Contrastive Learning Assisted-Alignment for Partial Domain Adaptation

Cuie Yang, Yiu-Ming Cheung, *Fellow, IEEE*, Jinliang Ding, *Senior Member, IEEE*, Kay Chen Tan, *Fellow, IEEE*, Bing Xue, *Senior Member, IEEE*, and Mengjie Zhang, *Fellow, IEEE*

***Abstract*— This work addresses unsupervised partial domain adaptation (PDA), in which classes in the target domain are a subset of the source domain. The key challenges of PDA are how to leverage source samples in the shared classes to promote positive transfer and filter out the irrelevant source samples to mitigate negative transfer. Existing PDA methods based on adversarial DA do not consider the loss of class discriminative representation. To this end, this article proposes a contrastive learning-assisted alignment (CLA) approach for PDA to jointly align distributions across domains for better adaptation and to reweight source instances to reduce the contribution of outlier instances. A contrastive learning-assisted conditional alignment (CLCA) strategy is presented for distribution alignment. CLCA first exploits contrastive losses to discover the class discriminative information in both domains. It then employs a contrastive loss to match the clusters across the two domains based on adversarial domain learning. In this respect, CLCA attempts to reduce the domain discrepancy by matching the class-conditional and marginal distributions. Moreover, a new reweighting scheme is developed to improve the quality of weights estimation, which explores information from both the source and the target domains. Empirical results on several benchmark datasets demonstrate that the proposed CLA outperforms the existing state-of-the-art PDA methods.**

***Index Terms*— Class-conditional alignment, contrastive learning, discriminative learning, partial domain adaptation (PDA), transfer learning.**

Manuscript received 6 April 2021; revised 9 November 2021; accepted 12 January 2022. Date of publication 7 February 2022; date of current version 6 October 2023. This work was supported in part by the National Natural Science Foundation of China (NSFC) under Grant 61672444, Grant 61988101, Grant 61876162, and Grant 62161160338; in part by the NSFC/Research Grants Council (RGC) Joint Research Scheme under Grant N_HKBU214/21; in part by the General Research Fund of RGC under Grant 12201321; in part by Hong Kong Baptist University (HKBU) under Grant RC-FNRA-IG/18-19/SCI/03 and Grant RC-IRCMs/18-19/SCI/01; in part by the Innovation and Technology Fund of Innovation and Technology Commission of the Government, Hong Kong, under Project ITS/339/18; in part by the National Key Research and Development Program of China under Grant 2018YFB1701104; and in part by the Science and Technology Program of Liaoning Province under Grant 2020JH2/10500001 and Grant 2020JH1/10100008. *(Corresponding authors: Yiu-Ming Cheung; Jinliang Ding.)*

Cuie Yang is with the Department of Computer Science, Hong Kong Baptist University, Hong Kong, also with the State Key Laboratory of Synthetical Automation for Process Industries, Northeastern University, Shenyang 110819, China, and also with the School of Engineering and Computer Science, Victoria University of Wellington, Wellington 6140, New Zealand (e-mail: cuieyang@outlook.com; cuieyang@comp.hkbu.edu.hk).

Yiu-Ming Cheung is with the Department of Computer Science, Hong Kong Baptist University, Hong Kong (e-mail: ymc@comp.hkbu.edu.hk).

Jinliang Ding is with the State Key Laboratory of Synthetical Automation for Process Industries, Northeastern University, Shenyang 110819, China (e-mail: jlding@mail.neu.edu.cn).

Kay Chen Tan is with the Department of Computing, The Hong Kong Polytechnic University, Hong Kong (e-mail: kctan@polyu.edu.hk).

Bing Xue and Mengjie Zhang are with the School of Engineering and Computer Science, Victoria University of Wellington, Wellington 6140, New Zealand (e-mail: bing.xue@ecs.vuw.ac.nz; mengjie.zhang@ecs.vuw.ac.nz).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TNNLS.2022.3145034.

Digital Object Identifier 10.1109/TNNLS.2022.3145034

I. Introduction

SUPERVISED learning has achieved impressive performance in numerous practical applications, such as image classification and object detection [1]–[3]. However, the success of many state-of-the-art methods often relies on the scenario that a huge amount of labeled training data is available. From the practice perspective, acquiring data labels is expensive, time-consuming, or even unrealistic in many real-world applications. To relax the need for abundant labeled data, domain adaptation (DA) has emerged as an alternative approach to leverage the useful knowledge of a related labeled source domain to the interested target domain. In general, the source and the target domains are not identical and exist a discrepancy, which is a fundamental problem in DA [4], [5].

Many DA methods have been developed to combat the domain discrepancy by minimizing the distribution of the two domains [6], such as marginal distribution [7], conditional distribution [8], [9], and joint distribution [10], [11]. To this end, one type of approach aims to match different statistic moments, e.g., maximum mean discrepancy (MMD) [5] and correlation alignment (CORAL) [12] by mapping features to a new space. The other kind of approach employs generative adversarial networks (GANs) [13] to generate domain confusion features. With the learned domain-invariant features, a classifier trained on the source domain is hopefully to generalize well to the target domain under the hypothesis that two domains have a shared label space [14]. However, finding a source domain that has an identical label space to the target domain is often difficult in real-world applications. A more general scenario is that classes in target domains are a subset of that in source domains, referred to as partial domain adaptation (PDA) [15].

PDA challenges standard DA methods because of the outlier classes in the source domain. Standard DA methods

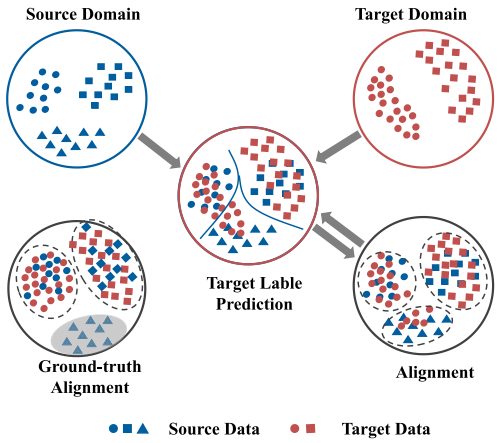


Fig. 1. Source domain involves three classes and the target domain has two classes. The different structures in different domains of the same class cause inaccurate classifier and error distribution alignment literately propagation.

aligning the whole distributions between domains are likely to force outlier classes to match the target domain. Thus, the trained source classifier would misclassify much of the target data, potentially triggering negative transfer. Several PDA approaches have been developed to overcome this issue via jointly filtering out irrelevant source data and matching distributions of shared classes across the two domains [15]–[18]. Most of them remove source-only instances by decreasing their weights in order to down-weight their importance on the domain alignment. Existing PDA methods often estimate weights by the probabilities of predicted target labels [16], the prediction of domain discriminators [15], [17], or the reconstruction error of source data [18], and do not use the information possessed in the target domain.

In classification problems, different classes generally preserve different characteristics, and structures of the same class in different domains are not always similar. Meanwhile, some class discriminability representation may lose in adversarial DA because the domain discriminator is dominated by the eigenvectors with the largest singular [19]. Subsequently, existing methods that align distributions across domains while neglecting discriminability representations would result in class distribution mismatch. Furthermore, the predicted target label distribution by using the classifier trained in the source domain is inaccurate if the class-conditional alignment mismatches. Under this circumstance, the source-only classes will get higher weights if only utilize source classifier in the reweighting scheme. In this respect, the outlier classes will be forced to align with the target data, in turn, thus increasing the wrong distribution alignment. For instance, as shown in Fig. 1, the source domain contains three classes, and the target domain involves two classes. In the ground-truth alignment, the source triangle class is the outlier and should be removed. In contrast, the dot and rectangle classes across the two domains should be matched, respectively. However, due to the complex distribution of dot class in the target domain, some dot samples are easily labeled to the triangle by the source classifier, and thus, progressively aligned with the triangle class. Therefore, previous methods that ignore class discriminative representation and ignore target information in the reweighting scheme are vulnerable, especially in challenging cases.

To alleviate the above-mentioned issues, this article presents a contrastive learning-assisted alignment (CLA) model incorporating a contrastive learning-assisted class-conditional alignment (CLCA) and a new reweighting scheme. Similar to the existing PDA methods, CLA aims to address fundamental problems in PDA, e.g., filtering out source-only instances and reducing distribution mismatch of the shared classes. On the one hand, CLA exploits CLCA to achieve a discriminative class-conditional matching between the two domains based on a marginal distribution alignment to perform a joint alignment. Contrastive learning is a learning technique that extracts general representations of a dataset without labels [20]. Thus, CLCA first exploits contrastive learning to preserve discriminative representations from both domains. Then, it formulates the contrastive representations in the target domain into discriminative clusters, where each cluster represents a class. After that, CLCA deploys a supervised constructive loss to match the classes associated with the same label in both domains. The reweighting scheme explores both the source and target information to improve the quality of recognizing relevant source data. Specifically, it combines the predictions from the target cluster and the source classifier together to estimate the weight of source samples.

To sum up, our contributions are threefold.

1) We propose a new strategy to enhance the discriminative representation of the target domain via contrastive learning.
2) We introduce a discriminative class-conditional and a marginal distribution alignment strategy by using contrastive learning, which can force samples with the same label to concentrate together and the one with various labels to be far away.
3) We incorporate the target information to estimate the weight of source samples, which is more accurate to estimate the importance of source data than only using source information.

The rest of this article is organized as follows. Related works consisting of DA, PDA, and contrastive learning are introduced in Section II. The proposed CLA model is described in detail in Section III. Experiments and analysis are presented in Section IV. Finally, the conclusion is drawn in Section V.

II. Related Work

This section makes an overview of the related research topics, i.e., DA, PDA, and contrastive learning.

A. Domain Adaptation

Over the past decades, many DA methods have been developed to reduce distribution discrepancies between domains, in which domain-invariant feature representation learning is a common method. This section focuses on homogeneous unsupervised DA problems with a single-source domain.

