

Prototype Combination for Multi-Source Unsupervised Domain Adaptation

Min Huang[✉], Zifeng Xie[✉], *Student Member, IEEE*, Han Huang[✉], *Senior Member, IEEE*, Chang Zhang[✉],
Liuqi Zhao[✉], and Ziyang Feng[✉]

Abstract—Multi-source unsupervised domain adaptation (MSUDA) is a technique that transfers knowledge from multiple labeled source domains to an unlabeled target domain. The challenge of MSUDA is to reduce the domain shift and effectively amalgamate knowledge from disparate source domains. To address this challenge, it is necessary to model the target domain as a weighted combination of the source domains at the category level. Therefore, we propose a prototype combination method for multi-source unsupervised domain adaptation, which establishes multiple domain alignment in a combinatorial manner. Our method is established on a set of semantic category prototypes, each of which is a representative category embedding. A prototype combination mechanism (i.e., a feature-fusion scheme) is designed to select which source class features should be aligned with the corresponding target class features. This method incorporates contrastive prototype adaptation (i.e., a category-wise alignment approach) to accommodate the label distributions of the target domain. Furthermore, a prototype combination regularization (i.e., a domain-wise alignment metric) is designed to reduce the distributional differences between the source category prototypes and the target samples of low-quality pseudo-labels. The experimental results on three benchmark datasets demonstrate that our prototype combination mechanism is capable of selecting and combining category-discriminative features across multiple

source domains, while the prototype combination regularization can further reduce the domain shift.

Index Terms—Multi-source unsupervised domain adaptation, pseudo-inverse matrix, prototype learning, contrastive learning.

I. INTRODUCTION

TRADITIONAL machine learning algorithms are fundamentally reliant on a substantial amount of labeled data to train the learning model effectively [1], [2], [3], [4], [5]. These algorithms operate under the assumption that the training data and the test data follow an identical feature distribution pattern [6]. Nevertheless, training data is invariably limited, while test data is unlabeled and unknown. The domain shift [7], [8] between a labeled training dataset (i.e., source domain) and another unlabeled testing dataset (i.e., target domain) can lead to severe performance degradation.

Domain adaptation is a domain-shift mitigation technique that adapts a model trained on a labeled source domain to an unlabeled target domain. This technique has been widely adopted across various fields, including computer vision [9], [10], natural language processing [11], [12], [13], and health care [14], [15], [16]. Currently, numerous single-source unsupervised domain adaptation (SUDA) methods [17], [18], [19] focus on transferring valuable knowledge from a labeled domain to another unlabeled domain. However, it is common that the labeled data are collected from multiple available source domains (e.g., data from different scenarios and devices). It is unreasonable to assume that all source domains follow the same distribution pattern [20]. Therefore, multi-source unsupervised domain adaptation (MSUDA) was developed to integrate the transferrable knowledge from multiple source domains to enhance the adaptation to a target domain. The simplest MSUDA approach is to treat the multi-source domains as one single source domain, followed by applying single-source domain adaptation methods to align distributions. Nonetheless, single-source domain adaptation methods are prone to overlooking distributional differences across different source domains, which impedes their ability to minimize the domain shift among the source domains [21].

With the advancement of deep learning, recent MSUDA efforts have concentrated on distributional alignment across multiple source domains and a target domain. Based on the fineness of alignment, these efforts can be primarily categorized into a category-wise alignment group [10], [22], [23] and a

Received 22 November 2024; revised 12 January 2025; accepted 28 April 2025. This work was supported in part by the Natural Science Foundation of Guangdong Province under Grant 2022A1515011370, in part by the Natural Science Foundation of China under Grant 62276103, in part by the Innovation Team Project of General Colleges and Universities in Guangdong Province under Grant 2023KCXTD002, in part by the Research and Development Project on Key Technologies for Intelligent Sensing and Analysis of Urban Events Based on Low-Altitude Drones under Grant 2024BQ010011, in part by the Philosophy and Social Science Project of Guangdong Province under Grant GD23SJJZ09, and in part by the Fundamental Research Funds for the Central Universities JLU under Grant 93K172024K24. (Corresponding author: Han Huang.)

Min Huang, Zifeng Xie, and Chang Zhang are with the School of Software Engineering, South China University of Technology, Guangzhou 510006, China (e-mail: minh@scut.edu.cn; 202220145216@mail.scut.edu.cn; sezhang_2020@mail.scut.edu.cn).

Han Huang is with the School of Software Engineering, South China University of Technology, Guangzhou 510006, China, also with the Key Laboratory of Big Data and Intelligent Robot (SCUT), MOE of China, Guangzhou 510006, China, also with the Guangdong Engineering Center for Large Model and GenAI Technology, Guangzhou 510006, China, and also with the Key Laboratory of Symbolic Computation and Knowledge Engineering of Ministry of Education, Jilin University, Changchun 130012, China (e-mail: hhan@scut.edu.cn).

Liuqi Zhao and Ziyang Feng are with the Operation and Maintenance Center of Information and Communication, CSG EHV Power Transmission Company, Guangzhou 510663, China (e-mail: zhaoliuqi@ehv.csg.cn; fengziyan@ehv.csg.cn).

This article has supplementary downloadable material available at <https://doi.org/10.1109/TETCI.2025.3572133>, provided by the authors.

Recommended for acceptance by L. Feng.

Digital Object Identifier 10.1109/TETCI.2025.3572133

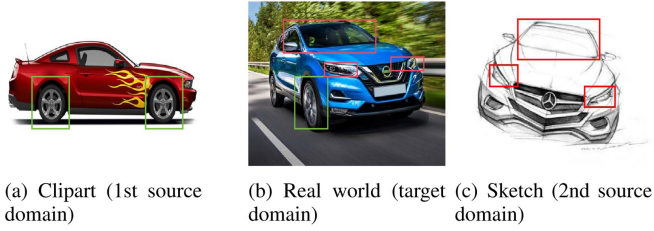


Fig. 1. Car images in different domains. Car features of the real-world image can be assembled from car features in Clipart and Sketch domains.

domain-wise alignment group [24], [25], [26]. Despite these advancements, most existing MSUDA methods face three primary problems. Firstly, they treat multiple source domains as equal [10], [24], [27], disregarding the varying contributions that each source domain may provide to the adaptation task. Such misconduct can lead to negative transfer [28] when some source domains diverge significantly from others. Secondly, domain-wise alignment methods [25], [26], [29] do not explore the domain alignment at the category or pixel levels, which may result in suboptimal performance. Thirdly, prior works on category-wise domain alignment [30], [31] omit the reliability of the pseudo-labels, causing misclassification. Recent works on category-wise domain alignment [10], [23] suggest that the model classification suffers from the inclusion of noisy pseudo-labeled target samples [28]. These approaches focus on extracting knowledge from high-confidence pseudo-labeled target samples, omitting the low-confidence ones. This results in suboptimal utilization of the training data.

