Progressive Feature Alignment for Unsupervised Domain Adaptation

Chaoqi Chen*[1], Weiping Xie*[1], Wenbing Huang[2], Yu Rong[2], Xinghao Ding[1],
Yue Huang†[1], Tingyang Xu†[2], Junzhou Huang[2]
[1] Fujian Key Laboratory of Sensing and Computing for Smart City,
School of Information Science and Engineering, Xiamen University, China
[2] Tencent AI Lab
cqchen94@stu.xmu.edu.cn, xiewp@stu.xmu.edu.cn, hwenbing@126.com, yu.rong@hotmail.com
dxh@xmu.edu.cn, huangyue05@gmail.com, Tingyangxu@tencent.com, jzhuang@uta.edu

Abstract

Unsupervised domain adaptation (UDA) transfers knowledge from a label-rich source domain to a fully-unlabeled target domain. To tackle this task, recent approaches resort to discriminative domain transfer in virtue of pseudo-labels to enforce the class-level distribution alignment across the source and target domains. These methods, however, are vulnerable to the error accumulation and thus incapable of preserving cross-domain category consistency, as the pseudo-labeling accuracy is not guaranteed explicitly. In this paper, we propose the Progressive Feature Alignment Network (PFAN) to align the discriminative features across domains progressively and effectively, via exploiting the intra-class variation in the target domain. To be specific, we first develop an Easy-to-Hard Transfer Strategy (EHTS) and an Adaptive Prototype Alignment (APA) step to train our model iteratively and alternatively. Moreover, upon observing that a good domain adaptation usually requires a non-saturated source classifier, we consider a simple yet efficient way to retard the convergence speed of the source classification loss by further involving a temperature variate into the soft-max function. The extensive experimental results reveal that the proposed PFAN exceeds the state-of-the-art performance on three UDA datasets.

1. Introduction

Hiving large-scale labeled datasets is one of the reasons for the recent success of deep convolutional neural networks (CNNs) [14]. Nevertheless, the collection and annotation of numerous samples in various domains is an extremely expensive and time-consuming process. Meanwhile, traditional CNNs trained on one large dataset show low generalization ability on another due to the data bias or shift [38]. Unsupervised domain adaptation (UDA) methods tackle the mentioned problem by transferring knowledge from a label-rich source domain to a fully unlabeled target domain [28, 27]. The deep UDA methods have achieved remarkable performance [40, 22, 9, 10, 2, 39, 25, 33, 30, 16], which usually seek to jointly achieve small source generalization error and cross-domain distribution discrepancy.

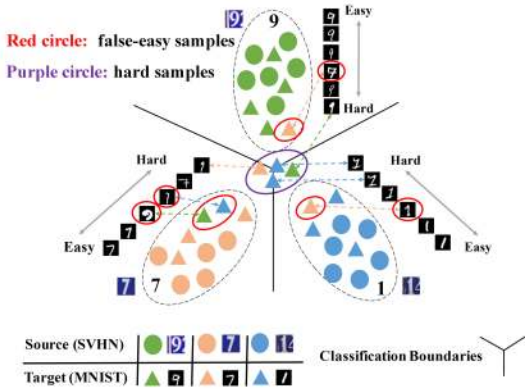


Figure 1: (Best viewed in color.) Motivations of the proposed work (SVHN→MNIST). The classification boundaries are first drawn by the fully labeled source domain. There is intra-class variation in the target domain.

Most prior efforts focus on matching global source and target data distributions to learn domain-invariant representations. However, the learned representations may not only bring the source and target domains closer, but also mix samples with different class labels together. Recent studies [23, 34, 13, 32, 44, 29, 21, 35, 44, 41] started to consider learning discriminative representations for the target domain. Specifically, some of them [34, 32, 44] proposed to use pseudo-labels to learn target discriminative representations, which encourages a low-density separation between classes in the target domain [20]. Despite their efficacy, these approaches faces two critical limitations. Firstly, they

* indicates equal contributions.
† Corresponding authors

require a strong pre-assumption that the correctly-pseudo-labeled samples can reduce the bias caused by falsely-pseudo-labeled samples. Nevertheless, it is challenged to satisfy the assumption, especially when the domain discrepancy is large. The learned classifiers might be incapable of confidently distinguishing target samples, or precisely pseudo-label them with an expected accuracy requirement. Secondly, they backpropagate the category loss for target samples based on pseudo-labeled samples, which makes the target performance vulnerable to the error accumulation.

During the exploration, we empirically observe the distinct data patterns in the target domain. The motivation is demonstrated in Fig. 1. The intra-class distribution variance exists in the target domain. Some target samples, which we call easy samples, are very likely to be classified correctly since they are sufficiently close to the source domain, and we can directly assign pseudo-labels to them without any adaptation. Some target samples, which we call hard samples, lay far away from the source domain and they are ambiguous for the classification boundaries. Moreover, some easy samples, which we call false-easy samples, lay in the support of non-corresponding source classes and are prone to be falsely pseudo-labeled with high confidence. These false-labeled samples introduce wrong information in the category alignment and potentially result in the error accumulation. Thus it is prerequisite to alleviate their negative influences in the context of UDA.

In this paper, we propose a Progressive Feature Alignment Network (PFAN), which largely extends the ability of prior discriminative representations-based approaches by explicitly enforcing the category alignment in a progressive manner. Firstly, an Easy-to-Hard Transfer Strategy (EHTS) progressively selects reliable pseudo-labeled target samples with cross-domain similarity measurements. However, the selected samples may include some misclassified target samples with high confidence. Then, to suppress the negative influence of falsely-labeled samples, we propose an Adaptive Prototype Alignment (APA) to align the source and target prototypes for each category. Rather than backpropagating the category loss for target samples based on pseudo-labeled samples, our work statistically align the cross-domain class distributions based on the source samples and the selected pseudo-labeled target samples.

The EHTS and APA update iteratively and alternatively, where EHTS boosts the robustness of APA by providing reliable pseudo-labeled samples, and the cross-domain category alignment learned by APA can effectively alleviate those falsely-labeled samples introduced by the EHTS. Moreover, upon observing that a good adaptation model usually requires a non-saturated source classifier, we consider a simple yet efficient way to retard the convergence speed of the source classification loss by further involving a temperature variate into the soft-max function. The experimental results reveal that the proposed PFAN exceeds the state-of-the-art performance on three UDA datasets.

2. Related Work

We summarize the work most relevant to our proposed approach. We focus primarily on deep UDA methods due to their empirical superiority in this problem.

