

Class Conditional Alignment for Partial Domain Adaptation

Mohsen Kheirandishfard, Fariba Zohrizadeh, and Farhad Kamangar

Department of Computer Science and Engineering,

University of Texas at Arlington, USA

{mohsen.kheirandishfard, fariba.zohrizadeh}@mavs.uta.edu farhad.kamangar@uta.edu

Abstract—Adversarial adaptation models have demonstrated significant progress towards transferring knowledge from a labeled source dataset to an unlabeled target dataset. Partial domain adaptation (PDA) investigates the scenarios in which the source domain is large and diverse, and the target label space is a subset of the source label space. The main purpose of PDA is to identify the shared classes between the domains and promote learning transferable knowledge from these classes. In this paper, we propose a multi-class adversarial architecture for PDA. The proposed approach jointly aligns the marginal and class-conditional distributions in the shared label space by minimizing a novel multi-class adversarial loss function. Furthermore, we incorporate effective regularization terms to encourage selecting the most relevant subset of source domain classes. In the absence of target labels, the proposed approach is able to effectively learn domain-invariant feature representations, which in turn can enhance the classification performance in the target domain. Comprehensive experiments on three benchmark datasets Office-31, Office-Home, and Caltech-Office corroborate the effectiveness of the proposed approach in addressing different partial transfer learning tasks.

I. INTRODUCTION

With the impressive power of learning representations, deep neural networks have shown superior performance in a wide variety of machine learning tasks such as classification [1], [2], object detection [3], [2], [4], etc. These notable achievements heavily depend on the availability of large amounts of labeled training data. However, in many applications, collecting sufficient labeled data is either difficult or time-consuming. One potential solution to reduce the labeling consumption is to build an effective predictive model using readily-available labeled data from a different but related source domain. Such learning paradigm generally suffers from the distribution shift between the source and target domains, which in turn poses a significant difficulty in adapting the predictive model to the target domain tasks. In the absence of target labels, unsupervised domain adaptation (UDA) seeks to enhance the generalization capability of the predictive model by learning feature representations that are discriminative and domain-invariant [5], [6], [7]. Various approaches have been proposed in the literature to tackle UDA problems by embedding domain adaptation modules in deep architectures [8], [9], [10], [11], [12], [13] (see [14] for a comprehensive survey on deep domain adaptation methods). A line of research is developed to align the marginal distributions of the source and target

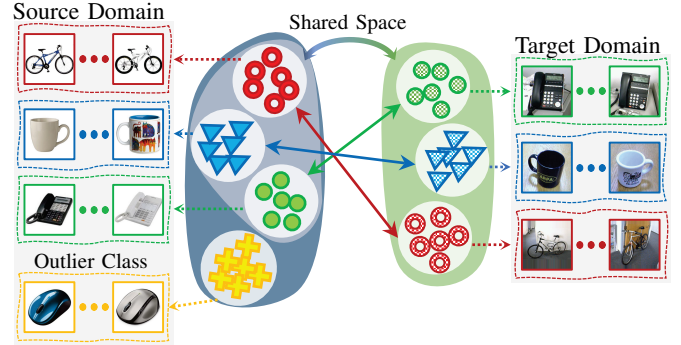


Fig. 1: Illustration of partial domain adaptation task. The objective is to transfer knowledge between the shared classes in the source and target domains. To this end, it is desired to identify and reject the outlier source classes and align both marginal and class-conditional distributions across the shared label space. *Best viewed in color.*

domains through minimizing discrepancy measures such as maximum mean discrepancy [15], [10], central moment discrepancy [16], correlation distance [17], [18], etc. In this way, they can map both domains into the same latent space, which results in learning domain-invariant feature representations. Another strand of research is focused on designing specific distribution normalization layers which facilitate learning separate statistics for the source and target domains [19], [20]. More recently, some research studies have been carried out based on the generative adversarial networks [21] that aim to alleviate the marginal disparities across the domains by adversarially learning domain-invariant feature representations which are indistinguishable for a discriminative domain classifier [22], [23], [24].

Despite the efficacy of the existing UDA methods, their superior performance is mostly limited to the scenarios in which the source and target domains share the same set of labels. With the goal of considering more realistic and practical cases, [25] introduced partial domain adaptation (PDA) as a new adaptation scenario in which the target label space is a subset of the source label space. The main challenge in PDA is to identify and reject the source domain classes

that do not appear in the target domain, known as *outlier classes*, mainly because they may exert negative impacts on the overall transfer performance [26], [27]. Addressing this challenge enables the PDA methods to transfer models trained on large and diverse labeled datasets (e.g. ImageNet) to small-scale unlabeled datasets from different but related domains.

In this paper, we propose a novel adversarial approach for partial domain adaptation which seeks to automatically reject the outlier source classes and improve the classification confidence on *irrelevant samples*, i.e. the samples that are highly dissimilar across the domains. The existing PDA methods often align the marginal distributions between the domains in the shared label space. Different from these methods, we propose a novel adversarial architecture that matches class-conditional feature distributions by minimizing a multi-class adversarial loss function. Moreover, we propose to boost the target domain classification performance by incorporating two novel regularization functions. The first regularizer is a row-sparsity term on the output of the classifier to promote the selection of a small subset of classes that are in common between the source and target domains. The second one is a minimum entropy term which increases the classifier confidence level in predicting the labels of irrelevant samples from both domains. We empirically observe that our proposed approach considerably improves the state-of-the-art performance for various partial domain adaptation tasks on three commonly-used benchmark datasets Office-31, Office-Home, and Caltech-Office.

II. RELATED WORK

To date, various unsupervised domain adaptation (UDA) methods have been developed to learn domain-invariant feature representations in the absence of target labels. Some studies have proposed to minimize the maximum mean discrepancy between the features extracted from the source and target samples [10], [13], [28], [29], [30]. In [31], a correlation alignment (CORAL) method is developed that utilizes a linear transformation to match the second-order statistics between the domains. [18] presented an extension of the CORAL method that learns a non-linear transformation to align the correlations of layer activations in deep networks. Despite the practical success of the aforementioned methods in domain alignment, it is shown that they are unable to completely eliminate the domain shift [9], [8]. Another line of work has proposed to reduce the discrepancy by learning separate normalization statistics for the source and target domains [19], [20]. [19] adopts different batch normalization layers for each domain to align the marginal distributions. [20] embeds domain alignment layers at different levels of a deep architecture to align the domain feature distributions to a canonical one.

