Domain-Specific Conditional Jigsaw Adaptation Network for Enhancing Transferability and Discriminability

Qi He
Southwest Jiaotong University
Chengdu, China
qihe96@gmail.com

Zhaoquan Yuan*
Southwest Jiaotong University
Chengdu, China
zqyuan@swjtu.edu.cn

Xiao Wu
Southwest Jiaotong University
Chengdu, China
wuxiaohk@swjtu.edu.cn

Jun-Yan He
Alibaba DAMO Academy
Shenzhen, China
leyuan.hjy@alibaba-inc.com

ABSTRACT

Unsupervised Domain Adaptation (UDA) aims to transfer knowledge from a label-rich source domain to a target domain where the label is unavailable. Existing approaches tend to reduce the distribution discrepancy between the source and target domains or assign the pseudo target labels to implement a self-training strategy. However, the transferability or discriminability lackage of the traditional methods results in the limited ability to generalize on the target domain. To remedy this issue, a novel unsupervised domain adaptation framework called Domain-specific Conditional Jigsaw Adaptation Network (DCJAN) is proposed for UDA, which simultaneously encourages the network to extract transferable and discriminative features. To improve the discriminability, a conditional jigsaw module is presented to reconstruct class-aware features of the original images by reconstructing that of corresponding shuffled images. Moreover, in order to enhance the transferability, a domain-specific jigsaw adaptation is proposed to deal with the domain gaps, which utilizes the prior knowledge of jigsaw puzzles to reduce mismatching. It trains conditional jigsaw modules for each domain and updates the shared feature extractor to make the domain-specific conditional jigsaw modules could perform well not only on the corresponding domain but also on the other domain. A consistent conditioning strategy is proposed to ensure the safe training of conditional jigsaw. Experiments conducted on the widely-used Office-31, Office-Home, VisDA-2017, and DomainNet datasets demonstrate the effectiveness of the proposed approach, which outperforms the state-of-the-art methods.

CCS CONCEPTS

• **Computing methodologies** → **Transfer learning**; *Computer vision*; Unsupervised learning.

*Corresponding author: Zhaoquan Yuan

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.
MM '22, October 10–14, 2022, Lisbon, Portugal
© 2022 Association for Computing Machinery.
ACM ISBN 978-1-4503-9203-7/22/10…$15.00
https://doi.org/10.1145/3503161.3547890

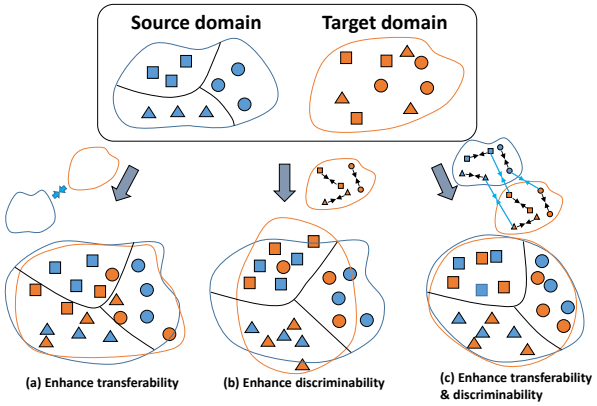


Figure 1: Comparison of traditional domain adaptation approaches and our proposed approach. (a) some methods focus on enhancing the transferability of both domains; (b) others utilize the pseudo-labels of the target domain to improve the discriminability of the target domain; (c) our method enhances the transferability and discriminability simultaneously.

KEYWORDS

Unsupervised Domain Adaptation, Neural Networks, Visual Classification, Self-supervised Learning

ACM Reference Format:
Qi He, Zhaoquan Yuan, Xiao Wu, and Jun-Yan He. 2022. Domain-Specific Conditional Jigsaw Adaptation Network for Enhancing Transferability and Discriminability . In *Proceedings of the 30th ACM International Conference on Multimedia (MM '22), October 10–14, 2022, Lisbon, Portugal.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3503161.3547890

1 INTRODUCTION

Currently, deep networks have achieved significant success on diverse vision tasks, including image recognition [18, 22, 59, 66], object detection [2, 19, 51, 52], semantic segmentation [7, 37, 53] and so on. With training deep networks on large-scale and well-labeling datasets, e.g., ImageNet [12] and MS COCO [33], it could easily attain state-of-the-art performance on given tasks. The bottleneck of applying the deep networks to real-world problems is that systems based on deep networks are unstable running in volatile situations, which is caused by the domain gaps between the training and real-world scenes. The ideal solution for model adaptation is transferring

the knowledge from the existing domain (dataset) to real-world ones instead of annotating the extra data from the target domain manually. Hence, the Unsupervised Domain Adaptation (UDA) framework is recently proposed and widely studied to remedy the domain gaps. It presents a task-guided knowledge transfer pipeline and derives a number of UDA methods [8, 15, 24, 25, 38, 40, 41, 61], which employ variant feature alignment strategies to obtain the domain-invariant features.

Traditional UDA methods can be summarized into two types: 1) the discrepancy metric-based methods, which tend to measure the discrepancy across distributions [25, 38, 40, 41, 61]; 2) the adversarial learning manner methods [8, 15, 24, 39], which learn domain-invariant features by training feature extractor to confuse the domain classifier. All these approaches focus on building a model to minimize the empirical risk on the labeled source domain and the feature alignment loss across domains. Ideally, by jointly optimizing the total losses, the model learns domain-invariant features for both domains, and the source-driven classifier generalizes well on the target domain.

However, the greatest challenge for UDA task still exists which is a conflict between discriminability and transferability. As shown in Figure 1(a), the features of the target domain lose their discriminability as overly penalizing domain variations, which results in that the crucial discriminative information being suppressed [10]. It leads to poor performance of target domain classification, particularly for the classifier is source-driven. To increase the discriminability of the target domain, recent works attempt to utilize pseudo labels of the target domain to perform self-training on the unlabeled target domain [17, 25, 39, 44, 46, 50, 68]. Nevertheless, as illustrated in Figure 1(b), some pseudo-label-based UDA methods direct utilize plausible pseudo labels, which causes the error accumulation of noisy labels.

To alleviate the aforementioned issues, this paper presents a Domain-specific Conditional Jigsaw Adaptation Network (DCJAN) for UDA, which guides the model to focus on the target domain as well as the source by considering both transferability and discriminability enhancement. Three aspects are comprehensively considered in DCJAN: 1) the self-learning in DCJAN builds up the domain-based knowledge prior to narrow the gap of the domain transfer; 2) the unity-knowledge prior is an extra constrain for the domain adaptation which improves the precision of alignment; 3) the class-aware consistency condition captures the cross-covariance dependency between the feature representations and class predictions. Specifically, the Jigsaw-solving task leveraged in DCJAN is adopted as a self-learning module, which learns the relationship between object-parts spatial correlation and the visual patch. This distinguishes what "kind" of the patches belong to and then predicts the original locations of the patches. This also constructs a fast and easy knowledge broadcast manner among the clusters corresponding to the source and target domain instead of transferring the knowledge between two disordered distributions. Moreover, the self-learning task constructs a mapping between object patch and geometry location as the knowledge prior for the source and target domain, respectively. It is a constrain in the latent semantic embedding space, which demands the domain adaptation not only aligning the feature and category information, but also the unity knowledge prior, helping improve the precision of domain alignment. Finally, the feature consistency condition constraints explicitly take jigsaw discrepancy between different categories into account by utilizing conditional jigsaw to reconstruct class-aware features of original images from features of shuffled. It means that the model plays a clustering role in model learning, which minimizes the distance of intra-class in the embedding feature space, enforces the cluster center separation, and improves the discriminability of the feature representation.