Inspired by the recent success of generative adversarial networks (GAN) [11], deep adversarial domain adaptation has received increasing attention in learning domain-invariant representations to reduce the domain discrepancy and provide remarkable results [9, 39, 29, 43, 44, 17, 45]. These methods try to find a feature space such that confusion between the source and the target distributions in that space is maximal. For example, [9] proposed a gradient reversal layer to train a feature extractor that produces features which maximize the domain binary classifier loss, while simultaneously minimizing the label predictor loss.

Many approaches utilize a distance metric to measure the domain discrepancy between the source and target domains, such as maximum mean discrepancy (MMD), KL-divergence or Wasserstein distance [12, 22, 37, 24, 42, 6]. Most prior efforts intend to achieve domain alignment by matching $P(X_s)$ and $P(X_t)$. However, an exact domain-level alignment does not imply a fine-grained class-to-class overlap. Thus, it is important to pursue the category-level alignment under the absence of target true labels.

[3, 5, 23, 34, 32, 44, 41] utilize the pseudo-labels to compensate the lack of categorical information in the target domain. [23] jointly matched both the marginal distribution and conditional distribution using a revised MMD. [32] utilized an asymmetric tri-training strategy to learn discriminative representations for the target domain. [44] iteratively selected pseudo-labeled target samples based on the classifier from the previous training epoch and re-trained the model by using the enlarged training set. [41] proposed to assign pseudo-labels to all target samples and utilize them to achieve semantic alignment across domains. However, these approaches highly relied on the hypothesis that correctly-pseudo-labeled samples can reduce the bias caused by falsely-pseudo-labeled samples. They do not explicitly alleviate those falsely-pseudo-labeled samples. When the falsely-pseudo-labeled samples take the prominent position, their performances will be limited.

3. Progressive Feature Alignment Network

In this section, we first provide the details of the proposed PFAN and then theoretically investigate the effectiveness of our approach. The overall architecture of PFAN is depicted in Fig. 2, which consists of three components, EHTS, APA, and the soft-max function with a temperature variate. EHTS provides reliable pseudo-labeld samples

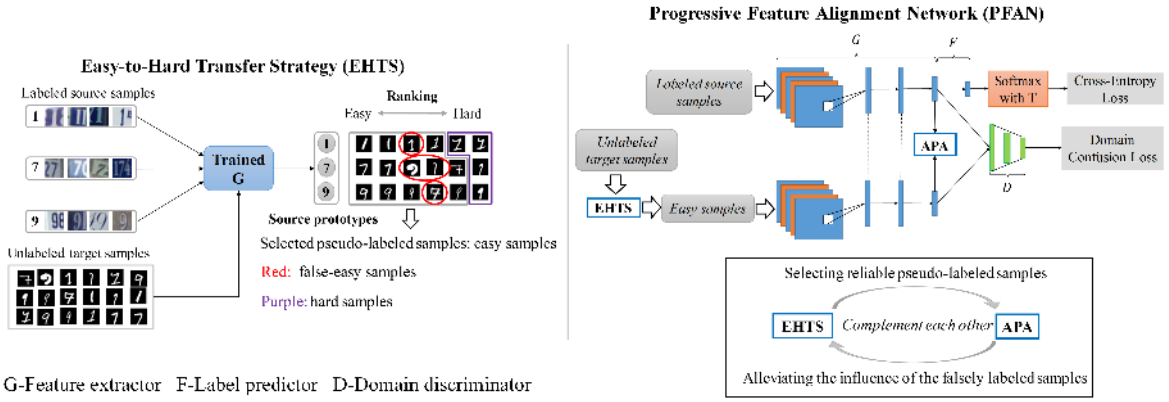


Figure 2: The overall structure of the proposed PFAN. We separate the network into three modules: feature extractor G, label predictor F, domain discriminator D and with associated parameters $\theta_g, \theta_f, \theta_d$. **Left**: The easy-to-hard strategy (ETHS). **Right**: The network structure: the dotted lines in PFAN denote weight-sharing.

from easy to hard by iterations and APA explicitly enforces the cross-domain category alignment.

3.1. Task Formulation

In UDA, we are given a source domain $D_s = \{(x_i^s, y_i^s)\}_{i=1}^{n_s}$ ($x_i^s \in X_s$, $y_i^s \in Y_s$) of n_s labeled samples and given a target domain $D_t = \{x_j^t\}_{j=1}^{n_t}$ ($x_j{}^t \in X_t$) of n_t unlabeled samples [28]. The source and target domains are drawn from the joint probability distributions $P(X_s, Y_s)$ and $Q(X_t, Y_t)$ respectively, and $P \neq Q$. We assume that the source and target domains contain the same object classes, and we consider C classes in all.

3.2. Easy-to-Hard Transfer Strategy

The EHTS is biased to favor easier samples and this bias helps to avoid including the hard samples which are more likely to be given false pseudo-labels. In our approach, the easy samples are increasing progressively. Thus the "hard" samples will potentially be selected in further steps. The selected pseudo-labeled samples by EHTS can be used to align with their corresponding source categories as described in Section 3.3.

The EHTS first computes a D-dimensional prototype $c_k^{\mathcal{S}} \in R^D$ of each class in the source domain. The source prototype is a mean vector of the embedded source samples in each class through an embedding function G (*i.e.* the feature extractor in Fig. 2) with trainable parameters θ_g,

$$c_k^{\mathcal{S}} = \frac{1}{N_s^k} \sum_{(x_i^s, y_i^s) \in D_s^k} G(x_i^s), \tag{1}$$

where D_s^k denotes the set of samples labeled with class k in the source domain and N_s^k is the number of corresponding samples. Then, a set of prototypes $\{c_k^{\mathcal{S}}\}_{k=1}^C$ are obtained. The embedded target samples are supposed to gather around the source prototypes in the latent feature space. Thus, we use a similarity measurement ψ to cluster j-th unlabeled target sample, x_j^t, to the corresponding source prototypes, where ψ is computed as follows,

$$\psi(x_j^t) = CS(G(x_j^t), c_k^{\mathcal{S}}), k = \{1, 2, ..., C\} \tag{2}$$

where $CS(.,.)$ denotes the cosine similarity function between two vectors. x_j^t is added into the target domain of the class $D_t^{k'}$ with a pseudo-label $\hat{y}_j^t = k'$ where $k' = arg\max_k \ \psi_k(x_j^t)$.