More recently, adversarial adaptation methods have been extensively investigated to boost the performance of UDA methods [32], [33], [34], [35], [22]. The basic idea behind these methods is to train a discriminative domain classifier for predicting domain labels and a deep network for learning

feature representations that are indistinguishable by the discriminator. By doing so, the marginal disparities between the source and target domains can be efficiently reduced, which results in significant improvement in the overall classification performance [32], [35], [23]. Transferable attention for domain adaptation [24] proposed an adversarial attention-based mechanism for UDA, which effectively highlights the transferable regions or images. [36] introduced an incremental adversarial scheme which gradually reduces the gap between the domain distributions by iteratively selecting high confidence pseudo-labeled target samples to enlarge the training set. While the existing UDA models have shown tremendous progress towards reducing domain discrepancy, they mostly rely on the assumption of fully shared label space and generally align the marginal feature distributions between the source and target domains. This assumption is not necessarily valid in partial domain adaptation (PDA) which assumes the target label space is a subset of the source label space.

Great studies have been conducted towards the task of PDA to simultaneously promote positive transfer from the common classes between the domains and alleviate the negative transfer from the outlier classes [27], [25], [37]. Importance weighted adversarial nets [37] develops a two-domain classifier strategy to estimate the relative importance of the source domain samples. Selective adversarial network (SAN) [27] trains different domain discriminators for each source class separately to align the distributions of the source and target domains across the shared label space. Partial adversarial domain adaptation (PADA) [25] adopts a single adversarial network and incorporates class-level weights to both source classifier and domain discriminator for down-weighting the samples of outlier source classes. Example Transfer Network (ETN) [38] improves upon the PADA approach by introducing an auxiliary domain discriminator to quantify the transferability of each source sample.

Despite the efficacy of the existing PDA approaches in various tasks, they often align the marginal distributions of the shared classes between the domains without considering the conditional distributions [25], [36], [38]. This may degenerate the performance of the model due to the negative transfer of irrelevant knowledge. To circumvent this issue, we utilize pseudo-labels for the target domain samples and develop a multi-class adversarial architecture to jointly align the marginal and class-conditional distributions (see Figure 1 for more clarification). Inspired by [39], we propose to align labeled source centroid and pseudo-labeled target centroid to mitigate the adverse effect of the noisy pseudo-labels. Similar to [25], we incorporate class-level weights into our cost function to down-weight the contributions of the source samples belonging to the outlier classes. Furthermore, we introduce two novel regularization functions to promote the selection of a small subset of classes that are in common between the source and target domains and enhance the classifier confidence in predicting the labels of irrelevant samples from both domains.

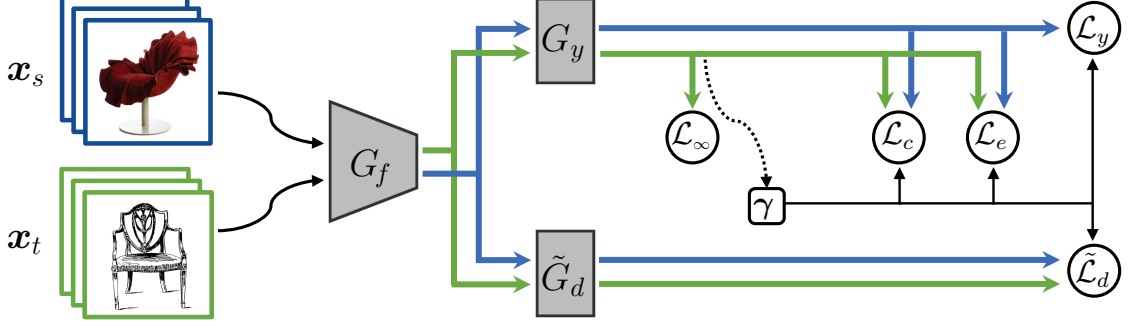


Fig. 2: Overview of the proposed adversarial architecture for partial transfer learning. The network consists of a feature extractor, a classifier, and a multi-class domain discriminator, denoted by G_f , G_y , and \tilde{G}_d , respectively. The blue arrows show the source flow and the green ones depict the target flow. Loss functions \mathcal{L}_y , $\tilde{\mathcal{L}}_d$, \mathcal{L}_c , \mathcal{L}_e , and \mathcal{L}_∞ denote the classification loss, the discriminative loss, the centroid alignment loss, the entropy loss, and the selection loss, respectively. Parameter γ is computed based on the classifier output to target samples and then is used to weight the loss of different classes. *Best viewed in color.*

III. PROBLEM FORMULATION

Let $\{(\mathbf{x}_s^i, \mathbf{y}_s^i)\}_{i=1}^{n_s}$ be a set of n_s sample images collected *i.i.d* from the source domain \mathcal{D}_s , where \mathbf{x}_s^i denotes the i^{th} source image with label \mathbf{y}_s^i . Similarly, let $\{\mathbf{x}_t^i\}_{i=1}^{n_t}$ be a set of n_t sample images drawn *i.i.d* from the target domain \mathcal{D}_t , where \mathbf{x}_t^i indicates the i^{th} target image. To clarify the notation, let $\mathcal{X} = \mathcal{X}_s \cup \mathcal{X}_t$ be the set of entire images captured from both domains, where $\mathcal{X}_s = \{\mathbf{x}_s^i\}_{i=1}^{n_s}$ and $\mathcal{X}_t = \{\mathbf{x}_t^i\}_{i=1}^{n_t}$. The UDA methods assume the source and target domains possess the same set of labels, denoted as \mathcal{C}_s and \mathcal{C}_t , respectively. In the absence of target labels, the primary goal of the UDA methods is to learn transferable features that can reduce the shift between the marginal distributions of both domains. One promising direction towards this goal is to train a domain adversarial network [11], [32] consisting of a discriminator G_d for predicting the domain labels, a feature extractor G_f to learn domain-invariant feature representations for deceiving the discriminator, and a classifier G_y that classifies the source domain samples. Training such adversarial network is equivalent to solving the following optimization problem

$$\max_{\theta_d} \min_{\theta_y, \theta_f} \frac{1}{n_s} \sum_{\mathbf{x}^i \in \mathcal{X}_s} L_y(G_y(G_f(\mathbf{x}^i; \theta_f); \theta_y), \mathbf{y}_s^i) - \frac{\lambda}{n} \sum_{\mathbf{x}^i \in \mathcal{X}} L_d(G_d(G_f(\mathbf{x}^i; \theta_f); \theta_d), d^i), \quad (1)$$

where $n = n_s + n_t$ denotes the total number of images, $\lambda > 0$ is a regularization parameter, \mathbf{y}_s^i is a one-hot vector denoting the class label of image \mathbf{x}^i , and $d^i \in \{0, 1\}$ indicates its domain label. L_y and L_d are cross-entropy loss functions corresponding to the classifier G_y and the domain discriminator G_d , respectively. Moreover, variables θ_f , θ_y , and θ_d are the network parameters associated with G_f , G_y , and G_d , respectively. For the brevity of notation, we drop the reference to the network parameters in the subsequent formulations.