Based on the three problems identified, we can draw the following two points. Firstly, it is essential to model the target domain as a weighted combination of multiple source domains in a combinatorial manner. The target category features can be represented as the combination of the corresponding multiple source category features. For example, as shown in Fig. 1, the feature (e.g., “windshield,” “automobile lamp,” and “wheel” features) of the real-world “car” image can be assembled from the feature (e.g., the “wheel” feature) of the clipart “car” image and the feature (e.g., “windshield” and “automobile lamp” features) of the sketch “car” image. We deem that the target domain features can benefit from the complementary information of the source domains, with each source domain contributing unique aspects to the overall target representation. Therefore, a highly effective prototype combination mechanism (i.e., a feature-fusion scheme) is necessary for such modeling. Secondly, category-wise and domain-wise domain alignment methods can be employed in conjunction with one another. On the one hand, category-wise alignment methods delve more deeply into category semantic information than domain-wise alignment. However, their efficacy is contingent upon the availability of target pseudo-labels. On the other hand, domain-wise alignment methods are unable to alleviate the intra-class differences among all domains, but their adaptation capacity is not contingent upon the target pseudo-labels. Therefore, it is necessary to design a method to resolve the diverse confidence pseudo-labeled target samples through the synergistic effect between category-wise alignment and domain-wise alignment.

In light of the above two points, we propose a prototype combination method for multi-source domain adaptation (PCMDA), which adaptively combines the source category-discriminative features so as to achieve the alignment between a target domain and multiple source domains. Firstly, a prototype combination mechanism governs the weight distribution to ensure the prototype combinations (i.e., the weighted combinations of the source prototypes) closely match their corresponding target prototypes. Through dynamic weight allocation, the model prioritizes the source prototypes that resemble the corresponding target prototype, thus extracting more discriminative features that enhance the domain adaptation. Secondly, the domain adaptation process comprises both category-wise and domain-wise alignments. At the category level, we adopt a contrastive prototype adaptation framework (i.e., a category-wise alignment framework). In this framework, each high-confidence pseudo-labeled target embedding is aligned with its corresponding prototype combination, while each source feature embedding is aligned with its corresponding target prototype. At the domain level, a prototype combination regularization (i.e., a domain-wise alignment metric) is designed to achieve the distribution alignment between the low-quality pseudo-labeled target samples and the prototype combinations. In the following sections, we will discuss related work and the PCMDA method and assess the effectiveness of PCMDA from both theoretical and experimental perspectives.

The main contributions of this paper are summarized as follows:

- We propose a prototype combination method for multi-source unsupervised domain adaptation (PCMDA) that models the target domain as a weighted combination of the source domains in a combinatorial manner. A prototype combination mechanism is designed for PCMDA to combine the source semantic information that is beneficial for cross-domain image classification. Due to the prototype combination mechanism, the PCMDA model can discriminate and combine source prototypes that are similar to the target prototype at the class level, realizing more effective domain adaptation compared to the existing method.
- Our PCMDA model can adaptively learn the interdependencies among the category-discriminative features of different domains through a prototype combination mechanism. Additionally, a prototype combination regularization is introduced to the PCMDA model for the distribution alignment of the low-quality pseudo-labeled target samples, which further reduces the domain shift between source and target domains. Compared to the existing class-alignment methods, the PCMDA model can learn knowledge from the low-quality pseudo-labeled target samples due to the prototype combination regularization. Therefore, the PCMDA model achieves competitive performance on Digits-5, PACS, and Office_caltech_10 with average accuracies of 93.2%, 89.3%, and 97.4%, respectively.

II. RELATED WORK

This section reviews related work in three aspects: single-source unsupervised domain adaptation, multi-source unsupervised domain adaptation, and instance-wise contrastive learning.

A. Single-Source Unsupervised Domain Adaptation (SUDA)

Recently, the field of SUDA has experienced a plethora of remarkable advancements. Existing SUDA methods can be divided into three categories. One is instance-based work [32], [33], which assigns high weights to the source samples that closely resemble the target domain samples and low weights to source samples that are very different from the target domain samples. The patch-mix transformer (PMTrans) [33] is a representative instance-based approach. Its model divides the images of all domains into several small blocks and recombines them to create new images, realizing domain adaptation. Another branch of SUDA is adversarial-based work [17], [18], [34], in which a discriminator network is trained to align the source domain with the target domain. The conditional adversarial domain adaptation (CDAN) [17], [35] framework provides a principled approach that leverages conditional adversarial networks to learn decoupled and transferable representations. The third category is feature-based work [19], [36], [37], whose core idea is to achieve the feature distributional alignment between the source and target domains in a common feature space. An example of a feature-based approach is structurally regularized deep clustering (SRDC) [37], which combines clustering and classifier learning to achieve unsupervised domain adaptation.

To achieve a fine-grained alignment, some existing works [30], [31] focus on category-wise distributional alignment. Nonetheless, these works cannot solve the MSUDA problem. To address this problem, we designed a prototype combination mechanism to combine the category-discriminative semantic information from multiple source domains.

B. Multi-Source Unsupervised Domain Adaptation (MSUDA)

SUDA methods mainly focus on transferring valuable knowledge from one source domain to a target domain. However, it is worth noting that available labeled data may come from multiple source domains. Therefore, multi-source unsupervised domain adaptation (MSUDA) was proposed to extend domain adaptation techniques to multi-source scenarios. Early efforts [38], [39] provide fundamental theoretical research for the hypothesis that the target distribution could be represented as a weighted combination of several source distributions. Furthermore, significant efforts [38], [40] have been allocated to theoretical research on the generalized boundary of MSUDA. Recently, Xu et al. [41] combined the source-specific perplexity scores to denote the possibilities of target samples. Similarly, Peng et al. [25] propose a multi-classifier approach to dynamically match the feature distribution moment for each source-target domain pair. Besides, the mutual learning-based alignment network (MLAN) [29] is designed based on model distillation, which promotes domain alignment by learning from each other through joint alignment branches and separate alignment branches. Wang et al. [10] leverage category prototypes from each source domain to construct a knowledge graph, aiming to maintain the invariance of inter-class relationships for domain alignment.