Then, the unlabeled target samples D_t are partitioned into C classes (*i.e.* $D_t = \{D_t^k\}_{k=1}^C = \{x_j^t, \hat{y}_j^t\}_{j=1}^{n_t}$) and each sample is scored by its similarity. To obtain the "easy" samples, we constrain that the similarity scores should above a certain threshold τ. During the training process, the values of the similarity ψ increase continuously because the source samples and the target samples become closer to each other in the hidden space as training proceeds. "Hard" samples in the earlier stages may be selected as "easy" in the later stages. However, the constant threshold will turn too much "hard" samples into "easy" samples in each step. To control the growth rate of the "easy" samples, we gradually adjust the threshold step by step as follows,

$$\tau = \frac{1}{1 + e^{-\mu \cdot (m+1)}} - 0.01, \tag{3}$$

where μ is a constant and m ($m = \{0, 1, 2, ...\}$) denotes the training step. Therefore, the sample selection function is formulated as follows,

$$\forall x_j \in D_t^k|_{k=1}^C, w_j = \begin{cases} 1 & \text{if} \quad \psi \geq \tau \\ 0 & \text{if} \quad \psi < \tau \end{cases} \tag{4}$$

where $w_j = 1$ indicates x_j to be selected; otherwise, $w_j = 0$ indicates x_j not to be selected. Finally, we obtain a selected pseudo-labeled target domain $\hat{D}_t = \{x_j^t, \hat{y}_j^t\}_{j=1}^{\hat{n}_t}$, where $\hat{n}_t$ denotes the number of selected samples.

3.3. Adaptive Prototype Alignment

In this section, we introduce the proposed APA, which considers the pairwise semantic similarity across domains

to explicitly alleviate the negative influence of those false-easy samples and enforce the cross-domain category consistency. It can be implemented by aligning the prototype of source and selected target samples for each category. We measure the distance between two prototypes as follows,

$$d(c_k^{\mathcal{S}}, c_k^{\mathcal{T}}) = \left\| c_k^{\mathcal{S}} - c_k^{\mathcal{T}} \right\|^2, \tag{5}$$

where $c_k^{\mathcal{S}}$ and $c_k^{\mathcal{T}}$ represent the source and target prototypes, respectively. We opt for the squared Euclidean distance as the distance measure function. The justification is that the cluster mean yields optimal cluster representatives when a Bregman divergence (*e.g.* squared Euclidean distance and Mahalanobis distance) is used [36]. An optional approach for prototype alignment is to compute and align the local prototypes based on the mini-batch sampled from D_s and $\hat{D}_t$ at each iteration. However, this approach is in a position of weakness because the categorical information in each mini-batch is expected to be insufficient, even one falsely labeled sample in the target mini-batch may cause huge bias between the computed prototype and true prototype.

To overcome the aforementioned problems, we propose to adaptively align the global prototypes. The APA first computes the initial global prototypes based on the selected pseudo-labeled target samples $\hat{D}_t$ as follows,

$$c_{k(0)}^{\mathcal{T}} = \frac{1}{\hat{D}_t^k} \sum_{(x_j^t, y_j^t) \in \hat{D}_t^k} G(x_j^t). \tag{6}$$

In each iteration, we compute a set of local prototypes $\{c_k^t\}_{k=1}^C$ using the mini-batch samples. The accumulated prototypes are computed as the average of all previous local prototypes in each iteration,

$$\bar{c}_{k(I)}^t = \frac{1}{I} \sum_{i=1}^{I} c_{k(i)}^t, \tag{7}$$

where I denotes the iteration times in the current training step. Then, the new c_k^T are updated as follows,

$$\begin{aligned} \rho_t &= CS(\bar{c}_{k(I)}^t, c_{k(I-1)}^{\mathcal{T}}), \\ c_{k(I)}^{\mathcal{T}} &\leftarrow {\rho_t}^2 \bar{c}_{k(I)}^t + (1 - {\rho_t}^2) c_{k(I-1)}^{\mathcal{T}}, \end{aligned} \tag{8}$$

where $CS(.,.)$ is the cosine distance which was defined in Eq. (2) and ρ is the trade-off parameters. let $c_{k(I)}^{\mathcal{S}}$ be analogously updated for the source domain. To this end, the APA loss is formulated as follows,

$$\mathcal{L}_{apa}(\theta_g) = \sum_{k=1}^{C} d(c_{k(I)}^{\mathcal{S}}, c_{k(I)}^{\mathcal{T}}). \tag{9}$$

The motivations of APA is intuitive: 1) the accumulated prototypes are introduced to estimate the accumulated shift caused by the falsely labeled samples, and then we can use their similarity with the previous global prototypes to decide the new global prototypes $c_k^{\mathcal{T}}$; and 2) we statistically align the cross-domain category distributions which can alleviate the error accumulation of the pseudo-labels.

Algorithm 1 Progressive Feature Alignment Network, $m = \{0, 1, ...\}$ denotes the training step, I denotes the iteration times, B_s and B_t denote the mini-batch training sets.

Require: labeled source samples $D_s = \{(x_i^s, y_i^s)\}_{i=1}^{n_s}$, unlabeled target samples $D_t = \{x_j^t\}_{j=1}^{n_t}$
Ensure: $\theta_g, \theta_d, \theta_f$

```
m = 0
Stage-1:
Initialize G and F using D_s, output: model_0
Stage-2:
while not converge do
    m = m + 1
    Run the EHTS based on model_{m-1}, output: D̂_t
    Calculate the initial global prototypes c^S_{k(0)} and c^T_{k(0)}
    using D_s and D̂_t based on model_{m-1}
    for I = 1 to max_iter do
        Derive B_s and B_t sampled from D_s and D̂_t
        Calculate local prototypes c^s_{k(I)} and c^t_{k(I)}
        Update: c^S_{k(I)}, c^T_{k(I)} by using Eq. 7 and Eq. 8
        Train model_m fine-tuned from model_{m-1} using
        B_s and B_t by optimizing 12, output: model_m
    end for
    D̂_t = ∅
end while
```

3.4. Training Losses

In this work, we empirically found that a good adaptor needs a *non-saturated* source classifier. This empirical result is supported by the theoretical analysis described in Section 3.5. The justification is that the adaptation model is biased towards minimizing the source classification loss, which usually converges rapidly since the available of the source true labels. However, this bias may lead the overfitting to the source samples and resulting in a limited target performance. Inspired by [15], we propose to add a high temperature variate T ($T > 1$) to the source classifier (as depicted in Fig. 2). By that means we can retard the convergence speed of the source classification loss and effectively guides the adaptor to a better adaptation performance. We achieve this behavior via the following softmax function,

$$q_i = \frac{exp(z_i/T)}{\sum_j exp(z_j/T)}, \tag{10}$$

where q_i denotes the class probabilities for a source samples and z is the logit that produced by source classifier. Using a higher value for T produces a softer output and naturally retards the convergence speed.