As noted earlier, standard domain adaptation approaches assume that the source and target domains possess the same label space, i.e. $\mathcal{C}_s = \mathcal{C}_t$. This assumption may not be fulfilled in a wide range of practical applications in which \mathcal{C}_s is large and diverse (e.g., ImageNet) and \mathcal{C}_t only contains a small subset of the source classes (e.g., Office-31), i.e. $\mathcal{C}_t \subset \mathcal{C}_s$. In such scenarios, it is hard to identify the shared label space between the domains since target labels and target label space \mathcal{C}_t are unknown during the training procedure. Under this condition, matching the marginal distributions may not necessarily facilitate the classification task in the target domain and a classifier with adaptation may perform worse than a standard classifier trained on the source samples. This is attributed to the adverse effect of transferring information from the outlier classes $\mathcal{C}_s \setminus \mathcal{C}_t$ [27], [25]. Hence, the primary goal in partial domain adaptation is to identify and reject the outlier classes and simultaneously align the conditional distributions of the source and target domains across the shared label space. One of the well-established works toward this goal is Partial Adversarial Domain Adaptation (PADA) [25] which highlights the shared classes and reduces the importance of the outlier classes via the following weighting procedure

$$\gamma = \frac{1}{n_t} \sum_{i=1}^{n_t} \hat{\mathbf{y}}_t^i, \quad (2)$$

where $\hat{\mathbf{y}}_t^i = G_y(G_f(\mathbf{x}_t^i))$ denotes the output of the classifier G_y to the target sample \mathbf{x}_t^i and it can be considered as a probability distribution over the source label space \mathcal{C}_s . The weight vector γ is further normalized as $\gamma \leftarrow \gamma / \max(\gamma)$ to demonstrate the relative importance of the classes. The weights associated with the outlier classes are expected to be much smaller than that of the shared classes, mainly because the target samples are significantly dissimilar to the samples belonging to the outlier classes. Ideally, γ is expected to be a

vector whose elements are non-zero except those corresponding to the outlier classes. Given that, PADA proposed to down-weight the contributions of the source samples belonging to the outlier classes $\mathcal{C}_s \setminus \mathcal{C}_t$ by adding the class-level weight vector γ to both source classifier G_y and domain discriminator G_d . Therefore, the objective of PADA can be formulated as follows

$$\begin{aligned} \max_{\theta_d} \min_{\theta_y, \theta_f} & \frac{1}{n_s} \sum_{\mathbf{x}^i \in \mathcal{X}_s} \gamma_{c_i} L_y(G_y(G_f(\mathbf{x}^i)), \mathbf{y}^i) \\ & - \frac{\lambda}{n_s} \sum_{\mathbf{x}^i \in \mathcal{X}_s} \gamma_{c_i} L_d(G_d(G_f(\mathbf{x}^i)), d^i) \\ & - \frac{\lambda}{n_t} \sum_{\mathbf{x}^i \in \mathcal{X}_t} L_d(G_d(G_f(\mathbf{x}^i)), d^i), \end{aligned} \quad (3)$$

where scalar γ_{c_i} denotes the class weight of sample \mathbf{x}^i and $c_i = \arg\max_j y_j^i$ indicates the index of the largest element in vector \mathbf{y}^i .

IV. PROPOSED METHOD

Although the weighting scheme (2) is able to effectively match the marginal distributions of the source and target domains in the shared label space, there is no guarantee that the corresponding class-conditional distributions can also be drawn close. This may significantly degenerate the performance of the model due to the negative transfer of irrelevant knowledge. To circumvent this issue, we introduce a novel adversarial architecture to jointly align the marginal and class-conditional distributions in the shared label space. The proposed model adopts a multi-class discriminator \tilde{G}_d , parameterized by $\tilde{\theta}_d$, to classify the feature representations $G_f(\mathbf{x}^i)$ into $2 \times |\mathcal{C}_s|$ categories, where the first and the last $|\mathcal{C}_s|$ categories respectively correspond to the probability distribution over the source label space \mathcal{C}_s and target label space \mathcal{C}_t ($\mathcal{C}_t \subset \mathcal{C}_s$). We propose to train the discriminator \tilde{G}_d with the following objective function

$$\begin{aligned} \tilde{\mathcal{L}}_d(\theta_f, \tilde{\theta}_d) = & -\frac{1}{n_s} \sum_{\mathbf{x}^i \in \mathcal{X}_s} \gamma_{c_i} L_d(\tilde{G}_d(G_f(\mathbf{x}^i)), \tilde{\mathbf{d}}^i) \\ & - \frac{1}{n_t} \sum_{\mathbf{x}^i \in \mathcal{X}_t} L_d(\tilde{G}_d(G_f(\mathbf{x}^i)), \tilde{\mathbf{d}}^i), \end{aligned}$$

where vector $\tilde{\mathbf{d}}^i \in \mathbb{R}^{2 \times |\mathcal{C}_s|}$ is defined as the domain-class label of sample point \mathbf{x}^i . Due to the lack of class labels for the target samples, we set $\tilde{\mathbf{d}}^i$ to $[\mathbf{y}_s^i, \mathbf{0}]$ if $\mathbf{x}^i \in \mathcal{X}_s$ and use $[\mathbf{0}, \tilde{\mathbf{y}}_t^i]$ if $\mathbf{x}^i \in \mathcal{X}_t$, where $\tilde{\mathbf{y}}_t^i$ corresponds to the pseudo-label generated by classifier G_y and is given by $\tilde{\mathbf{y}}_t^i = \arg\max_c \mathbf{e}_c^\top G_y(G_f(\mathbf{x}_t^i))$, where $\{\mathbf{e}_c\}_{c=1}^{|\mathcal{C}_s|}$ denotes the standard unit basis in $\mathbb{R}^{|\mathcal{C}_s|}$. Moreover, the negative transfer can be efficiently alleviated by incorporating the weight vector γ into the loss $\tilde{\mathcal{L}}_d$ which results in selecting out the source samples belonging to the outlier label space $\mathcal{C}_s \setminus \mathcal{C}_t$. It is noteworthy to mention that the direct use of pseudo-labels may degrade the classification performance as the pseudo-labels are predicted by the classifier and hence they may be noisy and inaccurate. Many literature methods leverage the

theory of domain adaptation [40] to present error analysis and derive certain bounds on the error introduced by incorporating the pseudo-labels [41], [39]. These analysis are not generally applicable to the problem of partial domain adaptation as they mainly rely on the assumption that the source and target domains possess the same set of labels.