Building upon the theoretical guarantee of weighted domain combination [38], [39], manual weighted combination schemes have been widely adopted in recent efforts [24], [25], [27], [29]. Nevertheless, such combination schemes can neither adapt

to each source domain with the model training nor combine multiple source domains accurately. In contrast, our PCMDA model can achieve dynamic weight allocation for each prototype through our prototype combination mechanism.

C. Instance-Wise Contrastive Learning

Contrastive learning [42], [43] is an unsupervised learning method used for representation learning. Its objective is to learn data representations such that similar instances are brought closer together in the latent representation space while dissimilar instances are pushed farther apart. Numerous contrastive losses have been designed to guide the model in learning contrastive representations. One is triplet loss [44], which is designed to minimize the distance between an anchor sample and a positive example while maximizing the distance between the anchor sample and a negative example. It is commonly employed in tasks with roughly equal sample proportions, such as face recognition and fine-grained classification tasks. In most cases, the number of negative examples is larger than that of positive ones, which leads to the information noise contrastive estimation (InfoNCE) loss [43] becoming mainstream in contrastive learning. Building upon the InfoNCE loss, Zhang et al. [30] introduced a prototype noise contrastive estimation (ProtoNCE) loss, which serves as a generalized version of the InfoNCE loss.

Based on the work of [30], [43], [45], we adopt a contrastive prototype adaptation framework to achieve alignment between the source and target domains at the category level. In this framework, the source feature embeddings are facilitated to match their corresponding target prototypes, while the target feature embeddings are facilitated to match their corresponding prototype combinations.

III. PROTOTYPE COMBINATION FOR MULTI-SOURCE UNSUPERVISED DOMAIN ADAPTATION

This section introduces our PCMDA method. First, we formulate the MSUDA problem. Second, we present the overall scheme of PCMDA. Third, we introduce how to estimate prototypes for each category in all domains. Fourth, we introduce our prototype combination mechanism to investigate how to combine the category-discriminative semantic information. Fifth, we describe the key components of our learning objectives: contrastive prototype adaptation, prototype combination regularization, and classification constraints. Finally, we provide the error-bound analysis of PCMDA.

A. Problem Definition

In the MSUDA setting, it is assumed that the training data have been collected from multiple different domains, with each domain exhibiting a distinct distribution. Let $D_s = \{D_{si} | i = 1, 2, \dots, N\}$ denote a collection of N different labeled source domains. The source domain $D_{si} = \{(X_{si}^{(j)}, y_{si}^{(j)}) | j = 1, 2, \dots, m_{si}\}$ contains m_{si} samples where $y_{si}^{(j)} \in \{1, 2, \dots, K\}$ (K is the number of categories) is the corresponding ground-truth labels. Meanwhile, the target domain $D_T = \{X_T^{(j)} | j = 1, 2, \dots, m_T\}$ contains m_T samples without available labels. It is worth noting that the target domain has

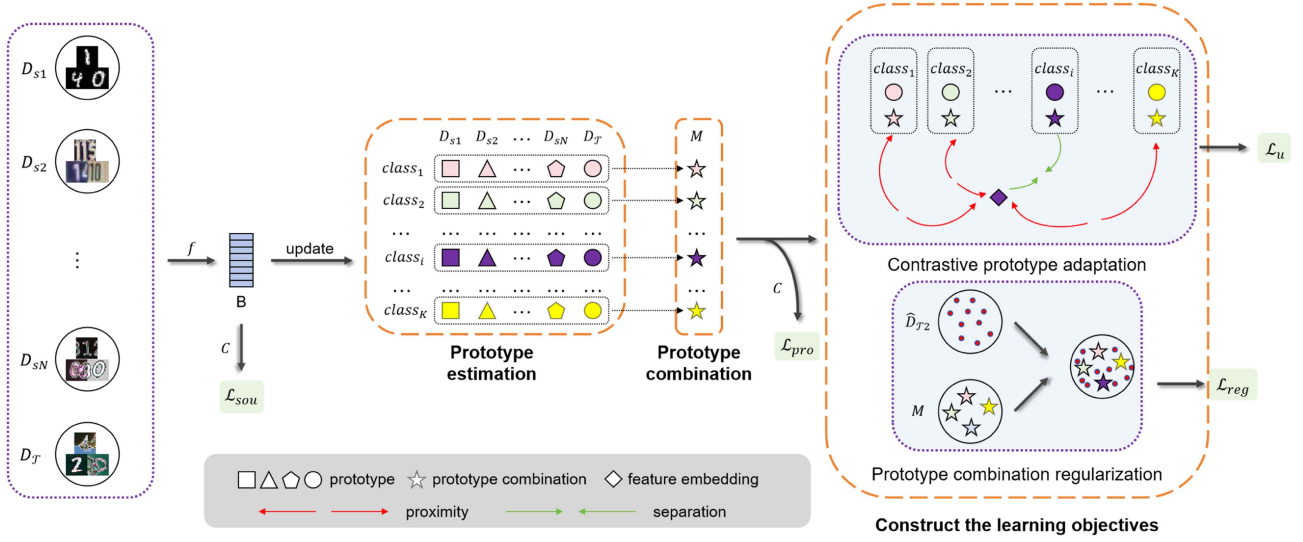


Fig. 2. The architecture of our method PCMDA. PCMDA comprises three stages: (i) The source and target prototypes are estimated to represent the category features in all domains by a randomly sampled mini-batch B . (ii) The PCMDA model adaptively assigns the weights to the source prototypes and then derives the source prototype combinations for each category. (iii) Through \mathcal{L}_u , each high-confidence pseudo-labeled target feature embedding is aligned to its corresponding prototype combination, while each source feature embedding is aligned to its corresponding target prototype. Meanwhile, the prototype combination regularization enhances the distributional alignment between the prototype combinations and low-confidence pseudo-labeled target samples.

no labels to support the model training. Additionally, we estimate a prototype for each category in all domains. The source prototypes are denoted as $\mathbb{C}_s = \{u_{si}^{(k)} | i = 1, 2, \dots, N; k = 1, 2, \dots, K\}$ while the target prototypes are denoted as $\mathbb{C}_T = \{u_T^{(k)} | k = 1, 2, \dots, K\}$.