Most early DA approaches align source and target distribution by minimizing different statistic moments, such as

MMD [5], [21] and CORAL [22]. For example, transfer component analysis (TCA) [5] bridges the marginal distribution distance across two domains via minimizing MMD, which maps data into a reproducing kernel Hilbert space (RKHS) and measures the embedding probabilities. Domain transfer multiple kernel learning (DTMKL) [21] extends TCA by simultaneously minimizing structural risk functional and the distribution distance. Meanwhile, it incorporates multiple kernels into MMD to achieve robust performance. Transfer joint matching (TJM) [23] introduces an instance reweighting scheme into MMD, aiming at down-weighting the importance of irrelevant instances when reducing the two marginal distributions. Considering that MMD is the first-moment matching approach, CORAL [22] is designed to compute both the first-order (mean) and second-order (covariance) statistics of the two latent features. Similar to CORAL, central moment matching (CMD) [24] introduces a higher order central moments matching strategy to align the domain-specific hidden representations. Instead of aligning global marginal distribution, joint distribution adaptation (JDA) [25] is an approach to reducing marginal and class-conditional distributions between two domains. Similar to JDA, stratified transfer learning (STL) [26] and asymmetric tritraining [27] are proposed to match the class-conditional distributions between different domains, where pseudolabels of the target domain are predicted by majority voting techniques. Furthermore, balanced distribution adaptation (BDA) [28] extends JDA by learning the importance of marginal and conditional discrepancies to tradeoff the two gaps. Recently, visual DA (VDA) [29] and joint statistical alignment (JGSA) [30] take the distance of intraclass and interclass into the training process, which minimizes intraclass discrepancies while maximizing interclass margins.

Deep learning (DL) has drawn great attention in recent years due to its powerful ability to learn rich representation. As a result, various DA approaches based on DL have been proposed to extract expressive transferable knowledge across domains. An early work of the deep DA approach is introduced in [31], which conducts extensive experiments to test the transferability of each layer. Similar to shallow DA methods, many deep DA approaches employ discrepancy measures to align hidden layers in order to achieve domain invariant representations. For instance, in domain adaptive neural networks (DANNs) [32], MMD is incorporated to explicitly reduce the distribution of hidden representations learned in the last layer. Deep adaptation network (DAN) [33] extends DANN by employing a deeper neural network and matching multiple task-specific layers. In an alternative way, residual transfer networks (RTNs) [34] attributes domain shifts to that the source classifier differs from the target classifier with a residual function. Therefore, it plugs several layers into the deep network to learn the residual function in order to adapt the source classifier. In recent, contrastive adaptation network (CAN) [35] develops a contrastive loss to minimize class-conditional distributions across domains. Except for MMD, CORAL is extended to learn a nonlinear transformation by aligning correlations of layer activations in DL [12]. In [36], the CORAL is further generalized to possibly infinite-dimensional covariance matrices in an RKHS.

More recently, GANs [37] gain increasing popularity in DA, referred to as domain adversarial networks, which learn domain-invariant representations via GANs [13]. The GAN is a structure that equips with a discriminator and a generator, where the two networks contest each other in a game. Typically, the generator tries to generate a data distribution of interest, while the discriminator aims to accurately evaluate whether a candidate from the generator or the true data distribution. In domain adversarial network [13], the generator plays the role of feature extractor to learn the latent feature of the source and the target domains, and the discriminator is designed as a domain classifier to differentiate the two distributions. Following this framework, in [38], the work replaces the domain discriminator with a network to learn an approximate Wasserstein distance. In [39], the loss of all blocks is incorporated to enhance the domain informative representations of lower blocks and uninformative representations from higher blocks. The adversarial discriminative DA (ADDA) [40] incorporates a discriminative modeling and an untied weight sharing technique into the domain adversarial network to achieve robust performance. Similar to ADDA, the work in [41] and [42] integrates a class-conditional distribution alignment to achieve a better adaptation.

B. Partial Domain Adaptation

PDA challenges standard DA methods due to source outlier classes, which may cause negative transfer if these outliers are aligned with the target domain. To mitigate negative transfer, most existing PDA methods focus on reducing contributions of unrelated source instances. For example, selective adversarial network (SAN) [15] adopts reweighting scheme to decrease the importance of outlier instances on the domain discriminator. The reweighting scheme employs multiple GANs to estimate the weights of source instances, where the weights of irrelevant instances are reduced. Partial adversarial DA (PADA) [16] adopts the trained classifier to estimate weights and puts the estimated weights on both the classifier and domain discriminator. In a similar way, the importance weighted adversarial net (IWAN) [17] exploits the prediction probability from a domain classifier to evaluate the importance of source samples. While example transfer network (ETN) [43] progressively quantifies the transferability of the source samples by introducing an auxiliary classifier. In a deep residual correction network (DRCN) [44], the importance of irrelevant source instances is weakened by plugging a residual block into the source task-specific feature layer. Dual Alignment for PDA (DAPDA) [45] proposes a reweighting network to generate class-level weights for source data and instance-level weights for target data. In contrast, the work in [18] and [46] selects the shared and filter out source-only samples via reinforced learning, where reconstruction errors of source data are adopted to calculate rewards. In [47], the PDA problem is treated as a class imbalance problem and the balanced adversarial alignment (BAA) technique is presented to reduce negative transfer. Instead of removing source outlier samples, BAA randomly leverages a few source samples to augment the target domain, which achieves state-of-the-art

results on several benchmark datasets. Most of the above-mentioned methods borrow adversarial domain networks to learn domain-shared representation. However, they do not mind discriminability representation which may lose as discussed in [19]. Meanwhile, they estimate source instance weight only using source information and neglect the target information.

C. Contrastive Learning

Even though deep supervised learning has achieved significant success in many fields, it often suffers from relying heavily on extensive labeled data, generalization error, spurious correlations, and adversarial attacks [20], [48]. Contrastive learning provides a promising alternative to alleviate the above drawbacks and enables learning underlying representation from unlabeled data. It learns representations based on data augmentation, which is the crop, resize, or recolor of the original sample in image data. With the augmentation, contrastive learning encourages encoding a sample (an anchor) and its augmentation (also known as similar sample, positive sample) be closer while the rest samples (different samples, negative sample) are far from each other [20], [48]. For a given anchor point x, suppose x^+ and x^- are the positive and negative points of x, the optimization function is formulated to as

$$\text{score}\big(f(x), f\big(x^+\big)\big) \gg \text{score}\big(f(x), f\big(x^-\big)\big) \tag{1}$$

where the score is a metric distance that measures the similarity between two features.

Following this framework, one class of work focuses on instance-based contrastive learning, where augmentations of an anchor point are considered as positives and the reset samples in the training data are negatives for every sample. For example, deep infoMax (DIM) [49] exploits a local patch from a global feature as a positive point and minimizes their mutual information distance. The global feature indicates the output of the final conventional layer, while the local patch is the output of an intermediate layer. In this way, DIM encourages the global feature vector to contain the information of the local region. AMDIM [50] improves DIM by providing multiple local patches, such as different locations, afferent modalities, or the different views of an image. Instead of using the local feature, contrastive multiview coding (CMC) [51] employs different transformations of an image as positive samples. It is known that the performance of contrastive learning often relies on the number of negative samples, which is redistricted by the batch size. To this end, momentum contrast (MoCo) [52] maintains a dynamic queue, in which representations of the current data are en-queued while the oldest are out-queued. In addition, it periodically updates the negative encoder to reduce the consistency of key representations. Most recently, SimCLR [53] improves MoCo by introducing multiple augmentation operations and dynamically learning the importance of a hard positive sample strategy.

Another line is cluster-based methods, which not only require representations of a pair of samples to be similar but also samples in a category to be closer. For instance, SwAV [54] simultaneously clusters the data and encourages the different augmentations of images in the same cluster to be closer in embedding space. In supervised learning, a contrastive loss instead of cross entropy loss is presented in [55]. It leverages label information to pull samples belonging to the same class together and achieves a better performance than cross entropy loss.

III. Proposed Method

A. Problem Definition and Overall Framework

Similar to the standard close-set unsupervised DA, an unsupervised PDA task constitutes a labeled source domain $D_s = \{(\mathbf{x}_i^s, y_i^s)\}_{i=1}^{n_s}$ associated with C_s classes, and an unlabeled target domain $D_t = \{\mathbf{x}_i^t\}_{i=1}^{n_t}$ drawn from C_t classes. n_s and n_t are the number of source and target data, respectively. $\mathbf{x}_i^s$ is a feature vector of a source sample and y_i^s is the corresponding label, $\mathbf{x}_i^t$ is a feature vector of a target sample. Suppose X_s, X_t, and Y_s, Y_t are feature and label spaces of D_s and D_t. The feature spaces X_s and X_t are identical in PDA, while the target label space Y_t is a subset of the source label space Y_s, that is $Y_t \subseteq Y_s$ and $C_t < C_s$. Due to the domain shift, the distributions of shared classes between the two domains are different, i.e., $P(\bar{X}_s) \neq P(X_t)$, where $P(\bar{X}_s)$ denotes the distribution of source data corresponding to Y_t. The goal of PDA is to filter out source outlier instances to mitigate negative knowledge transfer and reduce the difference between $P(\bar{X}_s)$ and $P(X_t)$ to boost positive knowledge transfer. With this in mind, the proposed CLA introduces contrastive learning that attempts to incorporate the discriminative class structure in distribution matching for better adaptation. Meanwhile, a new reweighting scheme involving the source and the target information is proposed to estimate the transferability of source data.

The architecture of CLA is shown in Fig. 2, which couples three parts: a domain adversarial network, a contrastive learning-based conditional alignment (CLCA), and a reweighting scheme. The domain adversarial network, including the feature exacter F, the domain discriminator D, and the classifier G_y, aims to learn domain invariant representations across the source and the target domains, and the transferable classifier. The CLCA involving a target cluster, G_t, exploits discriminative representations and class-conditional alignment between different domains to achieve better adaptation. In particular, the target cluster G_t is a fully connected layer that explores rich representations of the target data and separates them into C_t clusters via a contrastive loss. Subsequently, clusters in the target domain and classes in the source domain that corresponds to the sample class are separately aligned via a supervised contrastive loss, aiming at reducing class-level distribution mismatch. It is worth noting that the supervised contrastive loss encourages a rich representation of the source domain. In the proposed reweighting scheme, the information of the classifier G_y and the target cluster G_t are integrated to estimate the weight of source instances. Thereafter, these weights are fed back to the domain adversarial network, G_y, and G_t the source to reduce the effect of irrelevant source instances on the classifier and domain discriminator.