With the proposed multi-class adversarial loss $\tilde{\mathcal{L}}_d$, the key challenge is how to tackle the uncertainty in pseudo-labels. One promising approach to mitigate the adverse effect of falsely-pseudo-labeled target samples is to align labeled source centroids and pseudo-labeled target centroids in the feature space [39]. However, this approach hardly fits the partial domain adaptation scenario in which the target label space is a subset of source label space. We propose to modify the aforementioned approach by incorporating weight vector γ to highlight the mismatch between the centroids of the shared classes. Hence, the weighted centroid alignment loss function can be formulated as

$$\mathcal{L}_c(\theta_f, \theta_y) = \sum_{i=1}^{|\mathcal{C}_s|} \gamma_i \|M_s^i - M_t^i\|_2^2,$$

where M_s^i and M_t^i respectively denote the feature centroids for the i^{th} class in the source and target domains. These vectors are computed via the following formulas

$$M_s^i = \frac{1}{|\mathcal{O}_i|} \sum_{\mathbf{x}^i \in \mathcal{O}_i} G_f(\mathbf{x}^i), \quad M_t^i = \frac{1}{|\tilde{\mathcal{O}}_i|} \sum_{\mathbf{x}^i \in \tilde{\mathcal{O}}_i} G_f(\mathbf{x}^i),$$

where \mathcal{O}_i is the set of source samples belonging to the i^{th} class and $\tilde{\mathcal{O}}_i$ denotes the set of target samples assigned to the i^{th} class.

In what follows, we propose two novel regularization functions to derive more discriminative class weights and to increase the confidence level of the classifier in predicting the labels of the irrelevant samples across both domains.

Motivated by the assumption that the target samples are dissimilar to the samples of the outlier classes, we propose a row-sparsity regularization term that promotes the selection of a small subset of source domain classes that appear in the target domain. This, in turn, encourages the weight vector γ to be a vector of all zeros except for the elements corresponding to the shared classes. This selection regularization can be formulated as follows

$$\mathcal{L}_\infty(\theta_f, \theta_y) = \frac{1}{|\mathcal{C}_s|} \|G_y(G_f(\mathbf{x}_t^1)), \dots, G_y(G_f(\mathbf{x}_t^{|\mathcal{X}_t|}))\|_{1,\infty},$$

where $|\cdot|$ denotes the cardinality of its input set and $\|\cdot\|_{1,\infty}$ computes the sum of the infinity norms of the rows of an input matrix. To illustrate, for an arbitrary matrix $\mathbf{A} = [\mathbf{a}_1 | \mathbf{a}_2 | \dots | \mathbf{a}_n]^\top \in \mathbb{R}^{n \times m}$, scalar $\|\mathbf{a}_i\|_\infty$ denotes the maximum absolute value of i^{th} row. Therefore, regularization term $\|\mathbf{A}\|_{1,\infty} = \sum_{i=1}^n \|\mathbf{a}_i\|_\infty$ promotes sparsity on the maximum absolute value of each row which in turn leads to some zero rows in matrix \mathbf{A} .

The regularization term \mathcal{L}_∞ takes into consideration the relation between the entire target samples and encourages the

classifier to generate a sparse output vector with its non-zero entries located at certain indices correspond to the classes shared between the domains. Notice that this regularization term does not directly enforce a specific number of classes to be chosen but rather promotes the network to select a subset of source domain classes.

Besides the outlier classes, the irrelevant samples are inherently less transferable and they may significantly degrade the target classification performance in different PDA tasks. To reduce the negative effect of irrelevant samples in the training procedure, we propose to leverage the following entropy minimization term

$$\begin{aligned}\mathcal{L}_e(\theta_f, \theta_y) = & \frac{1}{n_s} \sum_{\mathbf{x}^i \in \mathcal{X}_s} \gamma_{c_i} L_y^e(G_y(G_f(\mathbf{x}^i))) \\ & + \frac{1}{n_t} \sum_{\mathbf{x}^i \in \mathcal{X}_t} L_y^e(G_y(G_f(\mathbf{x}^i))),\end{aligned}$$

where L_y^e is the entropy loss functions corresponding to the classifier G_y . Generally, regularization \mathcal{L}_e encourages the classifier to produce vectors with one dominant element denoting the label (or pseudo-label) of samples. This, in turn, enhances the performance of the feature extractor and helps to learn more transferable features for classification. Moreover, weight vector γ is incorporated to highlight the importance of samples belonging to the shared classes.

By combining the aforementioned loss functions, training our proposed model is equivalent to solving the following minimax saddle point optimization problem

$$\begin{aligned}\max_{\theta_d} \min_{\theta_y, \theta_f} & \frac{1}{n_s} \sum_{\mathbf{x}^i \in \mathcal{X}_s} \gamma_{c_i} L_y(G_y(G_f(\mathbf{x}^i)), \mathbf{y}^i) \\ & + \lambda \tilde{\mathcal{L}}_d(\theta_f, \tilde{\theta}_d) + \mathcal{L}_c(\theta_f, \theta_y) \\ & + \mu \mathcal{L}_\infty(\theta_f, \theta_y) + \zeta \mathcal{L}_e(\theta_f, \theta_y),\end{aligned}\quad (4)$$

where λ , μ , and ζ are positive hyperparameters to control the contribution of each loss component.

V. EXPERIMENTS

This section evaluates the efficacy of our approach, named CCPDA, through conducting empirical experiments on two widely used benchmark datasets for partial domain adaptation (PDA) problem. The experiments are performed on different PDA tasks in an unsupervised setting where neither the target labels nor the target label space is available. In what follows, we give more explanations about the datasets, the PDA tasks, and the network hyperparameters used in our experiments.