Adversarial learning has been successfully introduced to UDA by extracting domain-invariant features to achieve domain alignment [9]. However, the learned representations can not ensure category alignment, which is the main source of performance reduction. Therefore, our work simultaneously considers domain-level and category-level alignment.

In our PFAN, the input x is first embedded by G to a D-dimensional feature vector $\mathbf{f} \in R^D$, i.e. $\mathbf{f} = G(x; \theta_g)$. In order to make $\mathbf{f}$ domain-invariant, the parameters θ_g of feature extractor G are expected to be optimized by maximizing the loss of the domain discriminator D, while the parameters θ_d of domain discriminator D are trained by minimizing the loss of the domain discriminator, the discriminator is optimized following a standard classification loss:

$$\begin{aligned}\mathcal{L}_d(\theta_g, \theta_d) = E_{x\sim D_s}[logD(G(x))] \\ + E_{x\sim \hat{D}_t}[logD(1 - G(x))],\end{aligned} \tag{11}$$

In addition, we also need to simultaneously minimize the loss of the label predictor F for the labeled source samples and the APA loss. Formally, our ultimate goal is to optimize the following minimax objective:

$$\begin{aligned}\min_{\theta_g,\theta_f} \max_{\theta_d} \sum_{i=1}^{n_s} \mathcal{L}_c(F(G(x_i^s; \theta_g); \theta_f), y_i^s) \\ +\lambda\mathcal{L}_d(\theta_g, \theta_d) + \gamma\mathcal{L}_{apa}(\theta_g)\end{aligned} \tag{12}$$

where $\mathcal{L}_c$ is the standard cross-entropy loss, λ and γ are weights that control the interaction among the source classification loss, the domain confusion loss and the APA loss. The pseudo-code of training PFAN is shown in Algorithm 1, the EHTS and APA work alternatively and iteratively.

3.5. Theoretical Analysis

In this section, we theoretically show that our approach improves the boundary of the expected error on the target samples, making use of the theory of domain adaptation [1]. Formally, let $\mathcal{H}$ be the hypothesis class and given two domains $\mathcal{S}$ and $\mathcal{T}$, the probabilistic bound of the error of hypothesis h on the target domain is defined as,

$$\forall h \in \mathcal{H}, R_{\mathcal{T}}(h) \leq R_{\mathcal{S}}(h) + \frac{1}{2} d_{\mathcal{H}\Delta\mathcal{H}}(\mathcal{S}, \mathcal{T}) + C \tag{13}$$

where the expected error on the target samples, $R_{\mathcal{T}}(h)$, are bounded by three terms: (1) the expected error on the source domain, $R_{\mathcal{S}}(h)$; (2) $d_{\mathcal{H}\Delta\mathcal{H}}(\mathcal{S}, \mathcal{T})$ is the domain divergence measured by a discrepancy distance between two distributions $\mathcal{S}$ and $\mathcal{T}$ *w.r.t.* a hypothesis set $\mathcal{H}$; (3) the shared error of the ideal joint hypothesis, C.

In Inequality (13), $R_{\mathcal{S}}(h)$ is expected to be small and prone to be optimized by a deep network since we have source labels. On the other hand, prior efforts [9] seeks to minimize $d_{\mathcal{H}\Delta\mathcal{H}}(\mathcal{S}, \mathcal{T})$ by the domain classifier-based adversarial learning. However, A small $d_{\mathcal{H}\Delta\mathcal{H}}(\mathcal{S}, \mathcal{T})$ and a small $R_{\mathcal{S}}(h)$ do not guarantee small $R_{\mathcal{T}}(h)$. It is possible that C tends to be large when the cross-domain category alignment is not be explicitly enforced (*i.e.* the marginal distribution is well aligned, but the class conditional distribution is not guaranteed). Therefore, C needs to be bounded as well. Unfortunately, we cannot directly measure C due to the absence of target true labels. Thus, we resort to the pseudo-labels to give the approximate evaluation and minimization.

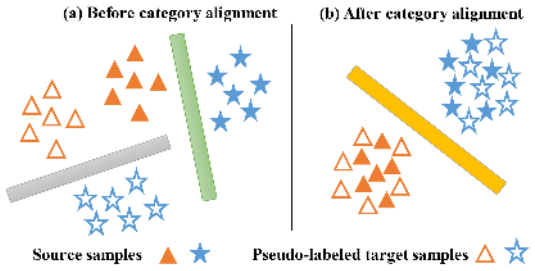


Figure 3: (a) Before category alignment: there exists an optimality gap. (b) After category alignment: the optimality gap does not exist any more.

Definition 1. *If $R_{\mathcal{T}'}(\cdot)$ denotes the expected risk on the selected pseudo-labeled target set $\hat{D}_t$, the ideal joint hypothesis is the hypothesis which minimizes the combined error*

$$h^* = \arg\min_{h\in\mathcal{H}} R_{\mathcal{S}}(h, f_{\mathcal{S}}) + R_{\mathcal{T}'}(h, f_{\mathcal{T}}),$$

and the combined error of the ideal hypothesis is

$$C = R_{\mathcal{S}}(h^*, f_{\mathcal{S}}) + R_{\mathcal{T}'}(h^*, f_{\mathcal{T}}), \tag{14}$$

where $f_{\mathcal{S}}$ and $f_{\mathcal{T}}$ are the labeling functions for the source and target domains, respectively.

To bound the combined error of the ideal hypothesis, the following inequality holds:

Theorem 1. *Let $f_{\hat{\mathcal{T}}}$ be the pseudo-labeling function. Given $R_{\mathcal{T}'}(f_{\mathcal{S}}, f_{\hat{\mathcal{T}}})$ and $R_{\mathcal{T}'}(f_{\mathcal{T}}, f_{\hat{\mathcal{T}}})$ as the minimum shared error and the degree to which the target samples are falsely labeled on $\hat{D}_t$, respectively. We have*

$$C \leq \min_{h\in\mathcal{H}} R_{\mathcal{S}}(h, f_{\mathcal{S}}) + R_{\mathcal{T}'}(h, f_{\hat{\mathcal{T}}}) + 2R_{\mathcal{T}'}(f_{\mathcal{S}}, f_{\hat{\mathcal{T}}}) + R_{\mathcal{T}'}(f_{\mathcal{T}}, f_{\hat{\mathcal{T}}}). \tag{15}$$

We show the derivation of Theorem 1 in the Supplementary Material. It is easy to respectively find a suitable h in $\mathcal{H}$ to approximate the $f_{\mathcal{S}}$ and $f_{\hat{\mathcal{T}}}$ since we have the source labels and target pseudo-labels. However, we assume that when the category alignment has not been achieved, there exists an optimality gap between $f_{\mathcal{S}}$ and $f_{\hat{\mathcal{T}}}$ (Fig. 3(a)). While most existing methods do not consider such phenomenon and directly minimizing $R_{\mathcal{S}}(h, f_{\mathcal{S}})$, which leads the overfitting to source samples.