B. Domain Adversarial Learning Revisited

Domain adversarial learning introduces GANs to overcome the gap between two domains so that a classifier trained on the

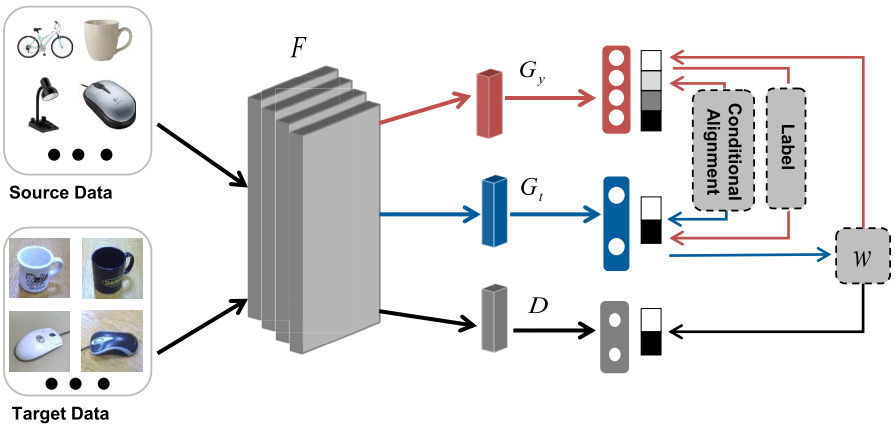


Fig. 2. Architecture of the proposed CAL, where F, D, G_y, and G_t are the feature extractor, the domain discriminator, the source classifier, and the target cluster, respectively.

source domain generalizes well to the target domain. As shown in Fig. 2, GANs achieve this goal by two procedures: a domain discriminator D is trained to distinguish the feature representation between the source and target domain, and a feature extractor F is trained to fool D. These two procedures are trained in a minimax optimization process, where D is learned by minimizing the loss of the domain discriminator, and F is learned by maximizing the loss of the domain discriminator. With the learned domain-invariant feature, a classifier G_y is trained on the source domain. Theoretically, by optimizing the above-mentioned three objectives, the domain shifts can be reduced on close-set domain adaption problems. However, due to the outlier classes, the pure domain adversarial learning network would cause performance degeneration or even induce negative transfer in addressing PDA. A popular method to alleviate this issue is reducing the importance of source-only instances by weighting source instances. The estimated weigh is put on the domain discriminator and the classifier. Following [43], we also implement the prediction uncertainty of target data into the training process, which is quantified by the entropy criterion $H(\mathbf{h}) = -\sum_{i=1}^{C_h} h_i \log(h_i)$, where C_h is the cardinality of $\mathbf{h}$. Suppose the weight of source instances is $\mathbf{w} = (w_1, w_2, \ldots, w_{n_s})$, the optimization objective of adversarial learning for PDA can be formally represented as

$$\begin{aligned}
&\min_{\theta_f,\theta_y} \max_{\theta_d} L_{\text{adv}}(\theta_f,\theta_d) + L_y(\theta_f,\theta_y) \\
&L_{\text{adv}}(\theta_f,\theta_d) = \frac{1}{n_s}\sum_{i=1}^{n_s} w_i \log\left[D\left(F\left(\mathbf{x}_i^s\right)\right)\right] \\
&\qquad\qquad + \frac{1}{n_t}\sum_{j=1}^{n_t} \log\left[1 - D\left(F\left(\mathbf{x}_j^t\right)\right)\right] \\
&L_y(\theta_f,\theta_y) = \frac{1}{n_s}\sum_{i=1}^{n_s} w_i l_{\text{ce}} G_y\left(F\left(\mathbf{x}_i^s\right)\right) \\
&\qquad\qquad + \frac{\gamma}{n_t}\sum_{j=1}^{n_t} H\left(G_y\left(F\left(\mathbf{x}_i^t\right)\right)\right)
\end{aligned} \tag{2}$$

where l_{ce} is a predefined supervised loss, such as cross entropy loss; $L_{\text{adv}}(\theta_f,\theta_d)$ is the domain adversarial loss, γ is a hyperparameter to tradeoff the supervised loss and the target prediction uncertain.

C. Contrastive Learning-Assisted Conditional Alignment

Contrastive learning aims to learn a embedding space, where similar samples are pulled together and diverse samples are pushed away in order to achieve robust representations for samples that semantic closer and discriminative representations between instances with dissimilar semantic [20], [48]. By using contrastive learning, the proposed CLCA is expected to benefit DA from two perspectives, enhancing the representation discriminability between different classes and the invariant representation corresponding to the same class across the two domains. To achieve this goal, we deploy two contrastive losses L_t and L_c to train the target cluster G_t and the class-conditional distribution alignment, respectively. Thus, the optimization objective of CLCA can be summarized as

$$L_{\text{clca}} = \min L_t(\theta_f,\theta_t) + L_c(\theta_f,\theta_t,\theta_y). \tag{3}$$

We detail the two contrastive losses as follows:

1) Contrastive Loss for the Target Cluster: The purpose of the target cluster G_t is to group target samples into C_t clusters, where C_t is the number of classes in the target domain. We expect samples in a single cluster to associate with the same label, while those in diverse clusters to different labels. Due to the lack of labels, we here adopt instance-based contrastive loss to train G_t. The instance-based contrastive loss considers an augmentation of a sample as a positive sample (similar samples), and the reset samples in a mini-batch as negatives (diverse samples). In this case, the common representation between original images and their corresponding augmentations are likely to be captured while specific details can be discarded, thus helpful to robust representation learning.

Let $\mathbf{X}^t = \{\mathbf{x}_1^t, \mathbf{x}_2^t, \ldots, \mathbf{x}_N^t\}$ be N original samples in a mini-batch. For the image sample in this work, we generate the augmentation of $\mathbf{X}$ via a crop and recolor [20], denoted as $\mathbf{X}^{t\prime} = \{\mathbf{x}_1^{t\prime}, \mathbf{x}_2^{t\prime}, \ldots, \mathbf{x}_N^{t\prime}\}$, via color transformation, jigsaw puzzle, Chen *et al.* [53]. After performing the feature extractor F

and the target cluster G_t, an image sample can be mapped to a C_t-dimension assignment feature. We refer the assignment features of original and augmentation samples in a mini-batch to as $\mathbf{Z}^t = \{\mathbf{z}_1^t, \mathbf{z}_2^t, \ldots, \mathbf{z}_N^t\}$ and $\mathbf{Z}^{t\prime} = \{\mathbf{z}_1^{t\prime}, \mathbf{z}_2^{t\prime}, \ldots, \mathbf{z}_N^{t\prime}\}$, respectively. Based on assignment features, we train G_t by the contrastive loss proposed in InfoNCE [56], which is formulated to as

$$L_t = -\sum_{i=1}^{N} \log \frac{\exp\left(\mathbf{z}_i^t \cdot \mathbf{z}_i^{t\prime}/\tau\right)}{\sum_{k\in A(i)} \exp\left(\mathbf{z}_i^t \cdot \mathbf{z}_k^t/\tau\right)} \tag{4}$$

where $\mathbf{z}_i^t$ is an anchor instance, $\mathbf{z}_i^{t\prime}$ is the positive point of $\mathbf{z}_i^t$, $A(i)$ are negative points, including the rest $2(N-1)$ samples, $\cdot$ denotes the inner product of two vectors, and $\tau > 0$ is a temperature parameter. By minimizing (4), the representations of paired instances $\mathbf{z}_i^t$ and $\mathbf{z}_i^{t\prime}$, $i = 1, 2, \ldots, N$, are forced closer.

2) Contrastive Loss for Class-Conditional Alignment: As pointed out in [42], the discriminative representation of source domains is important in DA, because the confusion distribution between diverse classes may lead to a target class aligning to a wrong source class, which further results in class-conditional distributions mismatch. To alleviate this issue, we employ a contrastive loss to align class-conditional distributions across the two domains. Previous close-set DA approaches often use predicted pseudolabels in the target domain and true labels in the source domain to achieve the class-conditional matching, where samples associated with the same label are aligned. Nevertheless, the target pseudolabels often exist noise because of the domain shit. Consequently, we select partial confident pseudolabels as the guidance to match classes across domains. Simply, we set confidence threshold ς to be the mean value of a mini-batch. Suppose $\mathbf{P}^t = \{\mathbf{p}_1^t, \mathbf{p}_2^t, \ldots, \mathbf{p}_N^t\}$ is assignment probabilities of the target data, ς is defined as

$$\varsigma = \frac{1}{N}\sum_{i=1}^{N} \max\left(\mathbf{p}_i^t\right). \tag{5}$$

The pseudolabel set U is accordingly defined as

$$U = \left\{\left(\mathbf{x}_i^t, \hat{\mathbf{y}}_i^t = \operatorname{argmax}\left(\mathbf{p}_i^t\right)\right); \text{if } \max\left(\mathbf{p}_i^t\right) > \tau\right\}, \quad i = 1, 2, \ldots, N. \tag{6}$$

Since labels are available, we make full use of label information and employ the supervised contrastive loss [55] for conditional alignment. Different from the target cluster contrastive loss, where only a single-positive sample for an anchor, samples with the same label are considered as positives, while items with various labels and their augmentations are treated as negative samples. In this way, the invariant features of samples in the same class can be extracted, and the discriminate characters between different classes are caught. As contrastive losses of the target cluster use assignment features, we employ the assignment probabilities to compute L_c to enhance the robust representation. Let $\mathbf{P}^s = \{\mathbf{p}_1^s, \mathbf{p}_2^s, \ldots, \mathbf{p}_N^s\}$ and $\mathbf{P}^{s\prime} = \{\mathbf{p}_1^{s\prime}, \mathbf{p}_2^{s\prime}, \ldots, \mathbf{p}_N^{s\prime}\}$ be assignment probabilities of original samples in the source domain and its augmentations, $\mathbf{P}^t = \{\mathbf{p}_1^t, \mathbf{p}_2^t, \ldots, \mathbf{p}_{|U|}^t\}$ and $\mathbf{P}^{t\prime} = \{\mathbf{p}_1^{t\prime}, \mathbf{p}_2^{t\prime}, \ldots, \mathbf{p}_{|U|}^{t\prime}\}$ denote assignment probabilities of target samples in U and its augmentations, where $|U|$ is the cardinality of U. We further let $\mathbf{P} = \mathbf{P}^s \cup \mathbf{P}^t$, and $\mathbf{P}\prime = \mathbf{P} \cup \mathbf{P}^{s\prime} \cup \mathbf{P}^{t\prime}$, that is $\mathbf{P} = \{\mathbf{p}_1, \mathbf{p}_2, \ldots, \mathbf{p}_N, \mathbf{p}_{N+1}, \mathbf{p}_{N+2}, \ldots, \mathbf{p}_{N+|U|}\}$, and $\mathbf{P}' = \{\mathbf{p}_1, \ldots, \mathbf{p}_{N+|U|}, \mathbf{p}_1', \ldots, \mathbf{p}_{N+|U|}'\}$. The contrastive loss for class-conditional alignment can be formulated as

$$L_c = -\sum_{i=1}^{2(N+|U|)} \frac{1}{2N_i - 1} \sum_{j\in C(i)} \log \frac{\exp\left(\mathbf{p}_i \cdot \mathbf{p}_j'/\tau\right)}{\sum_{k\in A(i)} \exp\left(\mathbf{p}_i \cdot \mathbf{p}_k'/\tau\right)} \tag{7}$$

where $C(i)$ and $A(i)$ are picked from $\mathbf{P}\prime$, $C(i)$contains positives that are with the same pseudolabel of the anchor $\mathbf{p}_i$, N_i is the number of samples in $\mathbf{P}$ having the same label with $\mathbf{p}_i$, and $A(i)$ consists of the negatives, which have different labels with $\mathbf{p}_i^s$.