A novel training strategy is designed for DCJAN to obtain domain-invariant features. First, features are fed into the corresponding domain-specific conditional jigsaw module for each domain, respectively, and jointly update both the feature extractor and domain-specific conditional jigsaw modules to reconstruct the features of the original images. Second, the shared feature extractor is updated to promote the reconstruction for each domain using the domain-specific conditional jigsaw module of the other domain. In order to safe conditional jigsaw reconstruction, the prediction consistency conditioning strategy is presented to generate consistency-aware weight. It measures the priority of all conditional jigsaw training pairs and prevents the deterioration caused by conditional jigsaw with inconsistent predictions.

Our contributions can be summarized as follows:

- An autoencoder-based jigsaw framework is proposed for UDA that constructs a multitask pipeline by integrating jigsaw puzzle self-learning and conditioning constrain, making the model to learn the conditional jigsaw for each category to boost the discriminability.
- This work presents a novel jigsaw-based adaptation framework, which performs the knowledge transferring not only supervised by the direct feature space alignment but also under the guidance of domain-specific jigsaw modules, which aligns the inherent distribution structure (jigsaw puzzle) of both domains.
- To better train the DCJAN model, a novel training algorithm is designed by considering interactive jigsaw weight updating among both domains, which accelerates the model training and improves the performance.
- Extensive experiments conducted on multiple UDA benchmarks, Office-31, Office-Home, VisDA-2017, and DomainNet demonstrate that the proposed method achieves the state-of-the-art performance.

2 RELATED WORKS

2.1 Unsupervised Domain Adaptation

The purpose of UDA is to transfer the knowledge from a label-rich source domain to an unlabeled target domain. Recent works [8, 11, 15, 23, 24, 36, 39, 55, 65] aim to learn domain-invariant features along with the source and target domain. It can be mainly summarized into two types, and the first one measures the discrepancy between the source and target domains, then reduces the domain gaps by minimizing the discrepancy. For instance, Deep Domain Confusion (DDC) [61] utilizes the Maximum Mean Discrepancy (MMD) to measure the distribution discrepancy. Margin Disparity Discrepancy (MDD) is proposed in [67] to reduce the discrepancy across domains. Another powerful line of research is the adversarial training manner, which optimizes the two-player games of generator and discriminator to obtain the domain-invariant features. Domain Adversarial Neural Network (DANN) [15] learns

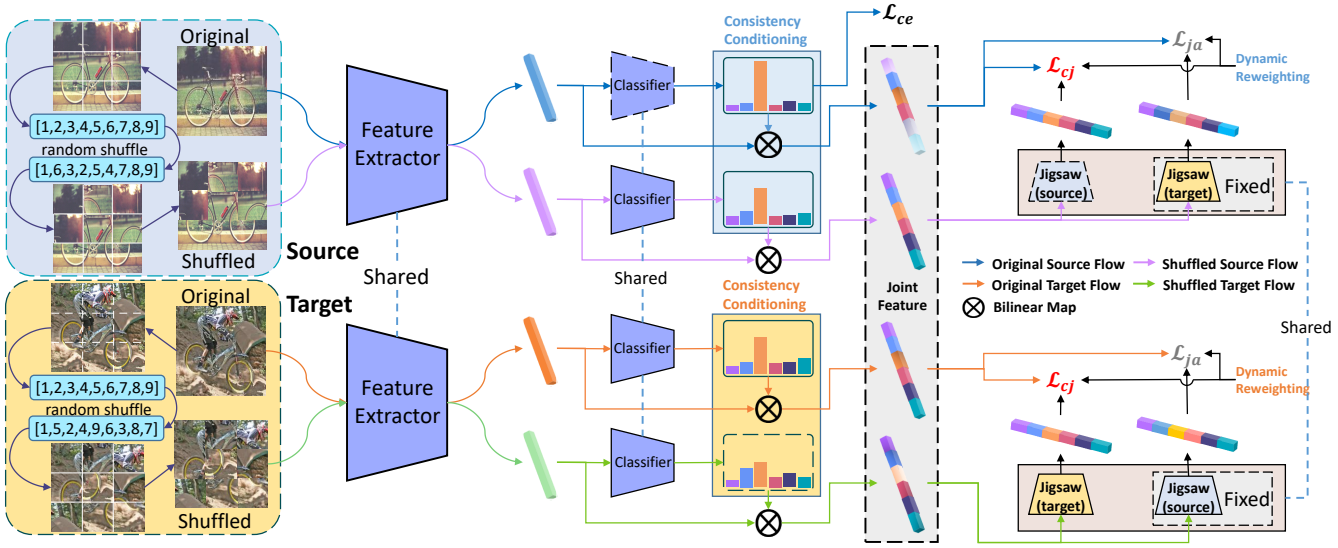


Figure 2: Illustration of our proposed DCJAN. It consists of three modules: conditional jigsaw module, domain-specific jigsaw adaptation module, and prediction consistency conditioning. For each image, the shuffled pipeline is shown in the left part, and the joint features are constructed to perform asymmetric reconstruction from the shuffled to the original. Conditional jigsaw module and domain-specific jigsaw adaptation module aim to enhance the discriminability and transferability under the guidance of jigsaw. The consistency conditioning is utilized to accelerate the training of conditional jigsaw.

domain-invariant features by applying an adversarial training strategy to train a feature generator and domain discriminator. A domain discriminator re-energize method is proposed in [24], which relabels the well-aligned samples of the target domain as samples of the source domain during the adversarial training phase. The task-special classifier is reused as a discriminator in [6] to align prediction-correlation across domains.

2.2 Pseudo-labels based Approaches

Traditional unsupervised domain adaptation approaches aim to learn high transferability features. However, the discriminability of the target domain would be decreased as the cost of the high transferability. Recent domain adaptation methods [17, 21, 25, 35, 42, 50, 64] attempt to use pseudo labels of the target domain to learn semantic features and enhance discriminability of the target domain. Self-training technique is adopted to the unlabeled target domain, [14] implements the self-ensembling methods to make the student network keep the consistent prediction with the teacher network for the unlabeled target domain. Cycle Self-Training (CST) [35] trains the domain-specific classifier and updates the feature extractor to make the output of domain-specific classifiers keep consistency. Others use the pseudo label of the target domain to apply class-specific feature alignment over different domains. [64] presents a moving semantic transfer network, which pushes the features of the same class but different domains closely. Conditional Kernel Bures (CKB) metric [41] aims to characterize conditional distribution discrepancy. Furthermore, the robust Pseudo-label loss [17] leverages the pseudo labels of the target domain and mitigates the negative transfer of the false pseudo labels. The negative transfer is considered in [50], which minimizes the entropy of reliable samples and maximizes the entropy of unreliable samples.

2.3 Solving Jigsaw Puzzles

Recently, solving jigsaw puzzles [3, 9, 28, 34, 43, 45, 47] has shown a remarkable advantage in self-supervised learning as the pretext task. These methods aim to capture the rich universal representation in the pre-trained model, which is powerful to be task-specific fined-tuned. Context Free Network (CFN) has been proposed in [45] to utilize solving jigsaw puzzles for learning whether each tile as an object part and how parts are combined to construct an object. Pretext-Invariant Representation Learning (PIRL) algorithm is presented in [43] , which learns invariant semantic information between the original images and the corresponding images randomly shuffled along patches. A novel jigsaw clustering pretext task is introduced in [9], which could take advantage of information from both intra- and inter-images. Moreover, solving jigsaw puzzles also be used in many specific tasks. [5] treats solving jigsaw puzzles as an auxiliary task for improving semantic understanding in domain generalization task, and GraphJigsaw is proposed in [32] to solving jigsaw puzzles at various stages with Graph Convolutional Network (GCN) [26] in cartoon face recognition. [4] introduces the jigsaw puzzle task into Partial Domain Adaptation to help reduce the domain gap in multi-task learning manner.