Remark 1 (Minimizing $R_{\mathcal{S}}(h, f_{\mathcal{S}}) + R_{\mathcal{T}'}(h, f_{\hat{\mathcal{T}}})$)**.** *The proposed softmax function with a temperature variate alleviates the overfitting to source samples (i.e. enforcing a non-saturated source classifier) by retarding the convergence speed of $R_{\mathcal{S}}(h, f_{\mathcal{S}})$. This guides the adaptation model to a better target performance, i.e., a smaller*

$R_{\mathcal{S}}(h, f_{\mathcal{S}}) + R_{\mathcal{T}'}(h, f_{\hat{\mathcal{T}}})$. *Note that when the cross-domain category distributions is well aligned, the aforementioned optimality gap is removed (Fig. 3(b)).*

Recall that the labeling function f can be decomposed into the feature extractor G and label classifier F. By considering the 0-1 loss function σ for $R_{\mathcal{T}'}$, we have

$$\begin{aligned} R_{\mathcal{T}'}(f_{\mathcal{S}}, f_{\hat{\mathcal{T}}}) &= E_{x\sim\mathcal{T}'}[\sigma(F_{\mathcal{S}}(G(x)), F_{\hat{\mathcal{T}}}(G(x)))] \\ &= E_{x\sim\mathcal{T}'}[|\sigma(F_{\mathcal{S}}(G(x)), y_1) - \sigma(F_{\hat{\mathcal{T}}}(G(x)), y_2)|] \end{aligned} \tag{16}$$

where

$$|\sigma(F_{\mathcal{S}}(G(x)), y_1) - \sigma(F_{\hat{\mathcal{T}}}(G(x)), y_2)| = \begin{cases} 1 & \text{if} \quad y_1 \neq y_2 \\ 0 & \text{if} \quad y_1 = y_2 \end{cases} \tag{17}$$

Remark 2 (Minimizing shared error)**.** *The proposed approach aims to progressively align feature in category-level, i.e., it aligns the kth class in source domain D_s^k with the same pseudo-labeled target class $\hat{D}_t^k$. When the categories are aligned, it is safe to assume that $y_1 = y_2$. Thus, $R_{\mathcal{T}'}(f_{\mathcal{S}}, f_{\hat{\mathcal{T}}})$ is expected to be minimized.*

Remark 3 (Minimizing the degree to which the target samples are falsely labeled on $\hat{D}_t$)**.** *The proposed EHTS aims to select reliable pseudo-labeled samples in the target domain which minimizes $R_{\mathcal{T}'}(f_{\mathcal{T}}, f_{\hat{\mathcal{T}}})$.*

4. Experiments

4.1. Datasets and Baselines

Office-31 [31] is a popular benchmark for evaluation on domain adaptation. It contains 4110 images of 31 categories in total, which are collected from three domains, including Amazon (**A**) comprising 2817 images downloaded from online merchants, Webcam (**W**) involving 795 low resolution images acquired from webcams, and DSLR (**D**) containing 498 high resolution images of digital SLRs. We try all 6 combinations of two domains for evaluation.

ImageCLEF-DA [4] originally used for the ImageCLEF 2014 domain adaptation challenge consists of twelve common classes from three domains: ImageNet ILSVRC 2012 (**I**), Pascal VOC 2012 (**P**), and Caltech-256 (**C**). Each doamin has 600 images in total and contains 50 images per class. We test 6 tasks by using all domain combinations.

MNIST [19], **SVHN** [26] and **USPS** [7] contain digital images of 10 classes. In particular, the images in MNIST and SVHN are grey, and are of size 28×28 and 16×16, respectively; USPS consists of color images of size 32×32, and there are often more than one digit in one image. Following previous works, we consider the three transfer tasks: MNIST→SVHN, SVHN→MNIST and MNIST→USPS.

4.2. Implementation Details

Joining previous practices, we instantiate our backbone by AlexNet that has been pre-trained on ImageNet for Office-31 and ImageCLEF-DA, and employ the CNN architecture by [39] for the digital datasets. As suggested by [25], we fine-tune the feature extractor G upon the backbone and train the predictor F from the scratch via back propagation. We utlize stochastic gradient descent (SGD) for the training with a momentum of 0.9 and a annealing learning rate (lr) given by $lr_p = \frac{lr_0}{(1+\alpha p)^\beta}$, where p is increased linearly from 0 to 1 as the training proceeds, $lr_0 = 0.01$, $\alpha = 10$, and $\beta = 0.75$. In order to suppress noisy signal especially for the initial training steps, we use the similar schedule method as [9] to adaptively change the values of λ and γ in Eq. (12) by computing $\lambda = \gamma = \frac{2}{1+exp(-\delta p)} - 1$ with $\delta = 10$. We set $T = 1.8$ in Eq. (10) and $\mu = 0.8$ in Eq. (3) for all experiments. The batch size is selected as 128. The means and standard derivations of all results are obtained over 5 random runs. All experiments are implemented by the Caffe framework.

4.3. Comparisons with State-of-the-Arts

State-of-the-arts. We compare our approach with various state-of-the-art UDA methods, including **AlexNet** [18], Deep Domain Confusion (**DDC**) [40], Deep Adaptation Network (**DAN**) [22], Residual Transfer Network (**RTN**) [24] , Reverse Gradient (**RevGrad**) [9], Adversarial Discriminative Domain Adaptation (**ADDA**) [39], Joint Adaptation Networks (**JAN**) [25], Asymmetric Tri-Training (**ATT**) [32] , Multi-Adversarial Domain Adaptation (**MADA**) [29], and Moving Semantic Transfer Network (**MSTN**) [41]. For all above methods, we summarize the results reported in their original papers. For similarity, we term our method as PFAN hereafter.