D. Reweighting Scheme

As discussed earlier, the weight $\mathbf{w}$ aims to estimate the importance of source data so that to remove source specific data, which plays a key role to mitigate negative transfer. In our proposed new reweighting scheme, the estimated $\mathbf{w}$ incorporates two parts, $\mathbf{w}^s$ and $\mathbf{w}^t$, which are generated by using source information and target information, respectively, but both of them predict the weight of source data. In particular, $\mathbf{w}^s$ is a class-level weight predicted via classifier G_y. Similar to [16], we first employ G_y to predict target data and get the outputs

$$\hat{\mathbf{y}}_i^t = G_y\left(F\left(\mathbf{x}_i^t\right)\right), \quad i = 1, 2, \ldots, n_t \tag{8}$$

where $\hat{\mathbf{y}}_i^t$ is a C_s-dimensional assignment feature of $\mathbf{x}_i^t$, denotes the probabilities assigned to each of the C_s classes. Then, we calculate the class-level weights by averaging all assignment features, that is,

$$\mathbf{w}^s = \frac{1}{n_t}\sum_{i=1}^{n_t} \hat{\mathbf{y}}_i^t \tag{9}$$

$\mathbf{w}^s$ is further normalized via

$$w_i^s = w_i^s / \sum_{i=1}^{C_s} w_i^s. \tag{10}$$

Since the target data does not belong to the source outlier label space, the assignment feature values corresponding to the source outlier classes are sufficiently small. Therefore, the weight of source-only samples is also significantly smaller than that of the shared samples.

Different from $\mathbf{w}^s$, $\mathbf{w}^t$ is an instance-level weight estimated by the target cluster G_t. It is reasonable that a source data is likely to get a higher probability to one of target classes if it belongs to the shared classes. On the contrary, an outlier source data is hardly recognized by the target cluster G_t, and thus, probabilities to any class are much smaller. Inspired by the above-mentioned findings, we apply G_t to predict source data and obtain their assignment features

$$\hat{\mathbf{y}}_i^s = G_t\left(F\left(\mathbf{x}_i^s\right)\right), \quad i = 1, 2, \ldots, n_s \tag{11}$$

where $\hat{\mathbf{y}}_i^s$ is a C_t-dimensional assignment feature of $\mathbf{x}_i^s$, denotes probabilities assigned to the C_t classes. These assignment

features $\hat{\mathbf{y}}_i^s, i = 1, 2, \ldots, n_s$ are further passed through a softmax activation to get assignment probabilities, which are denoted as $\hat{\mathbf{y}}_{p,i}^s, i = 1, 2, \ldots, n_s$. Thereafter, we calculate the instance-level weight $\mathbf{w}^t$ according to

$$w_i^t = 1 + e^{-H\left(\hat{\mathbf{y}}_{p,i}^s\right)}, \quad i = 1, 2, \ldots, N \tag{12}$$

where $H(\hat{\mathbf{y}}_{p,i}^s)$ is the entropy criterion of the ith instance, and N is the mini-batch size.

Once $\mathbf{w}^s$ and $\mathbf{w}^t$ are obtained, we simply combine them as the final weight $\mathbf{w}$ via

$$w_i = w_{d(i)}^s w_i^t, \quad i = 1, 2, \ldots, n_b \tag{13}$$

where $d(i)$ is the class index of the ith instance.

E. Overall Objective and Training

As the aforementioned description, the proposed CLA aligns discriminative class-conditional distributions and the marginal distribution across the source and the target domains in the shared class space. Meanwhile, it applies the trained classifier and the target cluster to estimate the weight of source data in order to rule out the source outlier instances. By incorporating (3) and (2), the overall objective of CLA is summarized as

$$\min L_{\text{adv}}\left(\theta_f, \theta_d\right) + L_y\left(\theta_f, \theta_y\right) + \beta\left(L_t\left(\theta_f, \theta_t\right) + L_c\left(\theta_f, \theta_y, \theta_t\right)\right) \tag{14}$$

where β is a hyperparameter to tradeoff the losses. We optimize (14) by the standard backpropagation training approach with two stages. In the first stage, it jointly minimizes the class-conditional distribution and the marginal distribution alignment losses by L_{adv}, L_t, and L_c. Meanwhile, it optimizes the classification loss via L_y and calculates the instance-level weight $\mathbf{w}^t$. In the second stage, the class-level weight $\mathbf{w}^t$ is estimated using the information from the source domain. Then, the final weight is applied to (14). The training process of the CLA is summarized in Algorithm 1.

Algorithm 1 CLA

Input: D_s: The source domain; D_t: The target domain; N: the size of minibatch; F: The feature extractor; D: The domain classifier; G_y: The classifier; G_s: The cluster in the source domain; T: The validation interval.
Output: F: The trained feature extractor; D_y: The trained classifier.
Pre-train the feature extractor F;
while not converge **do**
  Compute the class-level weight $\mathbf{w}^s$ by Eq. (10);
  for $t = 1 : T$ **do**
    Sample minibatch $\{(\mathbf{x}_i^s, y_i^s)\}_{i=1}^N$ from D_s;
    Get the argumentation of $\{(\mathbf{x}_i^s, y_i^s)\}_{i=1}^N$;
    Sample minibatch $\{\mathbf{x}_i^t\}_{i=1}^N$ from D_t;
    Get the argumentation of $\{\mathbf{x}_i^t\}_{i=1}^N$;
    Compute the instance-level weight $\mathbf{w}^t$ by Eq. (12);
    Compute the final weight $\mathbf{w}$ by Eq. (13);
    Compute the adversarial learning loss by Eq. (2);
    Select target confident pseudo-label data according to Eq. (6);
    Compute the CLCA loss by Eq. (3);
    Update parameters in θ_f, θ_g, θ_y, θ_s, and θ_t;
  end for
end while

Fig. 3. Images examples of datasets Office-31, Office-Caltech, and Office-Home.

IV. Experiments and Analyses

In this section, we first compare the proposed method with several competitive unsupervised PDA approaches on real-world datasets to evaluate the performance of CLA. We further conduct several empirical experiments to examine the flexibility and effectiveness of CLA. Code is available at: https://github.com/Peacefulyang/HE-CDTL.

A. Experimental Settings

1) Datasets: We verify the performance of our approach on three cross-domain recognition tasks: Office-31 [57], Office-Caltech [58], and Office-Home [59]. Fig. 3 visualizes sample images in per dataset.

The **Office-31** dataset is a widely used benchmark for DA, where each image is downloaded from amazon.com, or an Office environment picture picked up by a webcam or a DSLR camera. The dataset has around 4652 images collected from three real-world object domains: Amazon (A), DSLR (D), and Webcam (W), and each domain includes 31 classes. Similar to [15], in the PDA scenario, a source domain contains 31 classes, and a target domain involves ten classes that Office-31 and Caltech-256 share. Take one domain as the source domain and another as the target domain. Therefore, six PDA tasks across three domains are conducted: A31 → W10, A31 → D10, D31 → A10, D31 → W10, W31 → A10, and W31 → D10.

The **Office-Caltech** dataset is constructed by Office-31 and Caltech-256 datasets, which share ten common classes. The Caltech-256 (C) dataset [60] is a standard dataset for object recognition and consists of 30 607 images collected from 256 classes. Following to [17], we take one domain with ten shared classes as a source domain and one domain with the first five common classes (in alphabetical order) as a target domain, and thus, 12 across tasks can be built: A10 → C5, A10 → D5, A10 → W5, C10 → A5, C10 → D5,

TABLE I
ACCURACY % ON OFFICE-CALTECH FOR PDA VIA ALEXNET. THE BEST RESULTS ARE HIGHLIGHTED IN BOLD

Methods	A10→C5	A10→D5	A10→W5	C10→A5	C10→D5	C10→W5	D10→A5	D10→C5	D10→W5	W10→A5	W10→C5	W10→D5	Average
AlexNet [61]	85.27	85.29	76.30	93.58	91.18	83.70	89.51	80.82	98.52	87.37	74.14	**100.0**	87.14
DANN [63]	77.57	80.88	65.93	91.86	83.82	82.22	77.09	69.35	80.74	80.30	72.60	59.59	79.83
RTN [34]	80.99	70.59	69.63	91.86	80.88	93.99	70.02	59.08	91.11	74.73	59.08	**100.0**	78.44
ADDA [40]	85.27	89.71	87.41	93.15	97.06	94.07	93.79	89.90	98.52	92.08	86.82	**100.0**	92.31
IWAN [17]	89.90	88.24	87.41	94.22	98.53	97.78	94.43	91.61	98.52	95.29	90.24	**100.0**	93.85
PADA [16]	92.05	98.76	87.33	95.25	97.59	96.00	96.39	95.80	97.87	96.14	**96.85**	**100.0**	95.70
DARL [46]	92.47	**100.0**	88.89	96.36	98.52	**100.0**	95.93	92.64	99.26	96.15	93.15	**100.0**	96.11
DAPDA [45]	93.05	98.83	93.46	**98.66**	**100.0**	97.93	97.02	94.62	99.10	96.62	94.35	**100.0**	96.97
CLA	**96.32**	**100.0**	**100.0**	98.03	**100.0**	**100.0**	**98.15**	**96.52**	**99.94**	**98.57**	95.43	**100.0**	**98.58**

C10 → W5, D10 → A5, D10 → C5, D10 → W5, W10 → A5, W10 → C5, and W10 → D5.

The **Office-Home** dataset is an object recognition dataset that contains 15 500 images from four domains, Artistic (A), Clipart (C), Product (P), and Real-World (R), where each domain includes 65 object classes. We follow [16] to build PDA tasks, where a target domain is made of the first 25 classes (in alphabetical order) in that domain. In these four domains, we take one domain as the source domain and each of the rest domains as the target domain. Thus, 12 cross-domain PDA tasks can be constructed: A → C, A → P, A → R, C → A, C → P, C → R, P → A, P → C, P → R, R → A, R → C, and R → P.

2) Baseline Methods: We compare the proposed CLA with representative or state-of-the-art PDA and DA baselines, including two baseline convolutional neural networks: AlexNet [61] and ResNet-50 [62], eight PDA methods: SAN [15], IWAN [17], PADA [16], ETN [43], DRCN [44], DAPDA [45], DARL [46], and BA^{3}US [47], and three standard close-set DA methods, i.e., RTN [34], ADDA [40], DANN [63], and SHOT [64].