The core topic of MSUDA is how to transfer knowledge from multiple source domains to a target domain. It is challenging due to the distributional difference among all domains and the lack of target labels [46]. Conventional SUDA methods cannot tackle the multiple source domain issue. To address this issue, we propose a prototype combination method for multi-source domain adaptation (PCMDA) to model the target domain as a weighted combination of the source domains in a combinatorial manner.

B. Overall Scheme

As illustrated in Fig. 2, PCMDA comprises three stages: prototype estimation, prototype combination, and learning objectives.

In Stage 1 (Section III-C), the PCMDA model generates feature prototypes for each category in all domains. Specifically, estimations of the embedding mean are used for prototype estimation. To mitigate the problem of estimation bias, we introduce an exponential moving average scheme to promote smoother updates of the prototypes. Such a scheme has been widely used in [10], [30], [43].

In Stage 2 (Section III-D), each target prototype is modeled as a weighted combination of the corresponding source prototypes. There exists a problem of finding the optimal weights for the weighted combination of source prototypes, which can be transformed into a problem of finding an optimal solution for the linear equation system. Inspired by pseudo-inverse matrix theory, we design a prototype combination mechanism to calculate the

unique set of weights such that the Euclidean distance between the source prototype combination and the corresponding target prototype is minimal. According to the weights, the optimal prototype combinations can be obtained.

In Stage 3 (Section III-E), a contrastive prototype adaptation framework and a prototype combination regularization are designed for category-wise and domain-wise alignments, respectively. Through the contrastive prototype adaptation framework, the source feature embeddings are facilitated to match their corresponding target prototypes. Simultaneously, the target feature embeddings of high-quality pseudo-labels are promoted to match their corresponding prototype combinations. Since the pseudo-labels of some target samples are of low quality, the contrastive prototype adaptation framework cannot be applied to these samples. As a supplement to the framework, prototype combination regularization is employed to achieve the distributional alignment between these samples and all prototype combinations.

C. Prototype Estimation

To describe the category features, the PCMDA model learns the prototype of each category in all domains by a randomly sampled mini-batch. Specifically, for each iteration, a randomly sampled mini-batch $B = \{\hat{D}_{s1}, \hat{D}_{s2}, \dots, \hat{D}_{sN}, \hat{D}_T\}$ is mapped to the low-dimensional embedding space by a feature extractor $f(\cdot)$. To convey the high-level description, the source estimated prototype $\hat{u}_{si}^{(k)}$ is defined as the mean of all embeddings belonging to the k -th category in \hat{D}_{si} :

$$\hat{u}_{si}^{(k)} = \frac{1}{|\hat{D}_{si}^{(k)}|} \sum_{X_{si} \in \hat{D}_{si}^{(k)}} f(X_{si}), \quad (1)$$

where $\hat{D}_{si}^{(k)}$ is the set of the samples belonging to the k -th category in \hat{D}_{si} , and $f(\cdot)$ stands for the feature extractor that maps images to feature embedding vectors.

Since the target labels are unavailable, we adopt a pseudo-labeling strategy proposed by [22] to generate pseudo-labels for each target sample. Following the pseudo-labeling strategy, $\hat{D}_{\mathcal{T}}$ can be divided into $\hat{D}_{\mathcal{T}1}$ and $\hat{D}_{\mathcal{T}2}$, where the samples of $\hat{D}_{\mathcal{T}1}$ are of high-confidence pseudo-labels, while the samples of $\hat{D}_{\mathcal{T}2}$ are of low-confidence pseudo-labels. The samples of $\hat{D}_{\mathcal{T}1}$ are utilized to estimate target prototypes. Since the pseudo-labels of $\hat{D}_{\mathcal{T}2}$ are unavailable, a prototype combination regularization is designed to learn the domain-invariance of the samples of $\hat{D}_{\mathcal{T}2}$. The prototype combination regularization is detailed in Section III-E1. Therefore, the target estimated prototype $\hat{u}_{\mathcal{T}}^{(k)}$ is defined as the mean of all embeddings belonging to the k -th category in $\hat{D}_{\mathcal{T}1}$:

$$\hat{u}_{\mathcal{T}}^{(k)} = \frac{1}{|\hat{D}_{\mathcal{T}1}^{(k)}|} \sum_{X_{\mathcal{T}} \in \hat{D}_{\mathcal{T}1}^{(k)}} f(X_{\mathcal{T}}), \quad (2)$$

where $\hat{D}_{\mathcal{T}1}^{(k)}$ is the set of all samples belonging to the k -th category in $\hat{D}_{\mathcal{T}1}$.

However, such a prototype estimation belongs to the random mini-batch sampling estimation, which introduces estimation bias into the model. To reduce the estimation bias, an exponential moving average scheme is introduced to estimate the prototypes. In this way, the prototypes evolve more continuously than the estimated prototypes. Specifically, in each iteration, we update the prototypes as follows.

$$u_{si}^{(k)} \leftarrow \beta u_{si}^{(k)} + (1 - \beta) \hat{u}_{si}^{(k)}, \quad i = 1, 2, \dots, N \quad (3)$$

$$u_{\mathcal{T}}^{(k)} \leftarrow \beta u_{\mathcal{T}}^{(k)} + (1 - \beta) \hat{u}_{\mathcal{T}}^{(k)}, \quad (4)$$

where β is a momentum coefficient that is fixed as 0.7 in all experiments. Similar schemes have been broadly used in [10], [30], [43] to stabilize the learning process by promoting smoother updates of variables.

D. Prototype Combination

To combine the category-discriminative source semantic information, we design a prototype combination mechanism to govern the prototype weight distribution. The details of the prototype combination mechanism are as follows.

Given N prototypes $p^{(k)} = [u_{s1}^{(k)}, u_{s2}^{(k)}, \dots, u_{sN}^{(k)}]$ belonging to the k -th category, one from each source domain, we postulate that there exists a group of weights $w^{(k)} = [g_1, g_2, \dots, g_N]^T$ such that $w^{(k)} = \arg \min_{e \in \mathbb{R}^N} \|p^{(k)} \cdot e - u_{\mathcal{T}}^{(k)}\|^2$. It is worth mentioning that all prototypes and embeddings are column vectors, that $w^{(k)}$ is the optimal weight column vector of the source prototypes belonging to the k -th category, and that $u_{\mathcal{T}}^{(k)}$ is the target prototype belonging to the k -th category. The core problem is how to find $w^{(k)}$ based on $p^{(k)} \in \mathbb{R}^{d \times N}$ (i.e., d and N represent the dimension of feature embedding and the number of source domains, respectively.) and $u_{\mathcal{T}}^{(k)}$. It is worth noting that d is greater than N , that $p^{(k)}$ is a matrix, and that both $w^{(k)}$ and $u_{\mathcal{T}}^{(k)}$

are column vectors. Therefore, the optimal weight assignment to each source prototype can be transformed into the problem of finding an approximate solution for a linear equation system.