A. Setup

Dataset: We evaluate the performance of CCPDA on two commonly used datasets for the task of partial domain adaptation: Office-31 [42], Office-Home [43], and Caltech-Office. Office-31 object dataset consists of 4,652 images from 31 classes, where the images are collected from three different domains: *Amazon* (**A**), *Webcam* (**W**), and *DSLR* (**D**). We follow the procedure presented in the literature [25], [38] to

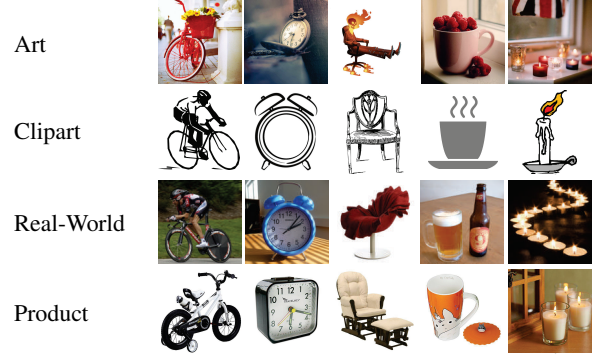


Fig. 3: Sample images from five classes in Office-Home dataset. Each column shows images from the same class but different domains.

transfer knowledge from a source domain with 31 classes to a target domain with 10 classes. The results are reported as the average classification accuracy of the target domain over five independent experiments across six different PDA tasks: **A** \rightarrow **W**, **W** \rightarrow **A**, **D** \rightarrow **W**, **W** \rightarrow **D**, **A** \rightarrow **D**, and **D** \rightarrow **A**.

Office-Home is a more challenging dataset that contains 15,500 images collected from four distinct domains: *Art* (**Ar**), *Clipart* (**Cl**), *Product* (**Pr**), and *Real-World* (**Rw**), where each domain has 65 classes. Example images from this dataset are provided in Figure 3. Following the procedure presented in [25], [38], we aim to transfer information from a source domain containing 65 classes to a target domain with 25 classes. The results on this dataset are also reported as the average classification accuracy of the target domain over five independent experiments across twelve pairs of source-target adaptation tasks: **Ar** \rightarrow **Cl**, **Ar** \rightarrow **Pr**, **Ar** \rightarrow **Rw**, **Cl** \rightarrow **Ar**, **Cl** \rightarrow **Pr**, **Cl** \rightarrow **Rw**, **Pr** \rightarrow **Ar**, **Pr** \rightarrow **Cl**, **Pr** \rightarrow **Rw**, **Rw** \rightarrow **Ar**, **Rw** \rightarrow **Cl**, and **Rw** \rightarrow **Pr**.

Caltech-Office [6] is constructed from Caltech-256 [44] dataset as the source domain and Office-31 as the target domain. Following the procedure in [27], we consider the ten categories shared by Caltech-256 and Office-31 as the shared label space. Denoting the source domain as **C**, the result on Caltech-Office dataset are reported as the average classification accuracy of the target domain over five independent experiments across three pairs of source-target adaptation tasks: **C** \rightarrow **W**, **C** \rightarrow **A**, and **C** \rightarrow **D**.

We follow the standard evaluation protocols for partial domain adaptation [27], [25] and compare the performance of CCPDA against several deep transfer learning methods: Reverse Gradient (RevGrad) [45], Domain Adversarial Neural Network (DANN) [32], Residual Transfer Networks (RTN) [28], Adversarial Discriminative Domain Adaptation (ADDA) [35], Importance Weighted Adversarial Nets (IWAN) [37], Multi-Adversarial Domain Adaptation (MADA) [23], Selective Adversarial Network (SAN) [27], Partial Adversarial Domain Adaptation (PADA) [25], and Example Transfer Network (ETN) [38]. Moreover, in order to demonstrate the efficacy brought by different components of the proposed

TABLE I: Classification accuracy of partial domain adaptation tasks on Office-31.

| Method | A \rightarrow W | D \rightarrow W | W \rightarrow D | A \rightarrow D | D \rightarrow A | W \rightarrow A | Avg |
|--------|-------------------|-------------------|-------------------|-------------------|-------------------|-------------------|--------------|
| ResNet | 75.59 | 96.27 | 98.09 | 83.44 | 83.92 | 84.97 | 87.05 |
| DANN | 73.56 | 96.27 | 98.73 | 81.53 | 82.78 | 86.12 | 86.50 |
| ADDA | 75.67 | 95.38 | 99.85 | 83.41 | 83.62 | 84.25 | 87.03 |
| MADA | 90.00 | 97.40 | 99.60 | 87.80 | 70.30 | 66.40 | 85.20 |
| RTN | 78.98 | 93.22 | 85.35 | 77.07 | 89.25 | 89.46 | 85.56 |
| IWAN | 89.15 | 99.32 | 99.36 | 90.45 | 95.62 | 94.26 | 94.69 |
| SAN | 93.90 | 99.32 | 99.36 | 94.27 | 94.15 | 88.73 | 94.96 |
| PADA | 86.54 | 99.32 | 100.0 | 82.17 | 92.69 | 95.41 | 92.69 |
| ETN | 94.52 | 100.0 | 100.0 | 95.03 | 96.21 | 94.64 | 96.73 |
| CCPDA | 99.66 | 100.0 | 100.0 | 97.45 | 95.72 | 95.71 | 98.09 |

TABLE II: Classification accuracy of partial domain adaptation tasks on Office-Home.

| Method | Ar \rightarrow Cl | Ar \rightarrow Pr | Ar \rightarrow Rw | Cl \rightarrow Ar | Cl \rightarrow Pr | Cl \rightarrow Rw | Pr \rightarrow Ar | Pr \rightarrow Cl | Pr \rightarrow Rw | Rw \rightarrow Ar | Rw \rightarrow Cl | Rw \rightarrow Pr | Avg |
|--------|---------------------|---------------------|---------------------|---------------------|---------------------|---------------------|---------------------|---------------------|---------------------|---------------------|---------------------|---------------------|--------------|
| ResNet | 46.33 | 67.51 | 75.87 | 59.14 | 59.94 | 62.73 | 58.22 | 41.79 | 74.88 | 67.40 | 48.18 | 74.17 | 61.35 |
| DANN | 43.76 | 67.90 | 77.47 | 63.73 | 58.99 | 67.59 | 56.84 | 37.07 | 76.37 | 69.15 | 44.30 | 77.48 | 61.72 |
| ADDA | 45.23 | 68.79 | 79.21 | 64.56 | 60.01 | 68.29 | 57.56 | 38.89 | 77.45 | 70.28 | 45.23 | 78.32 | 62.82 |
| RTN | 49.31 | 57.70 | 80.07 | 63.54 | 63.47 | 73.38 | 65.11 | 41.73 | 75.32 | 63.18 | 43.57 | 80.50 | 63.07 |
| IWAN | 53.94 | 54.45 | 78.12 | 61.31 | 47.95 | 63.32 | 54.17 | 52.02 | 81.28 | 76.46 | 56.75 | 82.90 | 63.56 |
| SAN | 44.42 | 68.68 | 74.60 | 67.49 | 64.99 | 77.80 | 59.78 | 44.72 | 80.07 | 72.18 | 50.21 | 78.66 | 65.30 |
| PADA | 51.95 | 67.00 | 78.74 | 52.16 | 53.78 | 59.03 | 52.61 | 43.22 | 78.79 | 73.73 | 56.60 | 77.09 | 62.06 |
| ETN | 59.24 | 77.03 | 79.54 | 62.92 | 65.73 | 75.01 | 68.29 | 55.37 | 84.37 | 75.72 | 57.66 | 84.54 | 70.45 |
| CCPDA | 55.31 | 80.11 | 88.07 | 73.28 | 71.21 | 77.63 | 71.89 | 52.97 | 81.41 | 81.81 | 56.21 | 85.15 | 72.92 |