3 DOMAIN-SPECIFIC CONDITIONAL JIGSAW ADAPTATION

3.1 Preliminary

In UDA setting, given a labeled source domain $S = \{(x_i^s, y_i^s)\}_{i=1}^{n_s}$ and an unlabeled target domain $T = \{x_i^t\}_{i=1}^{n_t}$, where n_s and n_t denote the number of samples of the source and target domain, respectively. The label of the source domain is denoted as $y_i^s \in \{1, 2, \cdots, K\}$, where K is the number of classes, and both domains share the same

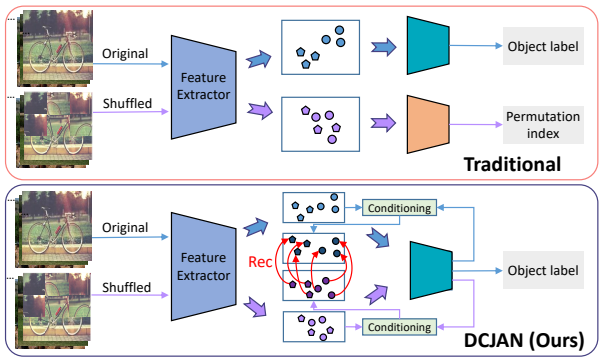


Figure 3: The difference between the traditional method and our proposed DCJAN method. Top: traditional methods insert solving jigsaw puzzle as an auxiliary classification task; Bottom: the proposed method combines the solving jigsaw puzzle and UDA task by replacing the prediction head of permutation index with an asymmetric reconstruction task.

label space. UDA attempts to learn a feature extractor $\mathcal{F}$ and classifier C on the both labeled source and the unlabeled target domains, which could make a precise prediction on the target domain.

3.2 Overview

The proposed Domain-specific Conditional Jigsaw Adaptation Network (DCJAN) will be introduced in this section. It attempts to keep the high transferability for both domains and make the target domain play an important role in the model training procedure as well as the source domain, such that the transferability and discriminability of the target domain can be improved at the same time. The framework of proposed DCJAN is shown in Figure 2. It contains three components: Conditional Jigsaw module, Domain-Specific Jigsaw Adaptation module, and Prediction Consistency Conditioning.

Conditional Jigsaw: It is a variant of the solving jigsaw puzzle method, which replaces the prediction head with an asymmetric autoencoder to solve conditional jigsaw puzzles by reconstructing class-aware joint features of original images from shuffled images.

Domain-Specific Jigsaw Adaptation: In this part, domain-specific jigsaw modules are trained for each domain. It aims to align the features across domains by optimizing feature extractor $\mathcal{F}$ with the proposed conditional jigsaw task.

Prediction Consistency Conditioning: The component measures the easy/hard-to-reconstruct examples. It leverages the consistency of the prediction between original images and corresponding shuffled images to perform the better reconstruction.

3.3 Conditional Jigsaw

To endow the model the ability to pay attention to the target domain as well as the source domain, the self-supervision is used in [13, 57, 58]. Here, the solving jigsaw puzzle technique is considered to UDA task. It is mainly used in self-supervised learning as a pretext task to enhance the understanding of the spatial relationship of images. In traditional solving jigsaw puzzles approaches, the extra shuffled images need to be generated from the original images as the inputs of jigsaw puzzles algorithms. Concretely, the original images are equally divided into $n \times n$ patches, and the shuffled images are re-assigned on these patches with random permutations, and the random permutations need to be recorded.

Traditional solving jigsaw puzzles algorithms put these shuffled images into the network, then get through a prediction head to predict the permutation used in building the shuffled images (the permutation was manually defined as class labels). Hence, if the prediction is equal to the predefined label, it can be seen as successfully solving the jigsaw puzzle of this image. Recent works add solving jigsaw puzzles methods into the object recognition as an auxiliary task to improve the understanding of images, but they do not take the discriminability into account. For instance, considering that humans can correctly recognize an object even if the image is shuffled along with patches, the intuitive humanlike method is to leverage shuffled images to solve the jigsaw puzzle and keep the consistency on the object predictions of original images and corresponding shuffled images simultaneously. Hence, an asymmetric reconstruction-based jigsaw puzzles solution and a conditional constraint are proposed to achieve this goal. The difference between traditional and the proposed method is shown in Figure 3. Traditional methods aim to learn the process of solving jigsaw puzzles, but ours could learn the results of jigsaw puzzles for each category. It is more powerful to capture the spatial relationships of object of different classes.

In the proposed approach, the predicting permutation head in traditional methods is replaced with an asymmetrical reconstruction task. It reconstructs the features of original images from that of corresponding shuffled images. The purpose of the reconstruction task is solve the jigsaw puzzle in the feature level. To keep consistent prediction, we apply the multilinear conditioning strategy of [39], which is defined as the outer product of multiple random vectors, to capture the multi-modal information and joint distributions of task-specific features and prediction categories. By using multi-modal information, it could attain optimizing the solving jigsaw puzzle under the guidance of conditional constrict.

Given the inputs x_i^s, x_j^t and the corresponding patch shuffled inputs $\hat{x}_i^s, \hat{x}_j^t$ from the source and target domains, where $i \in [1, 2, \cdots, n_s]$ and $j \in [1, 2, \cdots, n_t]$, we get the features $f_i^s, f_j^t, \hat{f}_i^s, \hat{f}_j^t$ and the predicted probability $p_i^s, p_j^t, \hat{p}_i^s, \hat{p}_j^t$, where $f = \mathcal{F}(x)$ and $p = C(f)$. Then, the joint distributions of f and p can be computed with bilinear map as follows:

$$g_i^s = f_i^s \otimes p_i^s,\ g_j^t = f_j^t \otimes p_j^t, \tag{1}$$

where $\otimes$ denotes the bilinear map operator and g denotes the joint feature. After computing the multimodal information, the conditional jigsaw module is utilized to reconstruct the joint features g of the original images from $\hat{g}$ of the shuffled images.

$$g_{rec(i)}^s = \mathcal{J}\left(\hat{f}_i^s \otimes \hat{p}_i^s\right),\ g_{rec(j)}^t = \mathcal{J}\left(\hat{f}_j^t \otimes \hat{p}_j^t\right), \tag{2}$$

where $\mathcal{J}$ is the conditional Jigsaw module for feature reconstruction. Finally, the objective function of conditional jigsaw can be formulated as follows:

$$\mathcal{L}_{cj}(\mathcal{F}, \mathcal{J}, C) = \frac{1}{n_s}\sum_{i=1}^{n_s} D\left(g_{rec(i)}^s, g_i^s\right) + \frac{1}{n_t}\sum_{j=1}^{n_t} D\left(g_{rec(j)}^t, g_j^t\right), \tag{3}$$

where D is the distance loss function. Following [17] to use sphere feature space, so the cosine similarity is selected to be the distance

loss function. Moreover, to satisfy minimizing optimization, D is defined as $D\left(g_{rec}, g\right) = 1.0 - cos\left(g_{rec}, g\right)$.

3.4 Domain-Specific Jigsaw Adaptation

This section introduces the details of that the novel jigsaw adaptation aligns the feature space of both domains under the guide of the conditional jigsaw puzzle task.

The conditional jigsaw module take the target domain into account, but it does not deal with the domain gaps between the source and target domains. Hence, a novel domain adaptation method based on the conditional jigsaw module is proposed to align features under the guidance of prior jigsaw puzzle knowledge.