Theorem 1. (Optimal weights of prototype combination): For $p^{(k)} \in \mathbb{R}^{d \times N}$ and $u_{\mathcal{T}}^{(k)} \in \mathbb{R}^d$ where $d > N$, the set of optimal weights can be represented as

$$w^{(k)} = ((p^{(k)})^T (p^{(k)}))^{-1} (p^{(k)})^T u_{\mathcal{T}}^{(k)} \quad (5)$$

such that $w^{(k)} = \arg \min_{e \in \mathbb{R}^N} \|p^{(k)} \cdot e - u_{\mathcal{T}}^{(k)}\|^2$.

Proof: We construct a function $h(e) = \|p^{(k)} \cdot e - u_{\mathcal{T}}^{(k)}\|^2$. When the function $h(e)$ reaches a minimum value, $h'(e)$ (the derivative of $h(e)$ with respect to e) equals zero. To get the minimum point, the following equation must be satisfied.

$$\frac{dh(e)}{de} = 2(p^{(k)})^T (p^{(k)} \cdot e - u_{\mathcal{T}}^{(k)}) = 0. \quad (6)$$

Upon simplifying the (6), we can derive the following equation.

$$(p^{(k)})^T p^{(k)} e = (p^{(k)})^T u_{\mathcal{T}}^{(k)}. \quad (7)$$

Since $d > N$, $r(p^{(k)}) = r((p^{(k)})^T p^{(k)}) = N$ where $r(\cdot)$ is the matrix rank function. $((p^{(k)})^T p^{(k)})$ is a square matrix of full rank, so $((p^{(k)})^T p^{(k)})^{-1}$ exists. It's worth noting that $((p^{(k)})^T p^{(k)})^{-1} (p^{(k)})^T$ is the pseudo-inverse matrix of $p^{(k)}$. Upon multiplying both sides of (7) by $((p^{(k)})^T p^{(k)})^{-1}$ simultaneously, we obtain the following equation.

$$((p^{(k)})^T p^{(k)})^{-1} (p^{(k)})^T p^{(k)} e = ((p^{(k)})^T p^{(k)})^{-1} (p^{(k)})^T u_{\mathcal{T}}^{(k)}. \quad (8)$$

Therefore, the minimum point of $h(e)$ can be obtained according to (9).

$$e = ((p^{(k)})^T p^{(k)})^{-1} (p^{(k)})^T u_{\mathcal{T}}^{(k)}. \quad (9)$$

According to Theorem 1, the source prototype combinations can be obtained by the appropriate weights:

$$\begin{aligned} M &= \{b^{(1)}, b^{(2)}, \dots, b^{(K)}\} \\ &= \{p^{(1)} \cdot w^{(1)}, p^{(2)} \cdot w^{(2)}, \dots, p^{(K)} \cdot w^{(K)}\}, \end{aligned} \quad (10)$$

where $b^{(k)}$ is the prototype combination that is the weighted combination of source prototypes belonging to the k -th category.

E. Learning Objectives

In this part, two learning objectives are constructed for model training. The first objective is to train the PCMDA model to achieve domain adaptation at the category and domain levels, respectively. The second objective is to enhance the feature discriminability of the PCMDA model. To pursue these learning objectives, the PCMDA model is optimized by three types of losses, including a contrastive loss $\mathcal{L}_u(f)$, a prototype combination regularization loss $\mathcal{L}_{reg}(M, \hat{D}_{\mathcal{T}2}; f)$, and a classification loss $\mathcal{L}_{cls}(C, f)$.

1) *Contrastive Prototype Adaptation:* To enhance category-wise alignment, a contrastive prototype adaptation framework

is designed for high-confidence pseudo-labeled target samples. The details of the framework are as follows.

With the available labels, the source feature embeddings are facilitated to match their corresponding target prototypes as closely as possible. Simultaneously, the feature embeddings of \hat{D}_{T_1} are facilitated to match their corresponding prototype combinations as closely as possible. Additionally, we enhance the compactness of category features by forcing the feature embeddings to match their corresponding prototypes, which drives the model to extract more discriminative features among different categories. Last but not least, the target prototypes are facilitated to match their corresponding prototype combinations as closely as possible. Under the guidance of these ideas, we design a contrastive loss $\mathcal{L}_u(f, C)$ which comprises a compactness loss $\mathcal{L}_{com}(f)$, an alignment loss $\mathcal{L}_{ali}(f)$ and a prototype alignment loss $\mathcal{L}_{pa}(M, \mathbb{C}_T)$:

$$\mathcal{L}_u(f) = \mathcal{L}_{com}(f) + \mathcal{L}_{ali}(f) + \mathcal{L}_{pa}(M, \mathbb{C}_T). \quad (11)$$

$\mathcal{L}_{com}(f)$ is designed to facilitate the feature compactness, as defined in (12). It comprises a source compactness loss $\mathcal{L}_1(\hat{D}_s, M; f)$ (13) and a target compactness loss $\mathcal{L}_1(\hat{D}_{T_1}, \mathbb{C}_T; f)$ (14). The source feature embeddings are facilitated to match their corresponding source prototype combinations via the source compactness loss $\mathcal{L}_1(\hat{D}_s, M; f)$, while the target feature embeddings of \hat{D}_{T_1} are facilitated to match their corresponding target prototypes via the target compactness loss $\mathcal{L}_1(\hat{D}_{T_1}, \mathbb{C}_T; f)$.

