Proceedings of the IEEE **69**(5), 529–541 (1981) 5, 6

39. Park, D.Y., Cha, M.H., Kim, D., Han, B., et al.: Learning student-friendly teacher networks for knowledge distillation. In: Advances in Neural Information Processing Systems (NeurIPS). vol. 34, pp. 13292–13303 (2021) 10

40. Peng, B., Jin, X., Liu, J., Li, D., Wu, Y., Liu, Y., Zhou, S., Zhang, Z.: Correlation congruence for knowledge distillation. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV). pp. 5007–5016 (2019) 4

41. Peng, X., Bai, Q., Xia, X., Huang, Z., Saenko, K., Wang, B.: Moment matching for multi-source domain adaptation. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV). pp. 1406–1415 (2019) 11

42. Pham, C., Nguyen, V.A., Le, T., Phung, D., Carneiro, G., Do, T.T.: Frequency attention for knowledge distillation. In: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV). pp. 2277–2286 (2024) 12, 13

43. Piotrowski, L.N., Campbell, F.W.: A demonstration of the visual importance and flexibility of spatial-frequency amplitude and phase. Perception **11**(3), 337–346 (1982) 5

44. Romero, A., Ballas, N., Kahou, S.E., Chassang, A., Gatta, C., Bengio, Y.: Fitnets: Hints for thin deep nets. arXiv:1412.6550 (2014) 12, 13

45. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. International Conference on Learning Representations (ICLR) (2014) 2

46. Tang, J., Chen, S., Niu, G., Sugiyama, M., Gong, C.: Distribution shift matters for knowledge distillation with webly collected images. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV) (2023) 4

47. Tung, F., Mori, G.: Similarity-preserving knowledge distillation. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV). pp. 1365–1374 (2019) 4, 13, 14

48. Tzeng, E., Hoffman, J., Saenko, K., Darrell, T.: Adversarial discriminative domain adaptation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 7167–7176 (2017) 4

49. Venkateswara, H., Eusebio, J., Chakraborty, S., Panchanathan, S.: Deep hashing network for unsupervised domain adaptation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 5018–5027 (2017) 11

50. Wang, Z., Li, C., Wang, X.: Convolutional neural network pruning with structural redundancy reduction. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 14913–14922 (2021) 2

51. Xu, Q., Zhang, R., Zhang, Y., Wang, Y., Tian, Q.: A fourier-based framework for domain generalization. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 14383–14392 (2021) 5, 7, 8

52. Yang, J., Martinez, B., Bulat, A., Tzimiropoulos, G.: Knowledge distillation via adaptive instance normalization. arXiv:2003.04289 (2020) 12, 13

53. Yang, S., Wang, Y., Herranz, L., Jui, S., van de Weijer, J.: Casting a bait for offline and online source-free domain adaptation. Computer Vision and Image Understanding (CVIU) p. 103747 (2023) 12

54. Yang, S., van de Weijer, J., Herranz, L., Jui, S., et al.: Exploiting the intrinsic neighborhood structure for source-free domain adaptation. In: Advances in Neural Information Processing Systems (NeurIPS). vol. 34, pp. 29393–29405 (2021) 12

55. Yang, Z., Li, Z., Shao, M., Shi, D., Yuan, Z., Yuan, C.: Masked generative distillation. In: European Conference on Computer Vision (ECCV). pp. 53–69. Springer (2022) 13, 14

56. Yang, Z., Zeng, A., Li, Z., Zhang, T., Yuan, C., Li, Y.: From knowledge distillation to self-knowledge distillation: A unified approach with normalized loss and customized soft labels. arXiv preprint arXiv:2303.13005 (2023) 12, 13

57. Zhang, B., Zhang, X., Liu, Y., Cheng, L., Li, Z.: Matching distributions between model and data: Cross-domain knowledge distillation for unsupervised domain adaptation. In: Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics (ACL). pp. 5423–5433 (2021) 4

58. Zhang, R., Xie, C., Deng, L.: A fine-grained object detection model for aerial images based on yolov5 deep neural network. Chinese Journal of Electronics (CJE) **32**(1), 51–63 (2023) 2
59. Zhang, Z., Chen, W., Cheng, H., Li, Z., Li, S., Lin, L., Li, G.: Divide and contrast: Source-free domain adaptation via adaptive contrastive learning. In: Advances in Neural Information Processing Systems (NeurIPS). vol. 35, pp. 5137–5149 (2022) 11
60. Zhao, B., Cui, Q., Song, R., Qiu, Y., Liang, J.: Decoupled knowledge distillation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 11953–11962 (2022) 12, 13
61. Zhao, S., Wang, G., Zhang, S., Gu, Y., Li, Y., Song, Z., Xu, P., Hu, R., Chai, H., Keutzer, K.: Multi-source distilling domain adaptation. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 34, pp. 12975–12983 (2020) 12
62. Zhou, K., Yang, Y., Hospedales, T., Xiang, T.: Deep domain-adversarial image generation for domain generalisation. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 34, pp. 13025–13032 (2020) 11