The mainstream of feature alignment adds an extra domain discriminator to make extracted features fool the domain discriminator. In contrast, the proposed approach aims to align features with domain-specific conditional jigsaw. Two domain-specific Jigsaw modules are trained on source domain and target domain samples, respectively. That is, $\mathcal{J}_s$ reconstructs shuffled features to the original features on the source domain, and $\mathcal{J}_t$ reconstructs features on the target domain. Due to the domain gap, it is hard to let the trained domain-specific conditional jigsaw modules reconstruct the corresponding features from the shuffled of the other domain. Therefore, the feature extractor $\mathcal{F}$ is updated to achieve that the domain-specific jigsaw module trained on the target domain could solve conditional jigsaw puzzles on inputs of the source domain. Meanwhile, the training strategy is also applied to features of the target domain. In this way, the features extracted by $\mathcal{F}$ are domain-invariant for both domains.

In order to train domain-specific jigsaw modules, Eqn. 2 can be rewritten as follows:

$$g^s_{rec(i)} = \mathcal{J}_s\left(\hat{f}^s_i \otimes \hat{p}^s_i\right),\ g^t_{rec(j)} = \mathcal{J}_t\left(\hat{f}^t_j \otimes \hat{p}^t_j\right), \tag{4}$$

where $\mathcal{J}_s, \mathcal{J}_t$ are the domain-specific conditional jigsaw modules trained on the source and target domain, respectively.

Then the cross domains conditional jigsaw reconstruction between features and conditional jigsaw module can be depicted as:

$$\overline{g}^s_{rec(i)} = \mathcal{J}_t\left(\hat{f}^s_i \otimes \hat{p}^s_i\right),\ \overline{g}^t_{rec(j)} = \mathcal{J}_s\left(\hat{f}^t_j \otimes \hat{p}^t_j\right), \tag{5}$$

where $\overline{g}^s_{rec(i)}$ and $\overline{g}^t_{rec(j)}$ denote the reconstruction results by replacing the corresponding domain-specific conditional jigsaw module with the other. For instance, the difference between $g^s_{rec(i)}$ and $\overline{g}^s_{rec(i)}$ is that $g^s_{rec(i)}$ is the output of J_s and $\overline{g}^s_{rec(i)}$ is the output of J_t while the input is the same. The domain-specific jigsaw adaptation objective function is defined as following:

$$\mathcal{L}_{ja}\left(\mathcal{F}, C\right) = \frac{1}{n_s}\sum_{i=1}^{n_s} D\left(\overline{g}^s_{rec(i)}, g^s_i\right) + \frac{1}{n_t}\sum_{j=1}^{n_t} D\left(\overline{g}^t_{rec(j)}, g^t_j\right), \tag{6}$$

where D is the distance loss function introduced in Sec. 3.3.

3.5 Prediction Consistency Conditioning

Considering the conditional jigsaw reconstruction procedure may be deteriorated by the hard-to-reconstruct examples. These examples have inconsistent predictions of the classifier between the original and the corresponding shuffled images. However, the conditional jigsaw module treats all samples with equal importance. It prevents the convergence of solving conditional jigsaw puzzles.

Algorithm 1: DCJAN for UDA

Input: source dataset and target dataset $\boldsymbol{D_s}$ and $\boldsymbol{D_t}$, feature extractor $\mathcal{F}$, classifier C and domain-specific conditional jigsaw module $\mathcal{J}_s$, $\mathcal{J}_t$.

while *not converge* **do**
- Sample training batch $\boldsymbol{B_s}, \boldsymbol{B_t}$ from $\boldsymbol{D_s}, \boldsymbol{D_t}$, respectively;
- Generate the shuffled images batch $\hat{\boldsymbol{B}}_{\boldsymbol{s}}, \hat{\boldsymbol{B}}_{\boldsymbol{t}}$ from the corresponding inputs $\boldsymbol{B_s}, \boldsymbol{B_t}$;
- Calculate the cross-entropy loss $\mathcal{L}_{ce}\left(\mathcal{F}, C\right)$ of the source domain using Eqn. 7;
- Compute $\mathcal{L}_{cj}\left(\mathcal{F}, \mathcal{J}, C\right)$ using Eqn. 3 with $\mathcal{J}_s, \mathcal{J}_t$ and consistency-aware weight;
- Apply feature alignment and compute $\mathcal{L}_{ja}\left(\mathcal{F}, C\right)$ using Eqn. 6 with consistency-aware weight;
- Back-propagate with the total loss Eqn. 8;
- Update the total parameters;

end

Output: Learned feature extractor $\mathcal{F}$ and classifier C

Towards safe reconstruct, we quantify the inconsistency of classifier prediction between the original images and shuffled images by the Kullback–Leibler Divergence (KLD), $D(\hat{p}, p) = \sum_i^K p_i * log(\frac{p_i}{\hat{p}_i})$, where K is the number of categories. Those easy-to-reconstruct examples are given priority in jigsaw puzzle procedure ($\mathcal{L}_{cj}$ and $\mathcal{L}_{ja}$) by reweighting each example with the consistency-aware weight $w(D(\hat{p}, p)) = 1 + e^{-D(\hat{p}, p)}$. By reweighting loss functions Eqn. 3 and Eqn. 6 with consistency-aware weights, the model training is accelerated and the performance is improved.

3.6 Training Procedure

In training phase, it needs to minimize the conditional jigsaw reconstruction loss across domains and the domain-specific jigsaw adaptation loss at the same time. Besides, the network trains on the source domain by minimizing the cross entropy loss,

$$\mathcal{L}_{ce}(\mathcal{F}, C) = -\frac{1}{n_s}\sum_{i=1}^{n_s} y^s_i log p^s_i, \tag{7}$$

where y^s_i indicates the one-hot coding of ground-truth label of the i-th sample of the source domain.

Therefore, the overall objective can be formulated as:

$$\mathcal{L}_{ce}\left(\mathcal{F}, C\right) + \mathcal{L}_{cj}\left(\mathcal{F}, \mathcal{J}, C\right) + \mathcal{L}_{ja}\left(\mathcal{F}, C\right). \tag{8}$$

Algorithm 1 shows the whole training procedure, which combines the novel conditional jigsaw and domain-specific jigsaw adaptation into UDA to enhance the transferability and discriminability.

4 EXPERIMENTS

4.1 Datasets

We evaluated the proposed approach on the following standard benchmarks for UDA.

Office-31 [54]. Office-31 is a traditional domain adaptation dataset. It consists of three domain partitions: *Amazon*, *Webcam* and *DSLR* (abbr. **A**, **W** and **D**), and contains 31 categories. Three domains contain 2,817, 498 and 795 images, respectively.

Table 1: Accuracy (%) on Office-31 for unsupervised domain adaptation (ResNet-50). (Avg* denotes values except D→W and W→D)

Method	A→W	W→A	A→D	D→A	D→W	W→D	Avg	Avg*
ResNet-50 [20]	68.4±0.2	60.7±0.3	68.9±0.2	62.5±0.3	96.7±0.1	99.3±0.1	76.1	65.1
DANN [15]	82.0±0.4	67.4±0.5	79.7±0.4	68.2±0.4	96.9±0.2	99.1±0.1	82.2	74.2
CDAN [39]	94.1±0.1	69.3±0.3	92.9±0.2	71.0±0.3	98.6±0.1	**100.0±0.0**	87.7	81.8
CAN [25]	94.5±0.3	77.0±0.3	95.0±0.3	78.0±0.3	99.1±0.2	99.8±0.2	90.6	86.1
SRDC [60]	95.7±0.2	77.1±0.1	95.8±0.2	76.7±0.3	99.2±0.1	**100.0±0.0**	90.8	86.3
TSA [30]	96.0	76.8	95.4	76.7	98.7	**100.0**	90.6	86.2
SCDA [31]	94.8	76.4	94.6	77.5	98.2	**100.0**	90.3	85.8
FixBi [44]	96.1±0.2	79.4±0.3	95.0±0.4	78.7±0.5	**99.3±0.2**	**100.0±0.0**	91.4	87.3
MSTN [64]	91.3	65.6	90.4	72.7	98.9	**100.0**	86.5	80.0
MSTN+S [17]	94.6±0.3	76.0±0.6	91.3±0.7	75.4±0.7	98.5±0.2	**100.0±0.0**	89.3	84.3
RSDA-MSTN [17]	96.1±0.2	78.9±0.3	95.8±0.3	77.4±0.8	**99.3±0.2**	**100.0±0.0**	91.1	87.1
DCJAN (Ours)	**97.3±0.2**	**79.7±0.2**	**96.4±0.2**	**79.4±0.5**	**99.3±0.2**	**100.0±0.0**	**92.0**	**88.2**

Office-Home [63]. Office-Home is a well-organized benchmark but more challenging than Office-31 for visual domain adaptation, which contains 15,500 images of four different domains about the office and home scenes: *Artistic*, *Clipart*, *Product* and *Real world* (abbr. **Ar**, **Cl**, **Pr** and **Rw**) with 65 classes.