$$\mathcal{L}_{com}(f) = \mathcal{L}_1(\hat{D}_s, M; f) + \mathcal{L}_1(\hat{D}_{T_1}, \mathbb{C}_T; f). \quad (12)$$

$$\mathcal{L}_1(\hat{D}_s, M; f) = - \sum_{k=1}^K \sum_{i=1}^N \frac{1}{|\hat{D}_{si}^{(k)}|} \sum_{X_{si} \in \hat{D}_{si}^{(k)}} \ln P(b^{(k)} | X_{si}; M). \quad (13)$$

$$\mathcal{L}_1(\hat{D}_{T_1}, \mathbb{C}_T; f) = - \sum_{k=1}^K \frac{1}{|\hat{D}_{T_1}^{(k)}|} \sum_{X_T \in \hat{D}_{T_1}^{(k)}} \ln P(u_T^{(k)} | X_T^{(j)}; \mathbb{C}_T). \quad (14)$$

where $\hat{D}_{si}^{(k)}$ is the set of all samples belonging to the k -th category in \hat{D}_{si} , $\hat{D}_s = \{\hat{D}_{s1}, \hat{D}_{s2}, \dots, \hat{D}_{sN}\}$, $M = \{b^{(1)}, b^{(2)}, \dots, b^{(K)}\}$, $\mathbb{C}_T = \{u_T^{(1)}, u_T^{(2)}, \dots, u_T^{(K)}\}$, and $\hat{D}_{T_1}^{(k)}$ is the set of target samples belonging to the k -th category in \hat{D}_{T_1} . The probability $P(v|x; \Omega)$ that an image sample x belongs to the same category as the prototype v ($v \in \Omega$) is derived from a contrastive loss [43]:

$$P(v|x; \Omega) = \frac{\exp(\langle f(x)^T, v \rangle / \tau)}{\sum_{h \in \Omega} \exp(\langle f(X)^T, h \rangle / \tau)}, \quad (15)$$

where $\langle \cdot, \cdot \rangle$ is the cosine similarity function, and τ is a temperature parameter.

$\mathcal{L}_{ali}(f)$ is designed to enhance the category feature alignment. It comprises a source-target alignment loss $\mathcal{L}_2(\hat{D}_s, \mathbb{C}_T; f)$ and a target-source alignment loss $\mathcal{L}_2(\hat{D}_{T_1}, M; f)$. Each source feature embedding is facilitated to match its corresponding target prototypes via $\mathcal{L}_2(\hat{D}_s, \mathbb{C}_T; f)$ (16), while each feature embedding of \hat{D}_{T_1} is facilitated to match its corresponding prototype

combination via $\mathcal{L}_2(\hat{D}_{T_1}, M; f)$ (17).

$$\mathcal{L}_2(\hat{D}_s, \mathbb{C}_T; f) = - \sum_{k=1}^K \sum_{i=1}^N \frac{1}{|\hat{D}_{si}^{(k)}|} \sum_{X_{si} \in \hat{D}_{si}^{(k)}} \ln P(u_T^{(k)} | X_{si}; \mathbb{C}_T), \quad (16)$$

$$\mathcal{L}_2(\hat{D}_{T_1}, M; f) = - \sum_{k=1}^K \frac{1}{|\hat{D}_{T_1}^{(k)}|} \sum_{X_T \in \hat{D}_{T_1}^{(k)}} \ln P(b^{(k)} | X_T; M), \quad (17)$$

$$\mathcal{L}_{ali}(f) = \mathcal{L}_2(\hat{D}_s, \mathbb{C}_T; f) + \mathcal{L}_2(\hat{D}_{T_1}, M; f). \quad (18)$$

Besides pursuing feature alignment, we seek to enhance the alignment between the target prototypes and their corresponding prototype combinations.

$$\mathcal{L}_{pa}(M, \mathbb{C}_T) = - \frac{1}{K} \sum_{k=1}^K \ln \frac{\exp(\langle (u_T^{(k)})^T, b^{(k)} \rangle / \tau)}{\sum_{k=1}^K \exp(\langle (u_T^{(k)})^T, b^{(h)} \rangle / \tau)}. \quad (19)$$

2) *Prototype Combination Regularization*: Existing category-wise alignment methods suffer from the noisy pseudo-label overfitting problem [9]. Since the samples in \hat{D}_{T_2} are of low-quality pseudo-labels, these samples would result in negative transfer to the contrastive prototype adaptation framework (i.e., category-wise alignment framework). Inspired by the maximum mean discrepancy [47], we design a prototype combination regularization (i.e., a domain-wise alignment method.) to estimate the distributional discrepancy between these samples and all prototype combinations. Our prototype combination regularization loss $\mathcal{L}_{reg}(M, \hat{D}_{T_2}; f)$ is reformulated as:

$$\mathcal{L}_{reg}(M, \hat{D}_{T_2}; f) = \left\| \mathbb{E}_{b^{(k)} \in M} \phi(b^{(k)}) - \mathbb{E}_{X_T \in \hat{D}_{T_2}} \phi(f(X_T)) \right\|^2, \quad (20)$$

where $\phi(\cdot)$ is the gaussian kernel function.

3) *Classification Constraints*: To formulate supervision signals, we design two classification losses over the standard cross-entropy loss. One is a source feature classification loss $\mathcal{L}_{sou}(\hat{D}_s; C, f)$. Another is the prototype classification loss $\mathcal{L}_{pro}(C, f)$.

Due to the source ground-truth labels, the samples of \hat{D}_s are evaluated as follows:

$$\mathcal{L}_{sou}(\hat{D}_s; C, f) = \frac{1}{N} \sum_{i=1}^N \mathbb{E}_{(X_{si}, Y_{si}) \in \hat{D}_{si}} J(C(f(X_{si}), Y_{si})), \quad (21)$$

where $J(\cdot, \cdot)$ is the cross-entropy loss function, $f(\cdot)$ is the feature extractor, and $C(\cdot)$ is the classifier.

Both the prototypes and the prototype combinations are labeled by their corresponding categories, which leads to the prototype classification loss $\mathcal{L}_{pro}(C, f)$:

$$\mathcal{L}_{pro}(C, f) = \frac{1}{NK} \sum_{i=1}^N \sum_{k=1}^K J(C(u_{si}^{(k)}), k) + \frac{1}{K} \sum_{k=1}^K J(C(u_T^{(k)}), k)$$

$$+ \frac{1}{K} \sum_{k=1}^K J(C(b^{(k)}), k). \quad (22)$$

To enhance the feature discriminability of the PCMDA model, the total classification loss function is defined as:

$$\mathcal{L}_{cls}(C, f) = \mathcal{L}_{sou}(\hat{D}_s; C, f) + \mathcal{L}_{pro}(C, f). \quad (23)$$

As mentioned above, the overall optimization objectives for the feature extractor f and the classifier C are as follows.

$$\min_f \mathcal{L}_{cls}(C, f) + \alpha \mathcal{L}_{reg}(M, \hat{D}_{T2}; f) + \gamma \mathcal{L}_u(f), \quad (24)$$

$$\min_C \mathcal{L}_{cls}(C, f), \quad (25)$$

where α and γ are trade-off parameters.