TABLE III: Classification accuracy of partial domain adaptation tasks on Caltech-Office.

| Method | C \rightarrow W | C \rightarrow A | C \rightarrow D | Avg |
|---------|-------------------|-------------------|-------------------|--------------|
| RevGrad | 54.57 | 72.86 | 57.96 | 61.80 |
| ADDA | 73.66 | 78.35 | 74.80 | 75.60 |
| RTN | 71.02 | 81.32 | 62.35 | 71.56 |
| SAN | 88.33 | 83.82 | 85.35 | 85.83 |
| CCPDA | 95.23 | 88.05 | 100.0 | 94.42 |

PDA model, we conduct an ablation study by evaluating three variants of CCPDA: CCPDA $_{\infty}$ is a variant of CCPDA without incorporating the selection regularization term \mathcal{L}_{∞} , CCPDA $_e$ denotes a variant without considering \mathcal{L}_e , and CCPDA $_{d,c}$ is a variant with a binary discriminator and without considering the weighted centroids alignment term \mathcal{L}_c .

Parameter: We use PyTorch to implement CCPDA and adopt ResNet-50 [46] model pre-trained on ImageNet [47], as the backbone for the network G_f . We fine-tune the entire feature layers and apply back-propagation to train the domain discriminator \tilde{G}_d and the classifier G_y . Since parameters θ_y and $\tilde{\theta}_d$ are trained from scratch, their learning rates are set to be 10 times greater than that of θ_f . To solve the minimax problem (3), we use mini-batch stochastic gradient descent (SGD) with a momentum of 0.95 and the learning rate is adjusted during SGD by: $\eta = \frac{\eta_0}{(1+\alpha \times \rho)^\beta}$ where $\eta_0 = 10^{-2}$, $\alpha = 10$, $\beta = 0.75$, and ρ , denoting the training progress, linearly changes from 0 to 1 [32], [25]. We use a batch size $b = 72$ with 36 samples for each domain. Parameter μ is set to 0.1 for Office-31, Office-Home, and Caltech-Office datasets. Notice that since the classifier is not appropriately trained in

the first few epochs, the value of μ can be gradually increased from 0 to 0.1. Other hyper-parameters are tuned by importance weighted cross validation [48] on labeled source samples and unlabeled target samples.

As we use mini-batch SGD for optimizing our model, categorical information in each batch is usually inadequate for obtaining an accurate estimation of the source and target centroids. This in turn may adversely affect the alignment performance. To mitigate this issue, we align the moving average centroids of the source and target classes in the feature space (with coefficient 0.7) rather than aligning the inaccurate centroids obtained in each iteration.

B. Results

The target domain classification accuracy for various methods on six PDA tasks of Office-31 dataset, twelve PDA tasks of Office-Home dataset, and three PDA tasks of Caltech-Office dataset are reported in Tables I, II, and III. The entire results are reported based on the ResNet-50 and the scores of the competitor methods are directly collected from [27] and [38].

Observe that unsupervised domain adaptation methods such as ADDA, DANN, and MADA have exhibited worse performance than the standard ResNet-50 on some PDA tasks in both datasets Office-31 and Office-Home. This can be attributed to the fact that these methods aim to align the marginal distributions across the domains and hence are prone to the negative transfer introduced by the outlier classes. On the other hand, the partial domain adaptation methods, such as PADA, SAN, IWAN, ETN, and CCPDA, achieve promising results on most of the PDA tasks since they leverage different

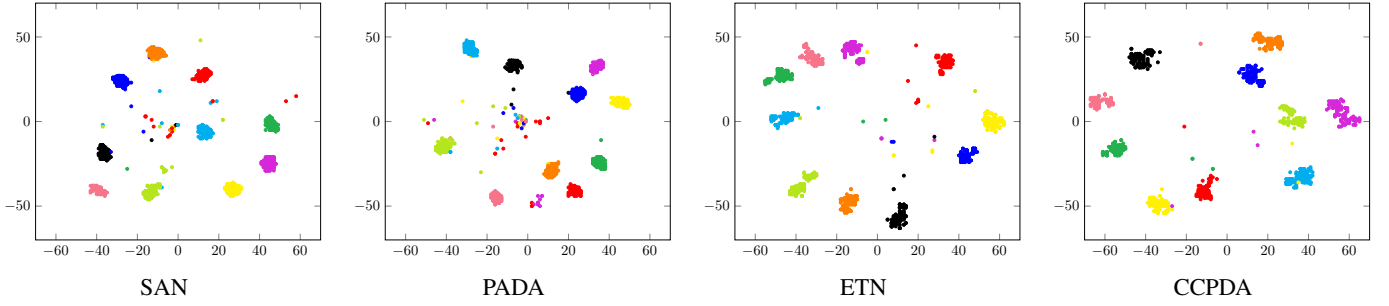


Fig. 4: The t-SNE visualization of SAN [27], PADA [25], ETN [38], and CCPDA on partial domain adaptation task $A \rightarrow W$ with class information (samples are colored w.r.t. their classes). *Best viewed in color.*

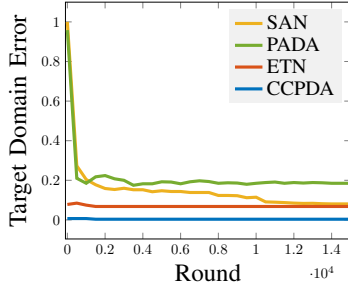


Fig. 5: Empirical analysis of the target domain error through the training process. *Best viewed in color.*

mechanisms to highlight a subset of samples that are more transferable across both domains.