VisDA-2017 [49]. VisDA-2017 is a large-scale visual domain adaptation dataset, which is composed of 12 categories come from two domains: *synthetic* and *real*. The *synthetic* domain includes 152,397 images and the *real* domain includes 55,388 images.

DomainNet [48]. DomainNet is the largest and more challenging domain adaptation dataset. It consists of six diverse domains: *Clipart*, *Infograph*, *Painting*, *Quickdraw*, *Real* and *Sketch* (abbr. **clp**, **inf**, **pnt**, **qdr**, **rel** and **skt**), which has about 0.6 million images drawn from 345 categories.

4.2 Implementation Details

The proposed model is implemented with deep learning toolkit Pytorch. A ResNet-50 [20] network pre-trained on ImageNet [12] is served as the feature extractor $\mathcal{F}$, the last Fully-Connected (FC) layer is replaced with task-specific FC layers as the classifier C for the four benchmarks. The domain-specific conditional jigsaw modules $\mathcal{J}_s$ and $\mathcal{J}_t$ are conducted with FC layers trained from scratch. The full network is trained with back-propagation, where the parameters trained from scratch with a learning rate are 10 times that of the pre-trained parameters. We adopt mini-batch stochastic gradient descent (SGD) optimizer with a momentum 0.9 for fully network optimization. Following the setting of [16], the initial learning rate η is set to 0.01 and weight decay is 0.005. The learning rate η is adjusted by $\eta = \frac{0.01}{(1+\alpha p)^{\beta}}$, where $\alpha = 10$, $\beta = 0.75$ and p is the training progress linearly changing from 0 to 1. The average classification accuracy is reported on three random experiments.

4.3 Performance Comparison

The proposed method is compared with several state-of-the-art methods on four public benchmarks. Moreover, the variants of DANN and MSTN proposed in [17] are the baselines of our methods for comparing state-of-the-arts. The best accuracy is indicated in bold and the second best one is underlined.

Results on Office-31. Table 1 presents the results of transfer tasks on Office-31 dataset, where results of existing methods are reported in their respective papers. We observe that the proposed method outperforms the comparison methods on almost all transfer tasks and achieves an accuracy of 92.0% on average. Particularly, our method improves baseline MSTN-S from 84.3% to 88.2% on average* (except for the easy transfer tasks D→W and W→D), revealing that DCJAN can enhance the discriminability on this cross-domain dataset.

Results on Office-Home. The results on the Office-Home dataset are summarized in Table 2. Office-Home is a more challenging dataset with large domain discrepancy than the Office-31 dataset for domain adaptation. Our method empowers the baseline with 4.7% improvement and strongly improves tasks with the larger domain discrepancy, e.g., Ar → Cl, Ar → Pr, Cl → Ar tasks. Based on these results, we can refer that the proposed DCJAN could enhance transferability.

Results on VisDA-2017. The results on the VisDA-2017 dataset are reported in Table 3. The proposed method enhances the baseline by 14.5% and surpasses state-of-the-arts with ResNet-50 backbone. It proves the powerful ability of our proposed method for enhancing transferability and discriminability.

Results on DomainNet. The results on the DomainNet dataset are reported in Table 4. Our method improves the baseline by 4.2% and surpasses state-of-the-art methods. It shows the proposed method has a great ability of generalization on complex datasets.

4.4 Ablation Study

To have a clear understanding of each component of the proposed approach, we conduct comprehensive ablation studies on the Office-31 dataset with DANN+S [17] as the baseline model.

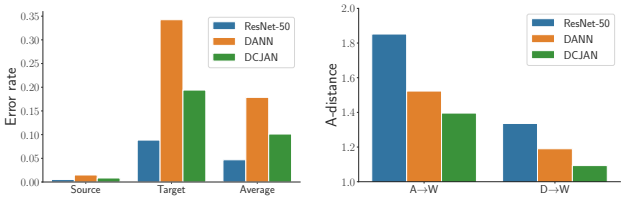
Effects of each component. Ablation studies are conducted to investigate the effects of each component of our proposed method, the details are depicted in Table 5. The proposed conditional jigsaw improves the baseline on average by 3.3% and 4.9% for average* (except tasks D→W and W→D) by enhancing the discriminability. We observe that the domain-specific jigsaw adaptation module massively improves the performance on the feature alignment method, indicating the proposed feature adaptation approach is compatible with others. Moreover, the consistency conditioning strategy

Table 2: Accuracy (%) on Office-Home for unsupervised domain adaptation (ResNet-50).

Method	Ar→Cl	Ar→Pr	Ar→Rw	Cl→Ar	Cl→Pr	Cl→Rw	Pr→Ar	Pr→Cl	Pr→Rw	Rw→Ar	Rw→Cl	Rw→Pr	Avg
ResNet-50 [20]	34.9	50.0	58.0	37.4	41.9	46.2	38.5	31.2	60.4	53.9	41.2	59.9	46.1
DANN [15]	45.6	59.3	70.1	47.0	58.5	60.9	46.1	43.7	68.5	63.2	51.8	76.8	57.6
CDAN [39]	49.0	69.3	74.5	54.4	66.0	68.4	55.6	48.3	75.9	68.4	55.4	80.5	63.8
SRDC [60]	52.3	76.3	81.0	69.5	76.2	78.0	68.7	53.8	81.7	76.3	57.1	85.0	71.3
TSA [30]	57.6	75.8	80.7	64.3	76.3	75.1	66.7	55.7	81.2	75.7	61.9	83.8	71.2
SCDA [31]	60.7	76.4	82.8	**69.8**	77.5	**78.4**	**68.9**	59.0	82.7	74.9	61.8	84.5	**73.1**
FixBi [44]	58.1	77.3	80.4	67.7	**79.5**	78.1	65.8	57.9	81.7	**76.4**	62.9	**86.7**	72.7
CST [35]	59.0	**79.6**	**83.4**	68.4	77.1	76.7	**68.9**	56.4	**83.0**	75.3	62.2	85.1	73.0
MSTN [64]	49.8	70.3	76.3	60.4	68.5	69.6	61.4	48.9	75.7	70.9	55.0	81.1	65.7
MSTN+S [17]	51.9	72.3	78.3	63.7	69.9	73.5	63.5	52.1	80.2	73.6	57.7	82.7	68.3
RSDA-MSTN [17]	53.2	77.7	81.3	66.4	74.0	76.5	67.9	53.0	82.0	75.8	57.8	85.4	70.9
DCJAN (Ours)	**61.2**	77.9	81.2	68.8	76.0	77.4	67.2	**59.4**	81.3	76.0	**63.2**	85.9	73.0

Table 3: Accuracy (%) on VisDA-2017 for unsupervised domain adaptation (ResNet-50).