To describe the working principle of our method, we summarize the training process of PCMDA in Algorithm 1.

F. Error Bound Analysis

We present a theoretical analysis of the classic generalization bound. Let \mathbb{H} be a hypothesis space. According to the assumption in Theorem 2 proved in [38], $\forall h \in \mathcal{H}$, the expected error of \hat{D}_T is bounded as:

$$\epsilon_T(h) \leq \epsilon_s(h) + \frac{1}{2} d_{\mathcal{H}\Delta\mathcal{H}}(\hat{D}_s, \hat{D}_T) + \sigma, \quad (26)$$

where $\epsilon_T(h)$ denotes true risk on the target domain, $\epsilon_s(h)$ denotes true risk on the source domains, $\hat{D}_s = \{\hat{D}_{si} | i = 1, 2, \dots, N\}$, $\mathcal{H}\Delta\mathcal{H}$ is the symmetric difference operator, $d_{\mathcal{H}\Delta\mathcal{H}}(\cdot, \cdot)$ is the $\mathcal{H}\Delta\mathcal{H}$ -divergence, and σ is the shared cross-domain error. It is worth noting that $\sigma = \min_{h \in \mathcal{H}} \zeta_s(h, g_s) + \zeta_T(h, g_T)$, where g_s is a domain-specific real labeling function, and ζ_s is the discrepancy between two labeling functions w.r.t. $\mathcal{X} \in \{\mathcal{T}, s\}$. By minimizing the standard cross-entropy loss (21), the PCMDA model learns the supervised knowledge from source data. Therefore, $\epsilon_s(h)$ is restricted. $d_{\mathcal{H}\Delta\mathcal{H}}(\hat{D}_s, \hat{D}_T)$ is constrained by our contrastive prototype adaptation and prototype combination regularization. Inspired by [9], [38], we have the shared cross-domain error:

$$\begin{aligned} \sigma &= \min_{h \in \mathcal{H}} \zeta_s(h, g_s) + \zeta_T(h, g_T) \\ &\leq \min_{h \in \mathcal{H}} \zeta_s(h, g_s) + \zeta_T(h, g_s) + \zeta_T(g_s, g_T) \\ &\leq \min_{h \in \mathcal{H}} \zeta_s(h, g_s) + \zeta_T(h, g_s) + \zeta_T(g_s, \hat{g}_T) \\ &\quad + \zeta_T(g_T, \hat{g}_T), \end{aligned} \quad (27)$$

where \hat{g}_T is a pseudo-labeling function for D_T , and $\hat{g}_T = C(f(\cdot))$.

Similar to $\epsilon_s(h)$, the approximation of h to g_s is facilitated by minimizing the standard cross-entropy loss. This ensures that $\zeta_s(h, g_s)$ and $\zeta_T(h, g_s)$ (i.e., the discrepancy between h and g_s) are reduced as much as possible. A key insight into the PCMDA model design is enabling g_T to closely resemble \hat{g}_T by effectively learning target-domain knowledge. To achieve this, we integrate the prototype classification loss (as defined in (22)) and a pseudo-labeling strategy [22], which together

Algorithm 1: PCMDA Solving Algorithm.

Input: N different labeled source domains

$D_s = \{(X_{si}^{(j)}, y_{si}^{(j)}) | j = 1, 2, \dots, m_{si}\} | i = 1, 2, \dots, N\}$, the target domain

$D_T = \{X_T^{(j)} | j = 1, 2, \dots, m_T\}$,

mini-batch-size $|B|$, the momentum coefficient β and the number of category K .

Output: well-trained feature extractors f^* ,
well-trained classifiers C^* .

```

1 Randomly sample  $\frac{m_T}{|B|}$  mini-batches.
2 Initialize the weights of  $f$  and  $C$ .
3 while the algorithm is not converged do
4   for each mini-batch do
5     Feed the mini-batch samples to  $f$ . \textcolor{blue}{\texttt{\textbackslash\textbackslash obtain feature embeddings}}
6     Feed the feature embeddings to  $C$ . \textcolor{blue}{\texttt{\textbackslash\textbackslash obtain classification probabilities}}
7     Assign pseudo-labels to the target samples according to classification probabilities.
8     Divide  $\hat{D}_T$  into  $\hat{D}_{T1}$  and  $\hat{D}_{T2}$ .
9     \textcolor{red}{\texttt{\textbackslash\textbackslash Prototype estimation.}}
10    for each  $k \in \{1, 2, \dots, K\}$  do
11      Generate source estimated prototype  $\hat{u}_{si}^{(k)}$  and target estimated prototype  $\hat{u}_T^{(k)}$  according to (1) and (2).
12       $u_{si}^{(k)} \leftarrow \beta u_{si}^{(k)} + (1 - \beta) \hat{u}_{si}^{(k)}$   $i = 1, 2, \dots, N$ 
13       $u_T^{(k)} \leftarrow \beta u_T^{(k)} + (1 - \beta) \hat{u}_T^{(k)}$  \textcolor{blue}{\texttt{\textbackslash\textbackslash update prototypes}}}
14    end
15     $\mathbb{C}_T = \{u_T^{(1)}, u_T^{(2)}, \dots, u_T^{(K)}\}$  \textcolor{blue}{\texttt{\textbackslash\textbackslash target prototypes}}}
16     $p^{(k)} = [u_{s1}^{(k)}, u_{s2}^{(k)}, \dots, u_{sN}^{(k)}]$   $k = 1, 2, \dots, K$  \textcolor{blue}{\texttt{\textbackslash\textbackslash source prototypes}}}
17    \textcolor{red}{\texttt{\textbackslash\textbackslash Prototype combination.}}
18     $w^{(k)} = ((p^{(k)})^T (p^{(k)}))^{-1} (p^{(k)})^T u_T^{(k)}$   $k = 1, 2, \dots, K$  \textcolor{blue}{\texttt{\textbackslash\textbackslash weight calculation}}}
19     $M = \{p^{(1)} \cdot w^{(1)}, p^{(2)} \cdot w^{(2)}, \dots, p^{(K)} \cdot w^{(K)}\}$  \textcolor{blue}{\texttt{\textbackslash\textbackslash calculate prototype combinations}}}
20    \textcolor{red}{\texttt{\textbackslash\textbackslash Construct the learning objectives.}}
21    Calculate the compactness loss  $\mathcal{L}_{com}(f)$  according to (12).
22    Calculate the alignment loss  $\mathcal{L}_{ali}(f)$  according to (18).
23    Calculate the prototype alignment loss  $\mathcal{L}_{pa}(M, \mathbb{C}_T)$  according to (19).
24     $\mathcal{L}_u(f) = \mathcal{L}_{com}(f) + \mathcal{L}_{ali}(f) + \mathcal{L}_{pa}(M, \mathbb{C}_T)$ 
25    Calculate the prototype combination regularization loss  $\mathcal{L}_{reg}(M, \hat{D}_{T2}; f)$  according to (20).
26    Calculate the classification losses  $\mathcal{L}_{cls}(C, f)$  according to (23).
27    Update  $f$  and  $C$  according to (24) and (25).
28 end
29  $f^* \leftarrow f$ 
30  $C^* \leftarrow C$ 