3) Implementation Details: We follow standard protocols and use all labeled source data and unlabeled target data for unsupervised PDA. For all the experiments, AlexNet and ResNet-50 are employed as two baselines. The proposed CLA and other compared models are implemented with PyTorch using NVIDIA Tesla V100. Similar to [43] and [47], we fine-tune the PyTorch-provided AlexNet and ResNet-50 that are pretrained on the ImageNet dataset for a fair comparison. Especially, we obtain the feature extractor F by replacing the last fully connected layer in the two pretrained models with a bottleneck layer (containing 256 units). The model is trained through backpropagation, and we set the learning rate of the residual layer to be one-tenth that of the other layers. We adopt minibatch stochastic gradient descent (SGD) with momentum of 0.9 and a learning rate annealing strategy as [63]: the learning rate is dynamically adjusted in the training process using $\eta_p = \eta_0(1 + \hat{\alpha}\, p)^{-\hat{\beta}}$, where p refers to the training progress changing from 0 to 1, $\eta_0 = 0.001$, $\hat{\alpha}\, p = 10$, and $\hat{\beta} = 0.75$. As suggested in [15], the penalty of the adversarial discriminator layer is gradually changing from 0 to 1 to achieve stable performance. We set the balancing coefficient of contrastive learning-based conditional alignment β to 0.1 for Office-Home and 0.5 for Office-31 and Office-Caltech. Following to [47] and [55], γ is set to 0.1, τ is set to 0.09. Besides, we set the batch size of each domain for each method to be 36. Furthermore, the number of validation intervals is set to 200, 200, and 500 for Office-31, Office-Caltech, and Office-Home, respectively.

B. Experimental Results on Partial Domain Adaptation

1) Results on Office-Caltech Dataset: Table I reports classification accuracies of all compared approaches on the Office-Caltech dataset using AlexNet as the baseline, where the best results are highlighted in bold. From Table I, we can make the following observations. First, some standard DA methods that assume the source and the target domain sharing the same label space achieve worse performances on PDA tasks. For example, DANN and RTN perform worse than AlexNet on most tasks, implying that only aligned distributions across domains are likely to cause negative transfer due to outlier instances. Whereas ADDA beats AlexNet by an average accuracy improvement of 5.17%, the underlying reason may be that incorporating source class discriminative information into domain distribution alignment can significantly reduce mismatch on PDA problems. Second, the PDA methods achieve much better performances than standard DA methods. For instance, the average classification accuracy of IWAN is 6.17% higher than AlexNet, and DAPDA, the state-of-the-art PDA method, is 9.83% higher than AlexNet. This observation shows that detecting and removing irrelevant samples is essential in alleviating negative transfer on PDA tasks. Third, CLA outperforms all the other comparison methods on most tasks and obtains the best performance on 10 out of 12 tasks. It is worth noting that CLA significantly enhances the accuracy on several tasks, e.g., A10 → D5, A10 → W5, and C10 → D5. For the average results, CLA outperforms DAPDA and achieves the highest accuracy among the compared methods. The promising performance of CLA can be attributed to the proposed strategies, the reweighting scheme, and the contrastive learning-assisted class-conditional distribution alignment.

2) Results on Office-31 Dataset: Table II shows comparison results of different methods on the Office-31 dataset with ResNet-50 as the baseline. Similar to Table I, CLA consistently outperforms the best PDA method BA^{3}US and achieves the highest average accuracy among the compared methods.

TABLE II
ACCURACY % ON OFFICE-31 FOR PDA VIA RESNET-50. THE BEST RESULTS ARE HIGHLIGHTED IN BOLD

Methods	A31 → D10	A31 → W10	D31 → A10	D31 → W10	W31 → A10	W31 → D10	Average
ResNet-50 [62]	83.44	75.59	83.92	96.27	84.97	98.09	87.05
DANN [63]	73.56	96.27	98.73	81.53	82.78	86.12	86.50
RTN [34]	66.88	75.25	85.59	97.12	85.70	98.32	84.81
ADDA [40]	83.41	75.67	83.62	95.38	84.25	99.85	87.03
IWAN [17]	90.45	89.15	95.62	99.32	94.26	99.36	94.69
SAN [15]	94.27	93.90	94.15	99.32	88.73	99.36	94.96
PADA [16]	82.17	86.54	92.69	99.32	95.41	**100.0**	92.69
DARL [46]	98.73	94.58	94.57	99.66	94.26	**100.0**	96.97
DAPDA [45]	92.15	95.06	95.13	**100.0**	97.40	**100.0**	96.62
DRCN [44]	86.00	88.05	95.60	**100.0**	95.80	**100.0**	94.30
ETN [43]	95.03	94.52	**96.21**	**100.0**	94.64	**100.0**	96.73
BA^{3}US [47]	99.36	98.98	94.82	**100.0**	94.99	98.73	97.81
CLA	**100.0**	**100.0**	94.53	**100.0**	**96.65**	**100.0**	**98.53**

TABLE III
ACCURACY % ON OFFICE-HOME FOR PDA VIA RESNET-50. THE BEST RESULTS ARE HIGHLIGHTED IN BOLD

Methods	A→C	A→P	A→R	C→A	C→P	C→R	P→A	P→C	P→R	R→A	R→C	R→P	Average
ResNet-50 [62]	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 [63]	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
RTN [34]	64.33	49.37	76.19	51.74	47.56	57.67	50.38	41.45	75.53	74.78	70.17	51.82	59.25
ADDA [40]	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
IWAN [17]	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 [15]	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 [16]	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
DARL [46]	55.31	80.73	86.36	67.93	66.16	78.52	68.74	50.93	87.74	79.45	57.19	85.60	72.06
DAPDA [45]	56.49	77.56	80.29	65.73	71.52	77.28	66.53	55.96	85.65	77.02	60.82	84.82	71.64
DRCN [44]	54.00	76.40	83.00	62.10	64.50	71.00	70.80	49.80	80.50	77.50	59.10	79.90	69.00
ETN [43]	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
BA^{3}US [47]	60.62	83.16	88.39	71.75	72.79	83.40	75.45	61.59	86.53	79.25	62.80	86.05	75.98
SHOT [64]	64.8	85.2	**92.7**	**76.3**	**77.6**	**88.8**	79.7	64.3	**89.5**	80.6	66.4	85.8	79.3
CLA	**66.72**	**85.64**	90.90	75.64	76.85	86.81	**78.79**	**67.35**	88.72	**81.69**	**66.85**	**87.83**	**79.48**

Specifically, it obtains the best results on five out of six tasks and no notable worse results on the other three tasks compared with the other methods. Besides, BA^{3}US and CLA dramatically improve the classification accuracy than the other methods on some tasks, such as A31 → W10 and A31 → D10. A possible reason is that both BA^{3}US and CLA encourage the source and the target class discriminability, which can increase the margin between different classes and improve the robustness of a classifier. This further implies the importance of discriminative representation learning in DA.

3) Results on Office-Home Dataset: Table III shows classification results of CLA and other compared methods on the Office-Home dataset with ResNet-50 as the baseline. The Office-Home dataset is a more complex dataset with 40 outlier classes, which is easier to cause negative transfer than Office-31. In this situation, the standard DA methods perform not much better than ResNet-50, again validating that they suffer from negative transfer in addressing PDA tasks. The proposed CLA continuously outperforms other compared methods and enhances an average improvement of 0.18% over the state-of-the-art method SHOT. In particular, CLA obtains the best performance on seven out of 12 tasks and is slightly worse than SHOT on the rest three tasks. The results in Table III further demonstrate the effectiveness of CLA in handling more challenging PDA problems.

In summary, the results in Tables I–III reveal that the standard DA methods achieve overall comparable performance with the baselines on PDA tasks. The early PDA methods, e.g., IWAN, SAN, and PADA, with considering source-only instance detection, significantly increase classification accuracy on the Office-Caltech and Office-31 datasets. This validates that the outlier instances in PDA tasks easily lead to negative transfer, which can be reduced by appropriately removing these outliers. Besides, the recent PDA methods with carefully designed outlier instance detection strategies, such as ETN and DARL, further enhance the performance of PDA tasks. The phenomenon also indicates the significance of outlier detection in reducing negative knowledge transfer. Furthermore, DAPDA that incorporates discriminative class-conditional distribution alignment and BA^{3}US that applies domain entropy minimization strategy, achieve much better performance on the Office-Home and Office-31 datasets, implying that the discriminative representation plays an essential role in enhancing positive knowledge transfer.

TABLE IV
ACCURACY % ON OFFICE-HOME FOR CLOSE-SET DOMAIN ADAPTATION VIA RESNET-50. THE BEST RESULTS ARE HIGHLIGHTED IN BOLD

Methods	A→C	A→P	A→R	C→A	C→P	C→R	P→A	P→C	P→R	R→A	R→C	R→P	Average
ResNet-50 [62]	34.9	50.0	58.0	37.4	41.9	46.2	38.5	31.2	60.4	53.9	41.2	59.9	46.1
JAN [65]	45.9	61.2	68.9	50.4	59.7	61.0	45.8	43.4	70.3	63.9	52.4	76.8	58.3
CDAN [63]	49.0	69.3	74.5	54.4	66.0	68.4	55.6	48.3	75.9	68.4	55.4	80.5	63.8
CDAN+E [63]	50.7	70.6	76.0	57.6	70.0	70.0	57.4	50.9	77.3	70.9	56.7	81.6	65.8
DRCN [44]	50.6	72.4	76.8	61.9	69.5	71.3	60.4	48.6	76.8	72.9	56.1	81.4	66.6
BA^{3}US [47]	51.2	73.8	78.1	63.3	73.4	73.6	63.3	54.5	80.4	72.6	56.7	83.7	68.7
SHOT [64]	57.1	**78.1**	81.5	**68.0**	**78.2**	**78.1**	**67.4**	54.9	**82.2**	73.3	58.8	84.3	71.8
CLA	**58.2**	77.4	**83.2**	67.2	76.7	77.5	65.6	**57.4**	81.7	**75.8**	**61.2**	**84.5**	**72.2**

TABLE V
ACCURACY % OF CLA AND ITS VARIANTS FOR PDA ON OFFICE-HOME VIA RESNET-50. THE BEST RESULTS ARE HIGHLIGHTED IN BOLD

Methods	A→C	A→P	A→R	C→A	C→P	C→R	P→A	P→C	P→R	R→A	R→C	R→P	Average
CLA/CLCA	57.13	76.81	84.98	72.13	71.15	81.03	73.83	60.90	85.97	77.78	62.78	84.59	74.09
CLA/$\mathbf{w}^t$	66.13	82.58	86.75	73.84	**77.25**	84.05	77.23	**67.42**	86.64	78.52	**67.83**	86.21	77.86
CLA	**66.72**	**85.64**	**90.90**	**75.64**	76.85	**86.81**	**78.79**	67.35	**88.72**	**81.69**	66.85	**87.83**	**79.48**

Moreover, the proposed CLA achieves the best results on the three datasets, indicating the effectiveness of the proposed reweighting scheme and CLCA strategies in CLA in promoting transferable knowledge for PDA.