Method	Synthetic→Real
CDAN [39]	70.0
MDD [67]	74.6
CST [35]	80.6
DANN [15]	63.7
DANN+S [17]	67.6
RSDA-DANN [17]	75.8
DCJAN (Ours)	**82.1**

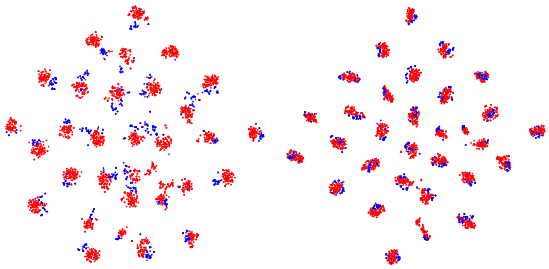


(a) Error Rate (b) A-distance

Figure 4: Measures of discriminability and transferability on learned features. (a) Classification error rate on each representation; (b) A-distance.

also shows the ability to improve performance, which makes the asymmetric reconstruction training pay attention to samples that keep the predictions consistency between the original and shuffled image. Overall, our method improves the baseline by an average from 86.7% to 91.1% and from 80.5% to 86.9% on average*. This shows that each component of DCJAN is effective for unsupervised domain adaptation tasks.

Ideal Joint Hypothesis. The ideal joint hypothesis is estimated to demonstrate the discriminability of feature embeddings, which can be achieved by training a multi-layer perceptrons (MLP) classifier with the pre-trained feature extractor on all source and target data with labels. As the analyzed in [10], it serves as a good description of discriminability. In Figure 4(a), although the pre-trained ResNet has a lower Error Rate than domain adversarial networks, our proposed approach significantly enhances the discriminability compared with baseline.



(a) DANN (b) DCJAN (Ours)

Figure 5: The T-SNE visualization of embedded features on the task A→W. Colors represent domains (red: source domain A; blue: target domain W).

Distribution discrepancy. The domain discrepancy could be measured by the A-distance [1]. The A-distance is defined as $d_A = 2(1-2\epsilon)$, where ϵ is the error rate of a domain classifier trained for distinguishing the source and target domain. It is used to quantify the transferability of feature embeddings. As shown in Figure 4(b), the A-distance is calculated on tasks A→W and W→D, which indicates the A-distance is smaller than DANN. It means that DCJAN could also enhance the transferability.

Comparison with different conditioning. The proposed consistency conditioning strategy is compared with the entropy conditioning strategy [39], which is proposed to reweight training examples for safety feature alignment by measuring the entropy. As shown in Table 6, although the entropy conditioning improves the performance by giving priority to samples having lower entropy, our consistency conditioning is more suitable for the proposed method.

Visualization. The task-specific features are visualized by using t-SNE [62] on transfer tasks A→W and W→A with Office-31 dataset in Figure 5. For DANN, the target domain features are wandering around the cluster of the source domain features. However, for the proposed method, the target domain features are closely embedded in the corresponding source domain feature cluster. It shows that the transferability and discriminability of DCJAN are better than DANN.

Table 4: Accuracy (%) on DomainNet for unsupervised domain adaptation (ResNet-50). In each sub-table, the column-wise domains are selected as the source domain and the row-wise domains are selected as the target domain. (* denotes implemented according to the original source code)

ResNet-50 [20]	clp	inf	pnt	qdr	rel	skt	Avg.
clp	-	14.2	29.6	9.5	43.8	34.3	26.3
inf	21.8	-	23.2	2.3	40.6	20.8	21.7
pnt	24.1	15.0	-	4.6	45.0	29.0	23.5
qdr	12.2	1.5	4.9	-	5.6	5.7	6.0
rel	32.1	17.0	36.7	3.6	-	26.2	23.1
skt	30.4	11.3	27.8	3.4	32.9	-	21.2
Avg.	24.1	11.8	24.4	4.7	33.6	23.2	20.3

CDAN [39]	clp	inf	pnt	qdr	rel	skt	Avg.
clp	-	13.5	28.3	9.3	43.8	30.2	25.0
inf	18.9	-	21.4	1.9	36.3	21.3	20.0
pnt	29.6	14.4	-	4.1	45.2	27.4	24.2
qdr	11.8	1.2	4.0	-	9.4	9.5	7.2
rel	36.4	18.3	40.9	3.4	-	24.6	24.7
skt	38.2	14.7	33.9	7.0	36.6	-	26.1
Avg.	27.0	12.4	25.7	5.1	34.3	22.6	21.2

SWD [27]	clp	inf	pnt	qdr	rel	skt	Avg.
clp	-	14.7	31.9	10.1	45.3	36.5	27.7
inf	22.9	-	24.2	2.5	33.2	21.3	20.0
pnt	33.6	15.3	-	4.4	46.1	30.7	26.0
qdr	15.5	2.2	6.4	-	11.1	10.2	9.1
rel	41.2	18.1	44.2	4.6	-	31.6	27.9
skt	44.2	15.2	37.3	10.3	44.7	-	30.3
Avg.	31.5	13.1	28.8	6.4	36.1	26.1	23.6

GDCAN [29]	clp	inf	pnt	qdr	rel	skt	Avg.
clp	-	18.2	41.9	16.5	58.7	44.0	35.9
inf	37.2	-	36.2	7.4	37.7	27.6	29.2
pnt	47.8	19.1	-	9.4	61.0	39.6	35.4
qdr	31.3	6.4	14.6	-	25.1	20.9	19.7
rel	52.3	20.4	48.5	9.8	-	37.6	33.7
skt	55.8	18.6	46.7	16.7	57.8	-	39.1
Avg.	44.9	16.5	37.6	12.0	48.1	33.9	32.2

MSTN+S* [17]	clp	inf	pnt	qdr	rel	skt	Avg.
clp	-	16.7	36.2	12.5	51.9	42.6	31.9
inf	30.1	-	30.2	2.5	43.5	26.4	26.5
pnt	41.5	18.1	-	4.1	53.1	36.2	30.6
qdr	23.1	2.3	7.2	-	14.9	16.3	12.7
rel	48.8	20.6	46.1	5.8	-	39.4	32.1
skt	52.1	17.9	41.5	14.2	50.6	-	35.2
Avg.	39.1	15.1	32.2	7.8	42.8	32.1	28.1

DCJAN	clp	inf	pnt	qdr	rel	skt	Avg.
clp	-	18.7	41.3	13.5	56.2	47.3	35.4
inf	36.1	-	36.6	4.5	50.3	31.8	31.8
pnt	46.3	19.2	-	6.3	58.4	41.5	34.3
qdr	28.1	4.3	10.3	-	21.8	19.8	16.8
rel	55.7	23.5	51.2	6.1	-	46.2	36.5
skt	58.5	20.6	46.2	14.5	56.1	-	39.2
Avg.	44.9	17.2	37.1	9.0	48.5	37.3	**32.3**

Table 5: Effect of proposed components on Office-31. (Avg* denotes values except D→W and W→D)

Baseline	Conditional jigsaw	Jigsaw adaptation	Conditioning	A→W	W→A	A→D	D→A	D→W	W→D	Avg	Avg*
√				93.2	71.0	87.5	70.3	98.0	100.0	86.7	80.5
√	√			95.2	76.0	93.5	77.0	98.4	100.0	90.0	85.4
√	√	√		96.3	76.4	94.3	77.8	98.7	100.0	90.6	86.2
√	√		√	96.2	76.2	94.3	77.5	98.6	100.0	90.5	86.0
√	√	√	√	96.7	77.5	95.1	78.1	99.0	100.0	91.1	86.9

Table 6: Effect of conditioning strategies on Office-31.