Among the competing partial domain adaptation approaches in Tables I, II, and III, SAN is the only approach that seeks to directly align the conditional distributions of the source and target domains. However unlike CCPDA, SAN uses a different architecture with $|\mathcal{C}_s|$ class-wise domain discriminators to identify the domain-class label of each sample. As reported in Tables I, II and III, CCPDA outperforms SAN with a large margin in all PDA tasks on both Office-31 and Office-Home datasets. Moreover, CCPDA requires fewer parameters compared to SAN. This in turn demonstrates the efficiency and efficacy of the proposed class-conditional model.

The results in Table I indicate that CCPDA outperforms the competing methods on most of the PDA tasks from Office-31 dataset. In particular, CCPDA achieves considerable improvement on $A \rightarrow W$ and $A \rightarrow D$ tasks. It also increases the average accuracy of all tasks by almost 1.36%. Moreover, Table II shows that CCPDA outperforms other PDA approaches with a large margin on five pairs of source-target adaptation tasks: $Ar \rightarrow Pr$, $Ar \rightarrow Rw$, $Cl \rightarrow Ar$, $Cl \rightarrow Pr$, and $Rw \rightarrow Ar$. The results reported in Table III indicate that CCPDA outperform all comparison methods on all the tasks even though the number of the outlier classes ($|\mathcal{C}_s \setminus \mathcal{C}_t|$) is much larger than that of the shared classes ($|\mathcal{C}_t|$). The numerical results provided in Tables I, II, and III corroborate CCPDA can effectively align the class-conditional distribution, mitigate transferring knowledge from the outlier source classes, and promote positive transfer between the domains in the shared

TABLE IV: Classification accuracy of CCPDA and its variants for Partial Domain Adaptation tasks on Office-31 dataset.

| Method | $A \rightarrow W$ | $D \rightarrow W$ | $W \rightarrow D$ | $A \rightarrow D$ | $D \rightarrow A$ | $W \rightarrow A$ | Avg |
|-------------------|-------------------|-------------------|-------------------|-------------------|-------------------|-------------------|--------------|
| PADA | 86.54 | 99.32 | 100.0 | 82.17 | 92.69 | 95.41 | 92.69 |
| CCPDA $_{\infty}$ | 95.12 | 99.32 | 100.0 | 93.21 | 96.03 | 95.19 | 96.48 |
| CCPDA $_e$ | 97.45 | 96.64 | 100.0 | 96.47 | 94.92 | 93.86 | 96.56 |
| CCPDA $_{d,c}$ | 93.42 | 97.62 | 100.0 | 90.43 | 93.45 | 95.53 | 95.07 |
| CCPDA | 99.66 | 100.0 | 100.0 | 97.45 | 95.72 | 95.71 | 98.09 |

label space.

Furthermore, we perform an ablation study to evaluate the efficacy brought by different components of the proposed PDA model. We consider PADA as a baseline variant of CCPDA with binary domain discriminator G_d and without regularization terms \mathcal{L}_c , \mathcal{L}_{∞} , and \mathcal{L}_e . The results are reported in Table IV and they reveal interesting observations. CCPDA $_{d,c}$ outperforms PADA in most of the tasks, which highlights the importance of the incorporated regularization terms \mathcal{L}_{∞} and \mathcal{L}_e in rejecting the outlier source samples. Moreover, we can see that both variants CCPDA $_{\infty}$ and CCPDA $_e$ improved the accuracy of the original baseline, which corroborate the efficacy of our class-conditional domain discriminator \tilde{G}_d . Overall, observe that different components of the proposed method bring complimentary information into the model and they have contribution in achieving the state-of-the-art classification results.

Visualization: To better demonstrate the ability of the proposed method in aligning the feature distributions in the shared label space, we visualize the bottleneck representations learned by SAN, PADA, ETN, and CCPDA on task A (**31 classes**) $\rightarrow W$ (**10 classes**) using t-SNE embedding [49] (Shown in Figure 4). It is desired to embed the source and target sample points of the same class close together while keeping embeddings from different classes far apart. Observe that CCPDA is able to effectively discriminate the classes shared between the domains, while minimizing the distance between the same classes in both domains.

Convergence Performance: To highlight other advantages of our approach, we compare the test error rate obtained by CCPDA against various methods SAN, PADA, and ETN on partial domain adaptation task A (**31 classes**) $\rightarrow W$

(10 classes), from Office dataset. Figure 5 illustrates the convergence behavior of the test errors in 15,000 iterations. Each curve is obtained by averaging over 5 independent runs for the entire test samples. Observe that comparing to the competitor methods, CCPDA not only converges very quickly but also achieves lower error rate.

VI. CONCLUSION

This work presented a novel adversarial architecture for the task of partial domain adaptation. The proposed model adopts a multi-class adversarial loss function to jointly align the marginal and class-conditional distributions across the shared classes between the source and target domains. Furthermore, it leverages two regularization functions to reduce the adverse effects of the outlier classes and the irrelevant samples in transferring information. Several experiments performed on the standard benchmark datasets for partial domain adaptation have demonstrated that our method can outperform the state-of-the-art methods on multiple adaptation tasks in terms of the classification performance.