C. Experimental Results on Close-Set Domain Adaptation

To verify the generalization ability of our proposed CLA method on closed-set DA, we further conduct experiments to compare CLA with both of state-of-the-art DA and PDA approaches, including BA^{3}US [28], DRCN [44], CDAN [63], SHOT [64], JAN [65], and on the Office-Home dataset. The average classification accuracies using ResNet-50 as the baseline are reported in Table IV. As shown in Table IV, CLA outperforms the best DA method, CDAN + E, which considers class-conditional distribution across domains on all the tasks. Meanwhile, it surpasses the best PDA approach, SHOT, by improving the average accuracy by 0.4%. This is because the designed CLCA in CLA preserves class-invariant knowledge across the two domains and the discriminability between different classes. The superior performance of CLA suggests that it can be successfully extended to address close-set DA problems.

D. Discussions and Analyses

1) Ablation Studies: To further examine the effectiveness of the proposed CLCA and reweighting scheme, we compare CLA with two variations: CLA/CLCA is a variant without contrastive learning-assisted class-condition alignment, and CLA/$\mathbf{w}^t$ is a variant that removes the target instance-level weight in the reweighting scheme. We show the comparison results of CLA, CLA/CLCA, and CLA/$\mathbf{w}^t$ on the Office-Home dataset in Table V. It can be observed that our CLA outperforms CLA/CLCA on all tasks and significantly promotes accuracies on several tasks, i.e., A → P and A → R.

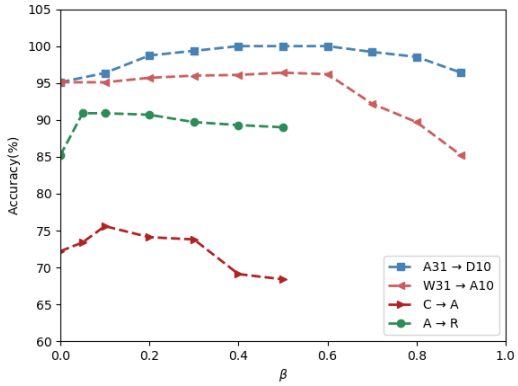


Fig. 4. Accuracy of four tasks with different β.

This observation demonstrates that the contrastive learning-assisted class-condition alignment enhances positive knowledge transfer by learning class discriminative representation. Compared with CLA/$\mathbf{w}^t$, CLA also performs better in a large margin, which verifies that the target instance-level weight can enhance the performance of filtering out outlier instances, thus reducing negative transfer.

2) Parameter Analysis: We study the sensitivity of parameter β of CLA on four tasks: C → A, A → R, A31 → D10, and W31 → A10. The experimental results by varying $\beta = \{0.0, 0.05, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9\}$ are shown in Fig. 4. We can see that the C → A and A → R tasks achieve the best accuracy around 0.1, while the best accuracy of A31 → D10 and W31 → A10 is obtained at about 0.5. This result is desirable because the CLCA cannot well exploit discriminative representation and align class-conditional distributions across the two domains when β is smaller. On the contrary, CLCA plays a major role in the performance of CLA if β is too large. In this scenario, the wrongly selected pseudolabels

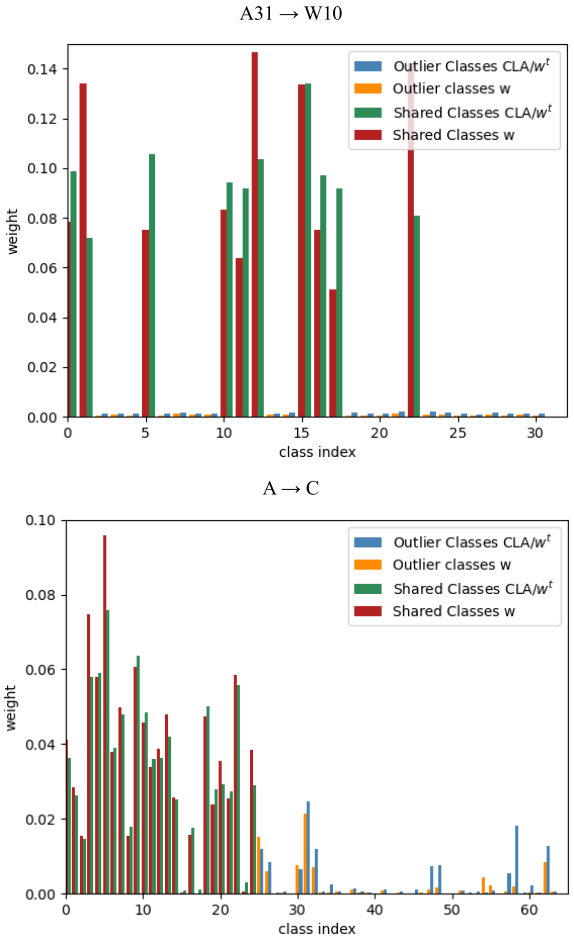


Fig. 5. Estimated class weights on two different transfer tasks. Blue and orange bins indicate the weights of source-only classes estimated by CLA without w^t and CLA, and green and red bins represent the weights of shared classes estimated by CLA without w^t and CLA.

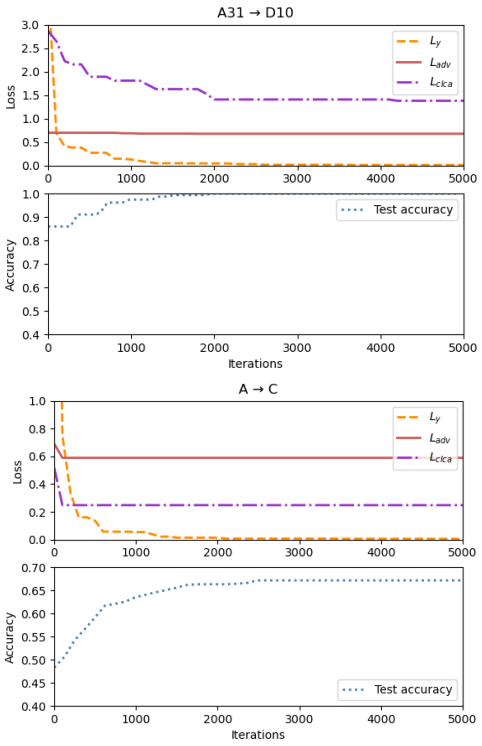


Fig. 6. Convergence analysis of proposed methods on two different transfer tasks, including domain adversarial loss (L_{adv}), cross entropy loss of the classifier (L_y), CLCA loss (L_{clca}), and the test accuracy.

in class-conditional alignment will lead to an error increase in the training process, thus resulting in performance derogation. Compare with tasks in the Office-Home dataset, tasks in the Office-31 dataset use a larger β. The underlying reason is that tasks in the Office-31 dataset are much easier and can obtain a lot of high-quality pseudolabels. Thus, a larger weight of CLCA can greatly increase the performance of this dataset. From the above-mentioned analysis, we can generally set β according to the complexity of tasks, where the value decreases with the complexity increasing.

3) Weight Visualization: Since the weighting scheme is essential to filter out outlier instances in PDA, we verify the statistical weights of source instances estimated by CLA. Fig. 5 presents the weights calculated per class on two specific tasks: A31 → W10 of Office-31 and A → C of Office-Home. In Fig. 5, blue and orange bins indicate the weights of source-only classes estimated by CLA without w^t and CLA, and green and red bins represent the weights of shared classes estimated by CLA without w^t and CLA. On the easy task A31 → W10, the weights predicted by CLA and CLA without w^t of source-only classes are significantly smaller than that of the shared classes. Even though the smallest weight of shared classes, it is about 45 times larger than the largest source-only weight, indicating the weight estimation is accurate. Accordingly, the prediction is close to the ground truth, as shown in Table II. On the more difficult task A → C, the weight estimation is relatively inaccurate and several shared-class weights are lower than the irrelevant-class weights. Meanwhile, the classification accuracy of A → C is much smaller than A31 → W10, which may result from the existence of a wrong class distribution match. Take a closer observation, it can be found that the orange bins are generally lower than the blue ones on both of the two tasks, indicating that w^t is able to help CLA to filter out outlier classes more accurately.

4) Convergence Analysis: We investigate the convergence performance of the proposed CLA on the A31 → W10 task and A → C task. Fig. 6 plots the domain adversarial loss (L_{adv}), cross entropy loss of the classifier (L_y), contrastive learning-based class-conditional alignment loss (L_{clca}), and the test accuracy with respect to training iterations. We can observe that the three losses in CLA gradually converge to the lowest test error, and its three losses are quickly converged to their lowest values. We can also can find that L_{adv} in the A31 → W10 quickly converges, while L_{clca} still decreases with the increase of test accuracy. This observation further implies that discriminative class-distribution alignment plays an important role in enhancing positive transfer.

5) Feature Visualization: Fig. 7 visualizes the t-SNE [66] of the feature using ResNet-50 as the baseline learned from four methods, PADA, ETN, BA^{3}US, and our CLA on A31 → W10. In Fig. 7, the gray, red, and blue dots indicate source samples in outlier classes, source samples in the shared

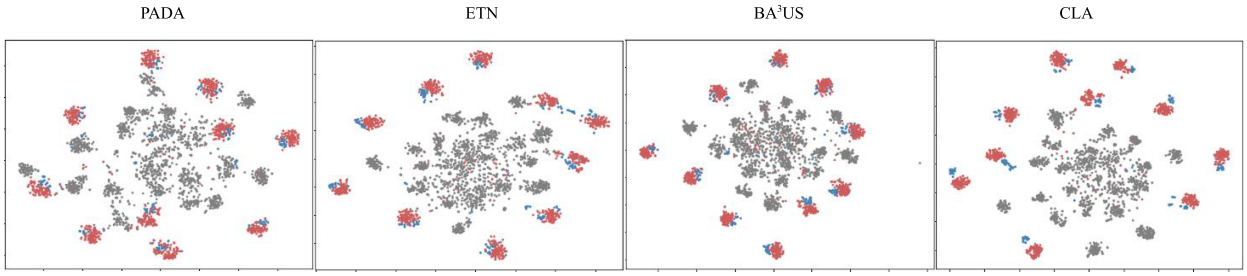


Fig. 7. t-SNE visualization of features learned from PADA, ETN, BA^{3}US, and CLA on task A31 → W10, where gray, red, and blue dots represent source data in the source-only classes, source data in the shared classes, and target data, respectively.

classes, and target data, respectively. In PADA, the data structures are more scattered as many outlier samples are close to the shared classes. ETN significantly improves the distribution alignment, but some target classes are mixed. By using domain entropy minimization strategy in both the source and the target domain, BA^{3}US separates the categories very well. However, many target data are aligned with the source-only samples. Unlike them, CLA can not only successfully match the shared data between the two domains but also preserve good structures of classes. These results reveal the superior performance of CLA in comparison with the other methods.