Method	A→W	W→A	A→D	D→A	Avg
w/o conditioning	95.2	76.0	93.5	77.0	85.4
+ Entropy conditioning	95.6	76.3	93.9	77.4	85.8
+ Consistency conditioning	96.2	76.2	94.3	77.5	86.0

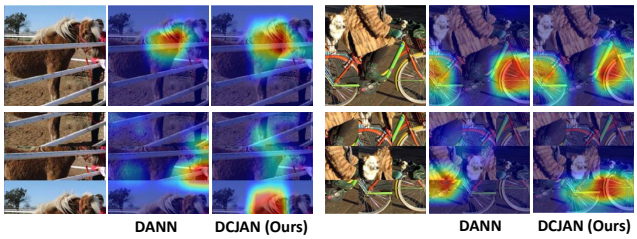


Figure 6: The Grad-CAM based visualization on VisDA-2017. The first row is the result of original images and the second row is the result of shuffled images.

The Grad-CAM [56] algorithm is utilized to visualize how the learned model predicts the shuffled images. It is an analytic technique to display the contribution level region of images for CNN models predicting by utilizing target gradients. As shown in Figure 6, DCJAN could understand the semantics of images and keeps prediction consistency between the original image and shuffled image, which is contrasted sharply with DANN.

5 CONCLUSION

This paper proposes a Domain-specific Conditional Jigsaw Adaptation Network (DCJAN) to ameliorate the absence of transferability or discriminability in traditional methods. By introducing jigsaw puzzles into UDA to achieve that model could take the target domain into account as well as the source domain. The proposed conditional jigsaw module improves the semantic understanding of each class, and the proposed domain-specific jigsaw adaptation and prediction consistency conditioning to efficiently enhance the transferability and discriminability. Comprehensive experiments on several cross-domain datasets demonstrate the efficacy of DCJAN.

ACKNOWLEDGMENTS

This work was supported in part by the National Natural Science Foundation of China (Grant No. 61772436, 62001400 and 61802053), Sichuan Science and Technology Program (Grant No. 2020YJ0207, 2020YJ0037 and 2021YJ0364), Foundation for Department of Transportation of Henan Province, China (2022-4-6 and 2019J-2-2), Grant of Institute of Applied Physics and Computational Mathematics, Beijing (Grant No. HXO2020-118) and China Postdoctoral Science Foundation (Grant No. 2020M683353).

REFERENCES

[1] Shai Ben-David, John Blitzer, Koby Crammer, Alex Kulesza, Fernando Pereira, and Jennifer W. Vaughan. 2010. A theory of learning from different domains. *Machine learning* 79, 1 (2010), 151–175.

[2] Alexey Bochkovskiy, Chien-Yao Wang, and Hong-Yuan M. Liao. 2020. Yolov4: Optimal speed and accuracy of object detection. *arXiv preprint arXiv:2004.10934* (2020).

[3] Dov Bridger, Dov Danon, and Ayellet Tal. 2020. Solving jigsaw puzzles with eroded boundaries. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 3526–3535.

[4] Silvia Bucci, Antonio D'Innocente, and Tatiana Tommasi. 2019. Tackling partial domain adaptation with self-supervision. In *Proceedings of International Conference on Image Analysis and Processing*. 70–81.

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

[6] Lin Chen, Huaian Chen, Zhixiang Wei, Xin Jin, Xiao Tan, Yi Jin, and Enhong Chen. 2022. Reusing the Task-specific Classifier as a Discriminator: Discriminator-free Adversarial Domain Adaptation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 7181–7190.

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

[8] Minghao Chen, Shuai Zhao, Haifeng Liu, and Deng Cai. 2020. Adversarial-learned loss for domain adaptation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 34. 3521–3528.

[9] Pengguang Chen, Shu Liu, and Jiaya Jia. 2021. Jigsaw clustering for unsupervised visual representation learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 11526–11535.

[10] Xinyang Chen, Sinan Wang, Mingsheng Long, and Jianmin Wang. 2019. Transferability vs. discriminability: Batch spectral penalization for adversarial domain adaptation. In *Proceedings of International Conference on Machine Learning*. 1081–1090.

[11] Zhi-Qi Cheng, Xiao Wu, Siyu Huang, Jun-Xiu Li, Alexander G. Hauptmann, and Qiang Peng. 2018. Learning to transfer: Generalizable attribute learning with multitask neural model search. In *Proceedings of the 26th ACM international conference on Multimedia*. 90–98.

[12] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. 2009. Imagenet: A large-scale hierarchical image database. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 248–255.

[13] Weijian Deng, Stephen Gould, and Liang Zheng. 2021. What does rotation prediction tell us about classifier accuracy under varying testing environments?. In *Proceedings of International Conference on Machine Learning*. 2579–2589.

[14] Geoffrey French, Michal Mackiewicz, and Mark Fisher. 2018. Self-ensembling for visual domain adaptation. In *Proceedings of International Conference on Learning Representations*.

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

[16] Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, Mario Marchand, and Victor Lempitsky. 2016. Domain-adversarial training of neural networks. *The journal of machine learning research* 17, 1 (2016), 2096–2030.

[17] Xiang Gu, Jian Sun, and Zongben Xu. 2020. Spherical space domain adaptation with robust pseudo-label loss. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 9101–9110.

[18] Jun-Yan He, Xiao Wu, Zhi-Qi Cheng, Zhaoquan Yuan, and Yu-Gang Jiang. 2021. DB-LSTM: Densely-connected Bi-directional LSTM for human action recognition. *Neurocomputing* 444 (2021), 319–331.

[19] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. 2017. Mask r-cnn. In *Proceedings of the IEEE International Conference on Computer Vision*. 2961–2969.

[20] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 770–778.

[21] Qi He, Qi Dai, Xiao Wu, and Jun-Yan He. 2021. A novel class restriction loss for unsupervised domain adaptation. *Neurocomputing* 461 (2021), 254–265.

[22] Gao Huang, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q. Weinberger. 2017. Densely connected convolutional networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 4700–4708.

[23] Siyu Huang, Xi Li, Zhi-Qi Cheng, Zhongfei Zhang, and Alexander Hauptmann. 2018. Gnas: A greedy neural architecture search method for multi-attribute learning. In *Proceedings of the 26th ACM international conference on Multimedia*. 2049–2057.

[24] Xin Jin, Cuiling Lan, Wenjun Zeng, and Zhibo Chen. 2021. Re-energizing Domain Discriminator with Sample Relabeling for Adversarial Domain Adaptation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 9154–9163.

[25] Guoliang Kang, Jiang Lu, Yi Yang, and Alexander G. Hauptmann. 2019. Contrastive adaptation network for unsupervised domain adaptation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 4893–4902.

[26] Thomas N. Kipf and Max Welling. 2017. Semi-supervised classification with graph convolutional networks. In *Proceedings of International Conference on Learning Representations*.

[27] Chen-Yu Lee, Tanmay Batra, Mohammad H. Baig, and Daniel Ulbricht. 2019. Sliced wasserstein discrepancy for unsupervised domain adaptation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 10285–10295.

[28] Ru Li, Shuaicheng Liu, Guangfu Wang, Guanghui Liu, and Bing Zeng. 2021. JigsawGAN: Self-supervised Learning for Solving Jigsaw Puzzles with Generative Adversarial Networks. *IEEE Transactions on Image Processing* 31 (2021), 513–524.

[29] Shuang Li, Chi Liu, Qiuxia Lin, Binhui Xie, Zhengming Ding, Gao Huang, and Jian Tang. 2020. Domain conditioned adaptation network. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 34. 11386–11393.

[30] Shuang Li, Mixue Xie, Kaixiong Gong, Chi Harold Liu, Yulin Wang, and Wei Li. 2021. Transferable semantic augmentation for domain adaptation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 11516–11525.