Table 1 displays the results on Office-31. The proposed PFAN outperforms all compared methods in general and improves the state-of-the-art result from 79.1% to 80.4% on average. If we focus more on the hard transfer tasks (*e.g.* **A** → **W** and **A** → **D**), PFAN substantially exhibits better transferring ability than others. In contrast to JAN, MADA and MSTN, our PFAN additionally considers both the target intra-class variation and the non-saturated source classifier. Our better performance over them could indicate the effectiveness of these two components. RevGrad has also taken the domain adversarial adaptation into account, but its results are still inferior to ours. The advantage of our model compared to RevGrad is that, we furhter perform EHTS and APA, which as supported by our experiments can explicitly enforce the cross-domain category alignment, hence delivering better performance.

The results of ImageCLEF-DA are reported in Table 2. Our approach outperforms all comparison methods on most transfer tasks, which reveals that PFAN is scalable for different datasets.

The results of digit classification are reported in Table 3. We follow the training protocol established in [39]. For adaptation between MNIST and USPS, we randomly sample 2000 images from MNIST and 1800 from USPS. For

Table 1: AlexNet-based approaches on Office-31 (%)

Method	A → W	D → W	W → D	A → D	D → A	W → A	Avg
AlexNet [18]	61.5±0.5	95.1±0.3	99.0±0.2	64.4±0.5	48.8±0.3	47.0±0.4	69.3
DDC [40]	61.8±0.4	95.0±0.5	98.5±0.4	64.4±0.3	52.1±0.6	52.2±0.4	70.6
DAN [22]	68.5±0.4	96.0±0.3	99.0±0.2	67.0±0.4	54.0±0.4	53.1±0.3	72.9
RTN [24]	73.3±0.3	96.8±0.2	99.6±0.1	71.0±0.2	50.5±0.3	51.0±0.1	73.7
RevGrad [9]	73.0±0.5	96.4±0.3	99.2±0.3	72.3±0.3	53.4±0.4	51.2±0.5	74.3
JAN [25]	74.9±0.3	96.6±0.2	99.5±0.2	71.8±0.2	58.3±0.3	55.0±0.4	76.0
MADA [29]	78.5±0.2	**99.8**±0.1	**100.0**±.0	74.1±0.1	56.0±0.2	54.5±0.3	77.1
MSTN [41]	80.5±0.4	96.9±0.1	99.9±0.1	74.5±0.4	62.5±0.4	60.0±0.6	79.1
PFAN	**83.0**±0.3	99.0±0.2	99.9±0.1	**76.3**±0.3	**63.3**±0.3	**60.8**±0.5	**80.4**

Table 2: AlexNet-based approaches on ImageCLEF-DA (%)

Method	I → P	P → I	I → C	C → I	C → P	P → C	Avg
AlexNet [18]	66.2±0.2	70.0±0.2	84.3±0.2	71.3±0.4	59.3±0.5	84.5±0.3	73.9
DAN [22]	67.3±0.2	80.5±0.3	87.7±0.3	76.0±0.3	61.6±0.3	88.4±0.2	76.9
RevGrad [9]	66.5±0.5	81.8±0.4	89.0±0.5	79.8±0.5	63.5±0.4	88.7±0.4	78.2
JAN [25]	67.2±0.5	82.8±0.4	91.3±0.5	80.0±0.5	63.5±0.4	91.0±0.4	79.3
MADA [29]	68.3±0.3	83.0±0.1	91.0±0.2	80.7±0.2	63.8±0.2	**92.2**±0.3	79.8
MSTN [41]	67.3±0.3	82.8±0.2	91.5±0.1	81.7±0.3	65.3±0.2	91.2±0.2	80.0
PFAN	**68.5**±0.5	**84.4**±0.4	**92.2**±0.6	**82.3**±0.4	**66.3**±0.3	91.7±0.2	**80.9**

Table 3: Accuracy on the digit classification task. (%)

Source Target	MNIST SVHN	SVHN MNIST	MNIST USPS
Source Only	33.0±1.2	60.1±1.1	75.2±1.6
RevGrad [9]	35.7	73.9	77.1±1.8
ADDA [39]	-	76.0±1.8	89.4±0.2
ATT [32]	52.8	85.0	-
MSTN [41]	did not converge	91.7±1.5	92.9±1.1
PFAN	**57.6**±1.8	**93.9**±0.8	**95.0**±1.3

Table 4: Ablation of PFAN on different transfer tasks. (%)

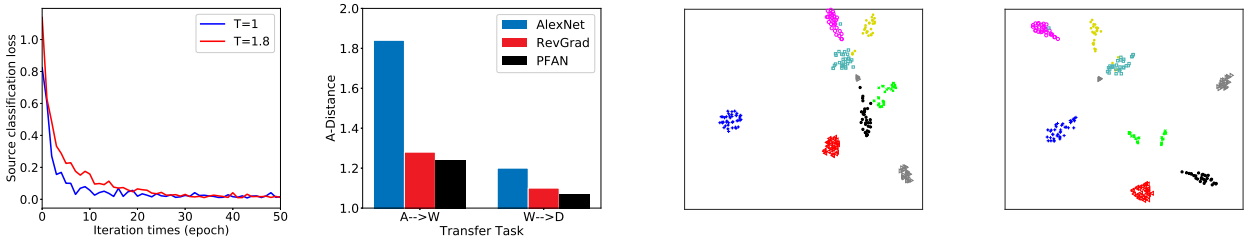
Model	A→W	I→P	SVHN→MNIST
Source Only	61.6	66.2	60.1
PFAN (Random)	77.0	67.0	87.2
PFAN (Full)	81.9	68.0	92.5
PFAN (woAPA)	76.4	67.1	82.0
PFAN (woA)	82.2	68.1	93.0
PFAN (woT)	80.6	67.9	92.1
PFAN	**83.0**	**68.5**	**93.9**

adaptation between SVHN and MNIST, we use the full training sets. For the hard transfer task MNIST→SVHN, we reproduced the MSTN [41] but were unable to get it to converge, since the performance of this approach depends strongly on the accuracy of the pseudo-labeled samples which was lower on this task. In contrast, our approach significantly outperforms the suboptimal result by +4.8%, which clearly demonstrates the effect of our approach on selecting reliable pseudo-labeled samples and alleviating the negative influence of falsely-labeled samples on the challenging scenario. For the easier tasks SVHN→MNIST and MNIST→USPS, our approach also shows superiority.