V. Conclusion

This article has introduced CLA to address PDA problems, where the label space of the source domain is a subset of the target domain. CLA exploits contrastive learning to cluster data in both domains, aiming to learn the class-conditional structures. Meanwhile, it matches the distributions of clusters in different domains but corresponds to the same class to match class-conditional distribution based on the marginal distribution alignment, which proposes to enhance positive knowledge transfer. Furthermore, CLA utilizes the predictions of the classier and the target cluster to estimate the weight of source data. In this way, the importance of irrelevant instances can be effectively decreased, thus reducing negative transfer. The effectiveness of the proposed CLA has been demonstrated on several benchmarks by comparing with existing methods.

In the reweighting scheme, we multiply the class-level and instance-level weights as the final weight. In reality, another simple way to combine these two weights is a summation. Thus, we prefer to investigate the performance of adding class-level and instance-level weights.

References

[1] S. Zhang, Z. Ma, G. Zhang, T. Lei, R. Zhang, and Y. Cui, "Semantic image segmentation with deep convolutional neural networks and quick shift," *Symmetry*, vol. 12, no. 3, p. 427, Mar. 2020.

[2] K. Muhammad, S. Khan, J. D. Ser, and V. H. C. D. Albuquerque, "Deep learning for multigrade brain tumor classification in smart healthcare systems: A prospective survey," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 32, no. 2, pp. 507–522, Feb. 2021.

[3] J. D. Kelleher, *Deep Learning*. Cambridge, MA, USA: MIT Press, 2019.

[4] S. J. Pan and Q. Yang, "A survey on transfer learning," *IEEE Trans. Knowl. Data Eng.*, vol. 22, no. 10, pp. 1345–1359, Oct. 2009.

[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, no. 2, pp. 199–210, Feb. 2010.

[6] G. Wilson and D. J. Cook, "A survey of unsupervised deep domain adaptation," *ACM Trans. Intell. Syst. Technol.*, vol. 11, no. 5, pp. 1–46, 2020.

[7] Q. Kang, S. Yao, M. Zhou, K. Zhang, and A. Abusorrah, "Effective visual domain adaptation via generative adversarial distribution matching," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 32, no. 9, pp. 3919–3929, Sep. 2021.

[8] M. Gong, K. Zhang, T. Liu, D. Tao, C. Glymour, and B. Schölkopf, "Domain adaptation with conditional transferable components," in *Proc. Int. Conf. Mach. Learn. (ICML)*, 2016, pp. 2839–2848.

[9] S. Cicek and S. Soatto, "Unsupervised domain adaptation via regularized conditional alignment," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 1416–1425.

[10] N. Courty, R. Flamary, A. Habrard, and A. Rakotomamonjy, "Joint distribution optimal transportation for domain adaptation," in *Advances in Neural Information Processing Systems*, vol. 30. Red Hook, NY, USA: Curran Associates, 2017. [Online]. Available: https://proceedings.neurips.cc/paper/2017/file/0070d23b06b1486a538c0eaa45dd167a-Paper.pdf

[11] J. Wang, Y. Chen, W. Feng, H. Yu, M. Huang, and Q. Yang, "Transfer learning with dynamic distribution adaptation," *ACM Trans. Intell. Syst. Technol.*, vol. 11, no. 1, pp. 1–25, Feb. 2020.

[12] B. Sun and K. Saenko, "Deep coral: Correlation alignment for deep domain adaptation," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2016, pp. 443–450.

[13] Y. Ganin *et al.*, "Domain-adversarial training of neural networks," *J. Mach. Learn. Res.*, vol. 17, no. 1, pp. 2030–2096, May 2015.

[14] E. Vural, "Generalization bounds for domain adaptation via domain transformations," in *Proc. IEEE 28th Int. Workshop Mach. Learn. Signal Process. (MLSP)*, Sep. 2018, pp. 1–6.

[15] Z. Cao, M. Long, J. Wang, and M. I. Jordan, "Partial transfer learning with selective adversarial networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 2724–2732.

[16] Z. Cao, L. Ma, M. Long, and J. Wang, "Partial adversarial domain adaptation," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 135–150.

[17] J. Zhang, Z. Ding, W. Li, and P. Ogunbona, "Importance weighted adversarial nets for partial domain adaptation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 8156–8164.

[18] Z. Chen, C. Chen, Z. Cheng, B. Jiang, K. Fang, and X. Jin, "Selective transfer with reinforced transfer network for partial domain adaptation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 12706–12714.

[19] X. Chen, S. Wang, M. Long, and J. Wang, "Transferability vs. discriminability: Batch spectral penalization for adversarial domain adaptation," in *Proc. Int. Conf. Mach. Learn.*, 2019, pp. 1081–1090.

[20] A. Jaiswal, A. R. Babu, M. Z. Zadeh, D. Banerjee, and F. Makedon, "A survey on contrastive self-supervised learning," *Technologies*, vol. 9, no. 1, p. 2, Dec. 2020.

[21] L. Duan, I. W. Tsang, and D. Xu, "Domain transfer multiple kernel learning," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 3, pp. 465–479, Mar. 2012.

[22] B. Sun, J. Feng, and K. Saenko, "Correlation alignment for unsupervised domain adaptation," in *Domain Adaptation in Computer Vision Applications*. Cham, Switzerland: Springer, 2017, pp. 153–171.

[23] M. Long, J. Wang, G. Ding, J. Sun, and P. S. Yu, "Transfer joint matching for unsupervised domain adaptation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2014, pp. 1410–1417.

[24] W. Zellinger, E. Lughofer, S. Saminger-Platz, T. Grubinger, and T. Natschläger, "Central moment discrepancy (CMD) for domain-invariant representation learning," in *Proc. Int. Conf. Learn. Represent.*, 2017, pp. 1–13.
[25] M. Long, J. Wang, G. Ding, J. Sun, and P. S. Yu, "Transfer feature learning with joint distribution adaptation," in *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 2013, pp. 2200–2207.
[26] J. Wang, Y. Chen, L. Hu, X. Peng, and P. S. Yu, "Stratified transfer learning for cross-domain activity recognition," in *Proc. IEEE Int. Conf. Pervasive Comput. Commun. (PerCom)*, Mar. 2018, pp. 1–10.
[27] K. Saito, Y. Ushiku, and T. Harada, "Asymmetric tri-training for unsupervised domain adaptation," in *Proc. Int. Conf. Mach. Learn.*, 2017, pp. 2988–2997.
[28] J. Wang, Y. Chen, S. Hao, W. Feng, and Z. Shen, "Balanced distribution adaptation for transfer learning," in *Proc. IEEE Int. Conf. Data Mining (ICDM)*, Nov. 2017, pp. 1129–1134.
[29] J. Tahmoresnezhad and S. Hashemi, "Visual domain adaptation via transfer feature learning," *Knowl. Inf. Syst.*, vol. 50, no. 2, pp. 585–605, Jun. 2017.
[30] J. Zhang, W. Li, and P. Ogunbona, "Joint geometrical and statistical alignment for visual domain adaptation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 1859–1867.
[31] J. Yosinski, J. Clune, Y. Bengio, and H. Lipson, "How transferable are features in deep neural networks?" in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 27, 2014, pp. 3320–3328.
[32] M. Ghifary, W. B. Kleijn, and M. Zhang, "Domain adaptive neural networks for object recognition," in *Proc. Pacific Rim Int. Conf. Artif. Intell.* Cham, Switzerland: Springer, 2014, pp. 898–904.
[33] M. Long, Y. Cao, J. Wang, and M. Jordan, "Learning transferable features with deep adaptation networks," in *Proc. Int. Conf. Mach. Learn.*, 2015, pp. 97–105.
[34] M. Long, H. Zhu, J. Wang, and M. I. Jordan, "Unsupervised domain adaptation with residual transfer networks," in *Proc. 30th Int. Conf. Neural Inf. Process. Syst.*, 2016, pp. 136–144.
[35] G. Kang, L. Jiang, Y. Yang, and A. G. Hauptmann, "Contrastive adaptation network for unsupervised domain adaptation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 4893–4902.
[36] Z. Zhang, M. Wang, Y. Huang, and A. Nehorai, "Aligning infinite-dimensional covariance matrices in reproducing kernel Hilbert spaces for domain adaptation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 3437–3445.
[37] I. J. Goodfellow *et al.*, "Generative adversarial nets," in *Proc. NIPS*, 2014, pp. 2672–2680.
[38] J. Shen, Y. Qu, W. Zhang, and Y. Yu, "Wasserstein distance guided representation learning for domain adaptation," in *Proc. AAAI Conf. Artif. Intell.*, 2018, vol. 32, no. 1, pp. 4058–4065.
[39] W. Zhang, W. Ouyang, W. Li, and D. Xu, "Collaborative and adversarial network for unsupervised domain adaptation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 3801–3809.
[40] E. Tzeng, J. Hoffman, K. Saenko, and T. Darrell, "Adversarial discriminative domain adaptation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 7167–7176.
[41] W. Hong, Z. Wang, M. Yang, and J. Yuan, "Conditional generative adversarial network for structured domain adaptation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 1335–1344.
[42] Z. Deng, Y. Luo, and J. Zhu, "Cluster alignment with a teacher for unsupervised domain adaptation," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 9944–9953.
[43] Z. Cao, K. You, M. Long, J. Wang, and Q. Yang, "Learning to transfer examples for partial domain adaptation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 2985–2994.
[44] S. Li *et al.*, "Deep residual correction network for partial domain adaptation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 43, no. 7, pp. 2329–2344, Jul. 2021.
[45] L. Li, Z. Wan, and H. He, "Dual alignment for partial domain adaptation," *IEEE Trans. Cybern.*, vol. 51, no. 7, pp. 3404–3416, Jul. 2021, doi: 10.1109/TCYB.2020.2983337.
[46] J. Chen, X. Wu, L. Duan, and S. Gao, "Domain adversarial reinforcement learning for partial domain adaptation," *IEEE Trans. Neural Netw. Learn. Syst.*, early access, Oct. 16, 2020, doi: 10.1109/TNNLS.2020.3028078.
[47] J. Liang, Y. Wang, D. Hu, R. He, and J. Feng, "A balanced and uncertainty-aware approach for partial domain adaptation," in *Proc. 16th Eur. Conf. Comput. Vis.*, Glasgow, U.K.: Springer, Aug. 2020, pp. 123–140.
[48] L. Jing and Y. Tian, "Self-supervised visual feature learning with deep neural networks: A survey," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 43, no. 11, pp. 4037–4058, Nov. 2021, doi: 10.1109/TPAMI.2020.2992393.
[49] R. D. Hjelm *et al.*, "Learning deep representations by mutual information estimation and maximization," in *Proc. Int. Conf. Learn. Represent.*, 2019, pp. 1–24. [Online]. Available: https://openreview.net/forum?id=Bklr3j0cKX
[50] P. Bachman, R. D. Hjelm, and W. Buchwalter, "Learning representations by maximizing mutual information across views," in *Proc. Adv. Neural Inf. Process. Syst.*, 2019, pp. 15509–15519.
[51] Y. Tian, D. Krishnan, and P. Isola, "Contrastive multiview coding," 2019, *arXiv:1906.05849*.
[52] K. He, H. Fan, Y. Wu, S. Xie, and R. Girshick, "Momentum contrast for unsupervised visual representation learning," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 9729–9738.
[53] T. Chen, S. Kornblith, M. Norouzi, and G. Hinton, "A simple framework for contrastive learning of visual representations," in *Proc. Int. Conf. Mach. Learn.*, 2020, pp. 1597–1607.
[54] M. Caron, I. Misra, J. Mairal, P. Goyal, P. Bojanowski, and A. Joulin, "Unsupervised learning of visual features by contrasting cluster assignments," in *Proc. 34th Conf. Neural Inf. Process. Syst. (NIPS)*, 2020, pp. 1–23.
[55] P. Khosla *et al.*, "Supervised contrastive learning," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 33, 2020, pp. 18661–18673.
[56] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," in *Proc. Conf. North Amer. Chapter Assoc. Comput. Linguistics, Hum. Lang. Technol.*, vol. 1, 2019, pp. 4171–4186.
[57] K. Saenko, B. Kulis, M. Fritz, and T. Darrell, "Adapting visual category models to new domains," in *Proc. Eur. Conf. Comput. Vis.* Berlin, Germany: Springer, 2010, pp. 213–226.
[58] B. Gong, Y. Shi, F. Sha, and K. Grauman, "Geodesic flow kernel for unsupervised domain adaptation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2012, pp. 2066–2073.
[59] H. Venkateswara, J. Eusebio, S. Chakraborty, and S. Panchanathan, "Deep hashing network for unsupervised domain adaptation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 5018–5027.
[60] G. Griffin, A. Holub, and P. Perona, "Caltech-256 object category dataset," California Inst. Technol., Pasadena, CA, USA, Tech. Rep. 7694, 2007.
[61] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 25, 2012, pp. 1097–1105.
[62] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 770–778.
[63] M. Long, Z. Cao, J. Wang, and M. I. Jordan, "Conditional adversarial domain adaptation," in *Proc. 32nd Int. Conf. Neural Inf. Process. Syst. (NIPS)*, 2018, pp. 1647–1657.
[64] J. Liang, D. Hu, and J. Feng, "Do we really need to access the source data? Source hypothesis transfer for unsupervised domain adaptation," in *Proc. Int. Conf. Mach. Learn.*, 2020, pp. 6028–6039.
[65] M. Long, H. Zhu, J. Wang, and M. I. Jordan, "Deep transfer learning with joint adaptation networks," in *Proc. Int. Conf. Mach. Learn.*, 2017, pp. 2208–2217.
[66] L. Van der Maaten and G. Hinton, "Visualizing data using t-SNE," *J. Mach. Learn. Res.*, vol. 9, no. 11, pp. 1–27, 2008.