[31] Shuang Li, Mixue Xie, Fangrui Lv, Chi Harold Liu, Jian Liang, Chen Qin, and Wei Li. 2021. Semantic concentration for domain adaptation. In *Proceedings of the IEEE International Conference on Computer Vision*. 9102–9111.

[32] Yong Li, Lingjie Lao, Zhen Cui, Shiguang Shan, and Jian Yang. 2021. Graph jigsaw learning for cartoon face recognition. *arXiv preprint arXiv:2107.06532* (2021).

[33] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. 2014. Microsoft coco: Common objects in context. In *Proceedings of European Conference on Computer Vision*. 740–755.

[34] Chenxi Liu, Piotr Dollár, Kaiming He, Ross Girshick, Alan Yuille, and Saining Xie. 2020. Are labels necessary for neural architecture search?. In *Proceedings of European Conference on Computer Vision*. 798–813.

[35] Hong Liu, Jianmin Wang, and Mingsheng Long. 2021. Cycle self-training for domain adaptation. In *Proceedings of Advances in neural information processing systems*.

[36] Xiaofeng Liu, Site Li, Yubin Ge, Pengyi Ye, Jane You, and Jun Lu. 2021. Recursively conditional gaussian for ordinal unsupervised domain adaptation. In *Proceedings of the IEEE International Conference on Computer Vision*. 764–773.

[37] Jonathan Long, Evan Shelhamer, and Trevor Darrell. 2015. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 3431–3440.

[38] Mingsheng Long, Yue Cao, Jianmin Wang, and Michael Jordan. 2015. Learning transferable features with deep adaptation networks. In *Proceedings of International Conference on Machine Learning*. 97–105.

[39] Mingsheng Long, Zhangjie Cao, Jianmin Wang, and Michael I. Jordan. 2018. Conditional adversarial domain adaptation. In *Proceedings of Advances in neural information processing systems*. 1647–1657.

[40] Mingsheng Long, Han Zhu, Jianmin Wang, and Michael I Jordan. 2017. Deep transfer learning with joint adaptation networks. In *Proceedings of International Conference on Machine Learning*. 2208–2217.

[41] You-Wei Luo and Chuan-Xian Ren. 2021. Conditional Bures Metric for Domain Adaptation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 13989–13998.

[42] Ke Mei, Chuang Zhu, Jiaqi Zou, and Shanghang Zhang. 2020. Instance Adaptive Self-training for Unsupervised Domain Adaptation. In *Proceedings of European Conference on Computer Vision*, Vol. 12371. 415–430.

[43] Ishan Misra and Laurens van der Maaten. 2020. Self-supervised learning of pretext-invariant representations. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 6707–6717.

[44] Jaemin Na, Heechul Jung, Hyung Jin Chang, and Wonjun Hwang. 2021. FixBi: Bridging Domain Spaces for Unsupervised Domain Adaptation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 1094–1103.

[45] Mehdi Noroozi and Paolo Favaro. 2016. Unsupervised learning of visual representations by solving jigsaw puzzles. In *Proceedings of European Conference on Computer Vision*. 69–84.

[46] Yingwei Pan, Ting Yao, Yehao Li, Yu Wang, Chong-Wah Ngo, and Tao Mei. 2019. Transferrable prototypical networks for unsupervised domain adaptation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2239–2247.

[47] Marie-Morgane Paumard, David Picard, and Hedi Tabia. 2020. Deepzzle: Solving visual jigsaw puzzles with deep learning and shortest path optimization. *IEEE Transactions on Image Processing* 29 (2020), 3569–3581.

[48] Xingchao Peng, Qinxun Bai, Xide Xia, Zijun Huang, Kate Saenko, and Bo Wang. 2019. Moment matching for multi-source domain adaptation. In *Proceedings of the IEEE International Conference on Computer Vision*. 1406–1415.

[49] Xingchao Peng, Ben Usman, Neela Kaushik, Judy Hoffman, Dequan Wang, and Kate Saenko. 2017. Visda: The visual domain adaptation challenge. *arXiv preprint*

arXiv:1710.06924 (2017).

[50] Viraj Prabhu, Shivam Khare, Deeksha Kartik, and Judy Hoffman. 2021. SENTRY: Selective Entropy Optimization via Committee Consistency for Unsupervised Domain Adaptation. In *Proceedings of the IEEE International Conference on Computer Vision*. 8558–8567.

[51] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. 2016. You only look once: Unified, real-time object detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 779–788.

[52] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. 2015. Faster r-cnn: Towards real-time object detection with region proposal networks. In *Proceedings of Advances in neural information processing systems*. 91–99.

[53] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. 2015. U-net: Convolutional networks for biomedical image segmentation. In *Proceedings of International Conference on Medical image computing and computer-assisted intervention*. 234–241.

[54] Kate Saenko, Brian Kulis, Mario Fritz, and Trevor Darrell. 2010. Adapting visual category models to new domains. In *Proceedings of European Conference on Computer Vision*. 213–226.

[55] Kuniaki Saito, Kohei Watanabe, Yoshitaka Ushiku, and Tatsuya Harada. 2018. Maximum classifier discrepancy for unsupervised domain adaptation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 3723–3732.

[56] Ramprasaath R. Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. 2017. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *Proceedings of the IEEE International Conference on Computer Vision*. 618–626.

[57] Yu Sun, Eric Tzeng, Trevor Darrell, and Alexei A Efros. 2019. Unsupervised domain adaptation through self-supervision. *arXiv preprint arXiv:1909.11825* (2019).

[58] Yu Sun, Xiaolong Wang, Zhuang Liu, John Miller, Alexei Efros, and Moritz Hardt. 2020. Test-time training with self-supervision for generalization under distribution shifts. In *Proceedings of International Conference on Machine Learning*. 9229–9248.

[59] Mingxing Tan and Quoc Le. 2019. Efficientnet: Rethinking model scaling for convolutional neural networks. In *Proceedings of International Conference on Machine Learning*. 6105–6114.

[60] Hui Tang, Ke Chen, and Kui Jia. 2020. Unsupervised domain adaptation via structurally regularized deep clustering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 8725–8735.

[61] Eric Tzeng, Judy Hoffman, Ning Zhang, Kate Saenko, and Trevor Darrell. 2014. Deep domain confusion: Maximizing for domain invariance. *arXiv preprint arXiv:1412.3474* (2014).

[62] Laurens Van der Maaten and Geoffrey Hinton. 2008. Visualizing data using t-SNE. *Journal of machine learning research* 9, 11 (2008), 2579–2605.

[63] Hemanth Venkateswara, Jose Eusebio, Shayok Chakraborty, and Sethuraman Panchanathan. 2017. Deep hashing network for unsupervised domain adaptation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 5018–5027.

[64] Shaoan Xie, Zibin Zheng, Liang Chen, and Chuan Chen. 2018. Learning semantic representations for unsupervised domain adaptation. In *Proceedings of International Conference on Machine Learning*. 5423–5432.

[65] Jianfei Yang, Han Zou, Yuxun Zhou, Zhaoyang Zeng, and Lihua Xie. 2020. Mind the discriminability: Asymmetric adversarial domain adaptation. In *Proceedings of European Conference on Computer Vision*. 589–606.

[66] Ji Zhang, Jingkuan Song, Yazhou Yao, and Lianli Gao. 2021. Curriculum-based meta-learning. In *Proceedings of the 29th ACM International Conference on Multimedia*. 1838–1846.

[67] Yuchen Zhang, Tianle Liu, Mingsheng Long, and Michael Jordan. 2019. Bridging theory and algorithm for domain adaptation. In *Proceedings of International Conference on Machine Learning*. 7404–7413.

[68] Yixin Zhang, Zilei Wang, and Yushi Mao. 2021. Rpn prototype alignment for domain adaptive object detector. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 12425–12434.