4.4. Further Empirical Analysis

Ablation Study. To isolate the contribution of our work, we perform ablation study by evaluating several variants of PFAN: (1) **PFAN (Random)**, which randomly selects the target samples instead of using the easy-to-hard order; (2) **PFAN (Full)**, which uses all target samples at the training period; (3) **PFAN (woAPA)**, which denotes training completely without the APA (i.e. $\gamma = 0$ in Eq. (12)); (4) **PFAN (woA)**, which denotes aligning the prototypes based on the current mini-batch without considering the global and accumulated prototypes; (5) **PFAN (woT)**, which removes the temperature from our model (i.e. $T = 1$ in Eq. (10)). The results are shown in Table 4. We can observe that all the components are designed reasonably and when any one of these components is removed, the performance degrades. It is noteworthy that PFAN outperforms both PFAN (Random) and PFAN (Full), which reveals that the EHTS can provide more reliable and informative target samples for the cross-domain category alignment.

Pseudo-labeling Accuracy. We show the relationship between the pseudo-labeling accuracy and test accuracy in Fig. 5. We found that (1) the pseudo-labeling accuracy keeps higher and stable throughout as training proceeds, which thanks to the EHTS by selecting reliable pseudo-labeled samples; (2) the test accuracy increases with the increasing of labeled samples, which implies that the number of correctly and falsely labeled samples are both proportionally increasing, but our approach can explicitly alleviate the negative influence of the falsely-labeled samples.



(a) Non-saturated source classifier (b) Distribution Discrepancy (c) RevGrad: target=W (d) PFAN: target=W

Figure 4: (a) The convergence speed of the source classification loss in different temperature setting. (b) Distribution Discrepancy. (c)-(d) The t-SNE visualization of network activations on target domain **W** generated by RevGrad and PFAN.

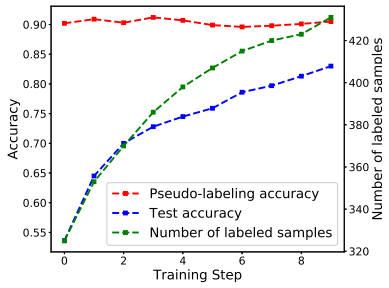


Figure 5: Comparison of the pseudo-labeling accuracy and the test accuracy on transfer task A $\rightarrow$ W. The pseudo-labeling accuracy is computed using (the number of correctly labeled samples)/(the number of labeled samples).

Non-saturated source classifier. To further verify our hypothesis about the non-saturated source classifier, we investigate the source classification loss in different temperature setting. The results are reported in Fig. 4(a). The $T = 1$ model converges faster than $T = 1.8$ especially at the beginning of training. However, such difference gradually decreases as training proceeds. The justification is that we use a higher T to retard the convergence speed of the source classification loss (*i.e.* alleviating the adaptor overfitting to the source samples), thus showing better adaptation.

Distribution Discrepancy. The domain adaptation theory [1] suggests that $\mathcal{A}$-distance can be used as a measure of domain discrepancy. The way of estimating empirical $\mathcal{A}$-distance was defined as $d_{\mathcal{A}} = 2(1 - \epsilon)$, where ϵ is the generalization error of a classifier trained to discriminate the source and target features. We utilize a kernel SVM to estimate the $\mathcal{A}$-distance. Fig. 4(b) demonstrates the $\mathcal{A}$-distance calculated with the features from AlexNet, RevGrad and PFAN on tasks **A** $\rightarrow$ **W** and **W** $\rightarrow$ **D**. We can observe that our method significantly reduces the $\mathcal{A}$-distance compared with the AlexNet. However, when compared with RevGrad, PFAN shows smaller improvement with respect to $\mathcal{A}$-distance, but improves the performance by large margin, which demonstrates that a low domain divergence does not imply better performance in the target domain. This phenomenon is consistent with the analysis in Section 3.5.

Feature Visualization. We utilize t-SNE [8] to visualize the deep feature of the network activations on task **A** $\rightarrow$ **W** (randomly selected 8 classes) learned by RevGrad (the bottleneck layer) and PFAN (the bottleneck layer). As shown in Fig. 4(c)-4(d), we can see that the RevGrad features on target domain can not be discriminated very well, some categories have been mixed up in the feature space. By contrast, PFAN can learn more discriminative representations, which jointly enlarges the inter-class dispersion and reduces the intra-class variations.

5. Conclusion

In this paper, we proposed a novel approach called Progressive Feature Alignment Network, to take advantage of target domain intra-class variance and cross-domain category consistency for addressing UDA problems. The proposed EHTS and APA complement each other in selecting reliable pseudo-labeled samples and alleviating the bias caused by the falsely-labeled samples. The performance is further improved by retarding the convergence speed of the source classification loss. The extensive experiments reveal that our approach outperforms state-of-the-art UDA approaches on three domain adaptation datasets.

6. Acknowledgements

This work was supported in part by the National Natural Science Foundation of China under Grants 61571382, 81671766, 61571005, 81671674, 61671309 and U1605252, in part by the Fundamental Research Funds for the Central Universities under Grants 20720160075 and 20720180059, in part by the CCF-Tencent open fund, and the Natural Science Foundation of Fujian Province of China (No.2017J01126).

References

[1] Shai Ben-David, John Blitzer, Koby Crammer, Alex Kulesza, Fernando Pereira, and Jennifer Wortman Vaughan.

A theory of learning from different domains. *Machine learning*, 79(1-2):151–175, 2010.

[2] Konstantinos Bousmalis, George Trigeorgis, Nathan Silberman, Dilip Krishnan, and Dumitru Erhan. Domain separation networks. In *Advances in Neural Information Processing Systems*, pages 343–351, 2016.

[3] Lorenzo Bruzzone and Mattia Marconcini. Domain adaptation problems: A dasvm classification technique and a circular validation strategy. *IEEE transactions on pattern analysis and machine intelligence*, 32(5):770–787, 2010.

[4] Barbara Caputo, Henning Müller, Jesus Martinez-Gomez, Mauricio Villegas, Burak Acar, Novi Patricia, Neda Marvasti, Suzan Üsküdarlı, Roberto Paredes, Miguel Cazorla, et al. Imageclef 2014: Overview and analysis of the results. In *International Conference of the Cross-Language Evaluation Forum for European Languages*, pages 192–211. Springer, 2014.

[5] Minmin Chen, Kilian Q Weinberger, and John Blitzer. Co-training for domain adaptation. In *Advances in neural information processing systems*, pages 2456–2464, 2011.

[6] Qingchao Chen, Yang Liu, Zhaowen Wang, Ian Wassell, and Kevin Chetty. Re-weighted adversarial adaptation network for unsupervised domain adaptation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7976–7985, 2018.