REFERENCES

- [1] C. Szegedy, A. Toshev, and D. Erhan, "Deep neural networks for object detection," in *NeurIPS*, 2013.
- [2] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," in *NeurIPS*, 2015.
- [3] R. Girshick, "Fast r-cnn," in *ICCV*, 2015.
- [4] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *CVPR*, 2016.
- [5] S. J. Pan, I. W. Tsang, J. T. Kwok, and Q. Yang, "Domain adaptation via transfer component analysis," *IEEE Trans. Neural Netw.*, vol. 22, pp. 199–210, 2011.
- [6] B. Gong, Y. Shi, F. Sha, and K. Grauman, "Geodesic flow kernel for unsupervised domain adaptation," in *CVPR*, 2012.
- [7] M. Baktashmotlagh, M. T. Harandi, B. C. Lovell, and M. Salzmann, "Unsupervised domain adaptation by domain invariant projection," in *ICCV*, 2013.
- [8] J. Yosinski, J. Clune, Y. Bengio, and H. Lipson, "How transferable are features in deep neural networks?" in *NeurIPS*, 2014.
- [9] J. Donahue, Y. Jia, O. Vinyals, J. Hoffman, N. Zhang, E. Tzeng, and T. Darrell, "Decaf: A deep convolutional activation feature for generic visual recognition," in *ICML*, 2014.
- [10] M. Long, Y. Cao, J. Wang, and M. Jordan, "Learning transferable features with deep adaptation networks," in *ICML*, 2015.
- [11] E. Tzeng, J. Hoffman, T. Darrell, and K. Saenko, "Simultaneous deep transfer across domains and tasks," in *ICCV*, 2015.
- [12] W. Deng, L. Zheng, Q. Ye, G. Kang, Y. Yang, and J. Jiao, "Image-image domain adaptation with preserved self-similarity and domain-dissimilarity for person re-identification," in *CVPR*, 2018.
- [13] H. Kazemi, S. Soleymani, F. Taherkhani, S. Iranmanesh, and N. Nasrabadi, "Unsupervised image-to-image translation using domain-specific variational information bound," in *NeurIPS*, 2018.
- [14] M. Wang and W. Deng, "Deep visual domain adaptation: A survey," *Neurocomputing*, vol. 312, pp. 135–153, 2018.
- [15] E. Tzeng, J. Hoffman, N. Zhang, K. Saenko, and T. Darrell, "Deep domain confusion: Maximizing for domain invariance," *arXiv preprint arXiv:1412.3474*, 2014.
- [16] W. Zellinger, T. Grubinger, E. Lughofer, T. Natschlager, and S. Saminger-Platz, "Central moment discrepancy (cmd) for domain-invariant representation learning," in *ICLR*, 2017.
- [17] B. Sun, J. Feng, and K. Saenko, "Return of frustratingly easy domain adaptation," in *AAAI*, 2016.
- [18] B. Sun and K. Saenko, "Deep coral: Correlation alignment for deep domain adaptation," in *ECCV*, 2016.
- [19] Y. Li, N. Wang, J. Shi, J. Liu, and X. Hou, "Revisiting batch normalization for practical domain adaptation," *arXiv preprint arXiv:1603.04779*, 2016.
- [20] F. M. Carlucci, L. Porzi, B. Caputo, E. Ricci, and S. R. Bulò, "Autodial: Automatic domain alignment layers," in *ICCV*, 2017.
- [21] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *NeurIPS*, 2014.
- [22] M. Long, Z. Cao, J. Wang, and M. I. Jordan, "Conditional adversarial domain adaptation," in *NeurIPS*, 2018.
- [23] Z. Pei, Z. Cao, M. Long, and J. Wang, "Multi-adversarial domain adaptation," in *AAAI*, 2018.
- [24] X. Wang, L. Li, W. Ye, M. Long, and J. Wang, "Transferable attention for domain adaptation," in *AAAI*, 2019.
- [25] Z. Cao, L. Ma, M. Long, and J. Wang, "Partial adversarial domain adaptation," in *ECCV*, 2018.
- [26] S. J. Pan and Q. Yang, "A survey on transfer learning," *IEEE T KNOWL DATA EN*, vol. 22, no. 10, pp. 1345–1359, 2010.
- [27] Z. Cao, M. Long, J. Wang, and M. I. Jordan, "Partial transfer learning with selective adversarial networks," in *CVPR*, 2018.
- [28] M. Long, H. Zhu, J. Wang, and M. I. Jordan, "Unsupervised domain adaptation with residual transfer networks," in *NeurIPS*, 2016.
- [29] H. Yan, Y. Ding, P. Li, Q. Wang, Y. Xu, and W. Zuo, "Mind the class weight bias: Weighted maximum mean discrepancy for unsupervised domain adaptation," in *CVPR*, 2017.
- [30] M. Ghifary, W. B. Kleijn, and M. Zhang, "Domain adaptive neural networks for object recognition," in *PRICAI*, 2014.
- [31] B. Sun, J. Feng, and K. Saenko, "Correlation alignment for unsupervised domain adaptation," in *Domain Adaptation in Computer Vision Applications*. Springer, 2017, pp. 153–171.
- [32] Y. Ganin, E. Ustinova, H. Ajakan, P. Germain, H. Larochelle, F. Laviolette, M. Marchand, and V. Lempitsky, "Domain-adversarial training of neural networks," *J MACH LEARN RES*, vol. 17, no. 1, pp. 2096–2030, 2016.
- [33] M. Ghifary, W. B. Kleijn, M. Zhang, D. Balduzzi, and W. Li, "Deep reconstruction-classification networks for unsupervised domain adaptation," in *ECCV*, 2016.
- [34] K. Bousmalis, N. Silberman, D. Dohan, D. Erhan, and D. Krishnan, "Unsupervised pixel-level domain adaptation with generative adversarial networks," in *CVPR*, 2017.
- [35] E. Tzeng, J. Hoffman, K. Saenko, and T. Darrell, "Adversarial discriminative domain adaptation," in *CVPR*, 2017.
- [36] W. Zhang, W. Ouyang, W. Li, and D. Xu, "Collaborative and adversarial network for unsupervised domain adaptation," in *CVPR*, 2018.
- [37] J. Zhang, Z. Ding, W. Li, and P. Ogunbona, "Importance weighted adversarial nets for partial domain adaptation," in *CVPR*, 2018.
- [38] Z. Cao, K. You, M. Long, J. Wang, and Q. Yang, "Learning to transfer examples for partial domain adaptation," in *CVPR*, 2019.
- [39] S. Xie, Z. Zheng, L. Chen, and C. Chen, "Learning semantic representations for unsupervised domain adaptation," in *ICML*, 2018.
- [40] S. Ben-David, J. Blitzer, K. Crammer, and F. Pereira, "Analysis of representations for domain adaptation," in *NeurIPS*, 2007.
- [41] K. Saito, Y. Ushiku, and T. Harada, "Asymmetric tri-training for unsupervised domain adaptation," in *Proceedings of the 34th International Conference on Machine Learning-Volume 70*. JMLR. org, 2017, pp. 2988–2997.
- [42] K. Saenko, B. Kulis, M. Fritz, and T. Darrell, "Adapting visual category models to new domains," in *ECCV*, 2010.
- [43] H. Venkateswara, J. Eusebio, S. Chakraborty, and S. Panchanathan, "Deep hashing network for unsupervised domain adaptation," in *CVPR*, 2017.
- [44] G. Griffin, A. Holub, and P. Perona, "Caltech-256 object category dataset," 2007.
- [45] Y. Ganin and V. Lempitsky, "Unsupervised domain adaptation by backpropagation," in *ICML*, 2015.
- [46] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *CVPR*, 2016.
- [47] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein *et al.*, "Imagenet large scale visual recognition challenge," *IJCV*, vol. 115, pp. 211–252, 2015.
- [48] M. Sugiyama, M. Krauledat, and K.-R. M  zler, "Covariate shift adaptation by importance weighted cross validation," *JMLR*, vol. 8, pp. 985–1005, 2007.
- [49] L. Van Der Maaten, "Barnes-hut-sne," in *ICLR*, 2013.