Cuie Yang received the Ph.D. degree in control theory and control engineering from Northeastern University, Shenyang, China, in 2019.

She was a Post-Doctoral Research Fellow with the Department of Computer Science, Hong Kong Baptist University, Hong Kong. She is currently a Post-Doctoral Research Fellow with the School of Engineering and Computer Science, Victoria University of Wellington, Wellington, New Zealand. Her research interests include transfer learning, domain adaptation, concept drift learning, evolutionary transfer optimization, and data-driven evolutionary optimization.

Yiu-Ming Cheung (Fellow, IEEE) received the Ph.D. degree from the Department of Computer Science and Engineering, The Chinese University of Hong Kong, Hong Kong, in 2000.

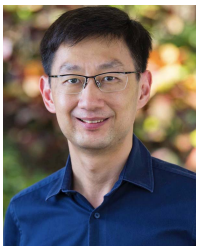
He is currently a Full Professor with the Department of Computer Science, Hong Kong Baptist University, Hong Kong. His research interests include machine learning, computer vision, pattern recognition, data mining, multi-objective optimization, and information hiding.

He is a fellow of the American Association for the Advancement of Science (AAAS), the Institution of Engineering and Technology (IET), and the British Computer Society (BCS). He serves as an Associate Editor for the IEEE TRANSACTIONS ON CYBERNETICS, the IEEE TRANSACTIONS ON EMERGING TOPICS IN COMPUTATIONAL INTELLIGENCE, the IEEE TRANSACTIONS ON COGNITIVE AND DEVELOPMENTAL SYSTEMS, the IEEE TRANSACTIONS ON NEURAL NETWORKS AND LEARNING SYSTEMS from 2014 to 2020, *Pattern Recognition*, and *Neurocomputing*. For details, please refer to: http://www.comp.hkbu.edu.hk/~ymc.

Jinliang Ding (Senior Member, IEEE) received the bachelor's, master's, and Ph.D. degrees in control theory and control engineering from Northeastern University, Shenyang, China, in 2001, 2004, and 2012, respectively.

He is currently a Professor with the State Key Laboratory of Synthetical Automation for Process Industry, Northeastern University. He has authored or coauthored over 100 refereed journal articles and refereed papers at international conferences. He has also invented or co-invented 20 patents. His current research interests include modeling, plant-wide control, optimization for complex industrial systems, stochastic distribution control, and multiobjective evolutionary algorithms and their applications.

Dr. Ding was a recipient of the three First-Prize of Science and Technology Awards of the Ministry of Education in 2006, 2012, and 2018, respectively, the International Federation of Automatic Control (IFAC) Control Engineering Practice for 2011–2013 Paper Prize, the National Technological Invention Award in 2013, the National Science Fund for Distinguished Young Scholars in 2015, and the Young Scholars Science and Technology Award of China in 2016. One of his articles published on control engineering practice was selected for the Best Paper Award from 2011 to 2013.



Kay Chen Tan (Fellow, IEEE) received the B.Eng. (Hons.) and Ph.D. degrees from the University of Glasgow, U.K., in 1994 and 1997, respectively.

He is currently a Chair Professor with the Department of Computing, The Hong Kong Polytechnic University, Hong Kong. He is an Honorary Professor with the University of Nottingham, Nottingham, U.K.

Dr. Tan is currently the Vice President (Publications) of the IEEE Computational Intelligence Society and an IEEE Distinguished Lecturer Program Speaker. He also serves as an editorial board member of more than ten journals. He was the Editor-in-Chief of the IEEE TRANSACTIONS ON EVOLUTIONARY COMPUTATION. He is the Chief Coeditor of Springer Book Series on *Machine Learning: Foundations, Methodologies, and Applications*.

Bing Xue (Senior Member, IEEE) received the B.Sc. degree from Henan University of Economics and Law, Zhengzhou, China, in 2007, the M.Sc. degree in management from Shenzhen University, Shenzhen, China, in 2010, and the Ph.D. degree in computer science from the Victoria University of Wellington (VUW), Wellington, New Zealand, in 2014.

She is currently a Professor in computer science and the Program Director of science with the School of Engineering and Computer Science, VUW. She has over 300 papers published in fully refereed international journals and conferences. Her research interests include evolutionary computation, machine learning, classification, symbolic regression, feature selection, evolving deep NNs, image analysis, transfer learning, and multi-objective machine learning.

Dr. Xue is currently the Chair of the IEEE Computational Intelligence Society (CIS) Task Force on Transfer Learning and Transfer Optimization and the Chair of the IEEE CIS Evolutionary Computation Technical Committee. She is currently the Vice Chair of the IEEE Task Force on Evolutionary Feature Selection and Construction and the IEEE CIS Task Force on Evolutionary Deep Learning and Applications. She has also served as an Associate Editor of several international journals, such as *IEEE Computational Intelligence Magazine*, the IEEE TRANSACTIONS ON EVOLUTIONARY COMPUTATION, and the IEEE TRANSACTIONS ON ARTIFICIAL INTELLIGENCE. She is currently an Editor of the IEEE CIS NEWSLETTER.

Mengjie Zhang (Fellow, IEEE) received the B.E. and M.E. degrees from the Artificial Intelligence Research Center, Agricultural University of Hebei, Hebei, China, in 1989 and 1992, respectively, and the Ph.D. degree in computer science from RMIT University, Melbourne, VIC, Australia, in 2000.

He is currently a Professor of Computer Science, the Head of the Evolutionary Computation Research Group, Victoria University of Wellington, Wellington, New Zealand, where he is also the Associate Dean (Research and Innovation) with the Faculty of Engineering. He has authored or coauthored over 700 papers in refereed international journals and conferences. His current research interests include evolutionary computation, with emphasis on genetic programming and particle swarm optimization with application areas of image analysis, multi-objective optimization, feature selection and reduction, job shop scheduling, and transfer learning.

Prof. Zhang is a fellow of the Royal Society of New Zealand. He is an IEEE Computational Intelligence Society (CIS) Distinguished Lecturer. He has been a Panel Member of the Marsden Fund (New Zealand Government Funding). He is also a Committee Member of the IEEE NZ Central Section. He was the Chair of the IEEE CIS Intelligent Systems and Applications Technical Committee, the IEEE CIS Emergent Technologies Technical Committee, and the Evolutionary Computation Technical Committee. He was a member of the IEEE CIS Award Committee. He is the Vice Chair of the IEEE CIS Task Force on Evolutionary Feature Selection and Construction and the Task Force on Evolutionary Computer Vision and Image Processing. He is the Founding Chair of the IEEE Computational Intelligence Chapter in New Zealand.