[7] John S Denker, WR Gardner, Hans Peter Graf, Donnie Henderson, Richard E Howard, W Hubbard, Lawrence D Jackel, Henry S Baird, and Isabelle Guyon. Neural network recognizer for hand-written zip code digits. In *Advances in neural information processing systems*, pages 323–331, 1989.

[8] Jeff Donahue, Yangqing Jia, Oriol Vinyals, Judy Hoffman, Ning Zhang, Eric Tzeng, and Trevor Darrell. Decaf: A deep convolutional activation feature for generic visual recognition. In *International conference on machine learning*, pages 647–655, 2014.

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

[10] Mingming Gong, Kun Zhang, Tongliang Liu, Dacheng Tao, Clark Glymour, and Bernhard Schölkopf. Domain adaptation with conditional transferable components. In *International conference on machine learning*, pages 2839–2848, 2016.

[11] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in neural information processing systems*, pages 2672–2680, 2014.

[12] Arthur Gretton, Karsten M Borgwardt, Malte J Rasch, Bernhard Schölkopf, and Alexander Smola. A kernel two-sample test. *Journal of Machine Learning Research*, 13(Mar):723–773, 2012.

[13] Philip Haeusser, Thomas Frerix, Alexander Mordvintsev, and Daniel Cremers. Associative domain adaptation. In *International Conference on Computer Vision (ICCV)*, volume 2, page 6, 2017.

[14] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[15] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015.

[16] Judy Hoffman, Eric Tzeng, Taesung Park, Jun-Yan Zhu, Phillip Isola, Kate Saenko, Alexei A Efros, and Trevor Darrell. Cycada: Cycle-consistent adversarial domain adaptation. In *International Conference on Machine Learning (ICML)*, 2018.

[17] Guoliang Kang, Liang Zheng, Yan Yan, and Yi Yang. Deep adversarial attention alignment for unsupervised domain adaptation: the benefit of target expectation maximization. In *European Conference on Computer Vision*, 2018.

[18] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012.

[19] Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

[20] Dong-Hyun Lee. Pseudo-label: The simple and efficient semi-supervised learning method for deep neural networks. In *Workshop on Challenges in Representation Learning, ICML*, volume 3, page 2, 2013.

[21] Shuang Li, Shiji Song, Gao Huang, Zhengming Ding, and Cheng Wu. Domain invariant and class discriminative feature learning for visual domain adaptation. *IEEE Transactions on Image Processing*, 27(9):4260–4273, 2018.

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

[23] Mingsheng Long, Jianmin Wang, Guiguang Ding, Jiaguang Sun, and Philip S Yu. Transfer feature learning with joint distribution adaptation. In *Proceedings of the IEEE international conference on computer vision*, pages 2200–2207, 2013.

[24] Mingsheng Long, Han Zhu, Jianmin Wang, and Michael I Jordan. Unsupervised domain adaptation with residual transfer networks. In *Advances in Neural Information Processing Systems*, pages 136–144, 2016.

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

[26] Yuval Netzer, Tao Wang, Adam Coates, Alessandro Bissacco, Bo Wu, and Andrew Y Ng. Reading digits in natural images with unsupervised feature learning. In *NIPS workshop on deep learning and unsupervised feature learning*, volume 2011, page 5, 2011.

[27] Sinno Jialin Pan, Ivor W Tsang, James T Kwok, and Qiang Yang. Domain adaptation via transfer component analysis. *IEEE Transactions on Neural Networks*, 22(2):199–210, 2011.

[28] Sinno Jialin Pan and Qiang Yang. A survey on transfer learning. *IEEE Transactions on knowledge and data engineering*, 22(10):1345–1359, 2010.

[29] Zhongyi Pei, Zhangjie Cao, Mingsheng Long, and Jianmin Wang. Multi-adversarial domain adaptation. In *AAAI Conference on Artificial Intelligence*, 2018.

[30] Pedro O Pinheiro and AI Element. Unsupervised domain adaptation with similarity learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8004–8013, 2018.

[31] Kate Saenko, Brian Kulis, Mario Fritz, and Trevor Darrell. Adapting visual category models to new domains. In *European conference on computer vision*, pages 213–226. Springer, 2010.

[32] Kuniaki Saito, Yoshitaka Ushiku, and Tatsuya Harada. Asymmetric tri-training for unsupervised domain adaptation. In *International Conference on Machine Learning*, pages 2988–2997, 2017.

[33] Kuniaki Saito, Kohei Watanabe, Yoshitaka Ushiku, and Tatsuya Harada. Maximum classifier discrepancy for unsupervised domain adaptation. *arXiv preprint arXiv:1712.02560*, 2017.

[34] Ozan Sener, Hyun Oh Song, Ashutosh Saxena, and Silvio Savarese. Learning transferrable representations for unsupervised domain adaptation. In *Advances in Neural Information Processing Systems*, pages 2110–2118, 2016.

[35] Rui Shu, Hung H Bui, Hirokazu Narui, and Stefano Ermon. A dirt-t approach to unsupervised domain adaptation. *arXiv preprint arXiv:1802.08735*, 2018.

[36] Jake Snell, Kevin Swersky, and Richard Zemel. Prototypical networks for few-shot learning. In *Advances in Neural Information Processing Systems*, pages 4077–4087, 2017.

[37] Baochen Sun and Kate Saenko. Deep coral: Correlation alignment for deep domain adaptation. In *European Conference on Computer Vision*, pages 443–450. Springer, 2016.

[38] Antonio Torralba and Alexei A Efros. Unbiased look at dataset bias. In *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, pages 1521–1528. IEEE, 2011.

[39] Eric Tzeng, Judy Hoffman, Kate Saenko, and Trevor Darrell. Adversarial discriminative domain adaptation. In *Computer Vision and Pattern Recognition (CVPR)*, volume 1, page 4, 2017.

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

[41] Shaoan Xie, Zibin Zheng, Liang Chen, and Chuan Chen. Learning semantic representations for unsupervised domain adaptation. In *International Conference on Machine Learning*, pages 5419–5428, 2018.

[42] Hongliang Yan, Yukang Ding, Peihua Li, Qilong Wang, Yong Xu, and Wangmeng Zuo. Mind the class weight bias: Weighted maximum mean discrepancy for unsupervised domain adaptation. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, volume 3, 2017.

[43] Jing Zhang, Zewei Ding, Wanqing Li, and Philip Ogunbona. Importance weighted adversarial nets for partial domain adaptation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8156–8164, 2018.

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

[45] Yang Zou, Zhiding Yu, BVK Vijaya Kumar, and Jinsong Wang. Unsupervised domain adaptation for semantic segmentation via class-balanced self-training. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 289–305, 2018.