```

enable the PCMDA model to leverage both semi-supervised and high-confidence pseudo-labeled target data. When the prototype classification loss (22) is minimized, the approximation of $C(u_{\mathcal{T}}^{(k)})$ to the real label k is achieved. Additionally, the pseudo-labeling strategy selects high-quality pseudo-labeled target samples, which are then utilized for prototype estimation and contrastive prototype adaptation. Therefore, $\zeta_{\mathcal{T}}(g_{\mathcal{T}}, \hat{g}_{\mathcal{T}})$ can be restrained.

In the PCMDA model, the generalization risk $\zeta_{\mathcal{T}}(g_s, \hat{g}_{\mathcal{T}})$ is bounded by the following weighted risk:

$$\zeta_{\mathcal{T}}(g_s, \hat{g}_{\mathcal{T}}) \leq \sum_{i=1}^N \sum_{k=1}^K e_i^{(k)} \zeta_{\mathcal{T}}(g_{si}^{(k)}, \hat{g}_{\mathcal{T}}^{(k)}), \quad (28)$$

where $g_{si}^{(k)}$ is the labeling function of all samples belonging to the k -th category in \hat{D}_{si} , $g_{\mathcal{T}}^{(k)}$ is the labeling function of all samples belonging to the k -th category in $\hat{D}_{\mathcal{T}}$, and $e_i^{(k)}$ is the weight of $\zeta_{\mathcal{T}}(g_{si}^{(k)}, \hat{g}_{\mathcal{T}}^{(k)})$.

$\zeta_{\mathcal{T}}(g_{si}^{(k)}, \hat{g}_{\mathcal{T}}^{(k)})$ is the discrepancy metric between $D_{si}^{(k)}$ and $D_{\mathcal{T}}^{(k)}$, where $D_{\mathcal{T}}^{(k)}$ is the set of all samples belonging to the k -th category in $D_{\mathcal{T}}$. The key to constraining $\zeta_{\mathcal{T}}(g_{si}^{(k)}, \hat{g}_{\mathcal{T}}^{(k)})$ is to facilitate the alignment between $D_{si}^{(k)}$ and $D_{\mathcal{T}}^{(k)}$. To establish this alignment, we design a dual-alignment strategy that integrates contrastive prototype adaptation and prototype combination regularization (as defined as (20)). The contrastive prototype adaptation framework aligns source feature embeddings with their corresponding target prototypes according to (16), while high-confidence pseudo-labeled target feature embeddings are aligned with their corresponding prototype combinations according to (17). This ensures that category-specific semantic information is effectively transferred across domains. Additionally, the prototype combination regularization aligns the distribution of prototype combinations with the low-confidence pseudo-labeled target samples (i.e., samples in $\hat{D}_{\mathcal{T}2}$). This regularization step enhances the model's ability to address domain-level discrepancies by leveraging all available target samples, regardless of pseudo-label quality. These mechanisms enable the PCMDA model to dynamically balance category-level alignment and domain-wide consistency, reducing discrepancies and facilitating more robust domain adaptation. Therefore, $\zeta_{\mathcal{T}}(g_{si}^{(k)}, \hat{g}_{\mathcal{T}}^{(k)})$ can be constrained.

For each $k \in \{1, 2, \dots, K\}$, the PCMDA model can dynamically assign weights to the source prototypes by our prototype combination mechanism such that $w^{(k)} = \arg \min_{e \in \mathbb{R}^N} \|p^{(k)} - u_{\mathcal{T}}^{(k)}\|^2$, so $w^{(k)} (w^{(k)} \in \mathbb{R}^N)$ can be rewritten as $w^{(k)} =$

$\arg \min_{[e_1, e_2, \dots, e_N] \in \mathbb{R}^N} \sum_{i=1}^N e_i \zeta_{\mathcal{T}}(g_{si}^{(k)}, \hat{g}_{\mathcal{T}}^{(k)})$. Therefore, the PCMDA model can optimize the following objective over the weight array $\mathcal{W} = [e_{ki}]_{K \times N}$ to obtain a tight bound.

$$\zeta_{\mathcal{T}}(g_s, \hat{g}_{\mathcal{T}}) \leq \min_{\mathcal{W} \in \mathbb{R}^{K \times N}} \sum_{i=1}^N \sum_{k=1}^K e_{ki} \zeta_{\mathcal{T}}(g_{si}^{(k)}, \hat{g}_{\mathcal{T}}^{(k)}). \quad (29)$$

The upper bounds will be reduced in their entirety. This theoretical result demonstrates that our method is effective.

IV. EXPERIMENTS

In this section, we evaluated our method and compared it with other state-of-the-art domain adaptation methods on three datasets: Digits-5, PACS [48], and Office_caltech_10 [49]. Besides, we are interested in the effectiveness of the main model components, for which we conducted related experiments for validation. Finally, we construct further visualization experiments to substantiate the theses as we claim.

A. Datasets

1) Digits-5 is a set of handwritten digit images sampled from five domains: MNIST (mt) [50], MNIST-M (mm) [51], USPS (up) [52], SynthDigits (syn) [51], and SVHN (sv) [53]. Following the setting in [41], a random sample of 25,000 images was selected for training, 6,000 images for validation, and 9,000 images for testing on Digits-5. All samples are images of numbers ranging from 0 to 9.

2) PACS [48] is composed of four different datasets, each representing a different visual domain: Photo (P), Art Painting (A), Cartoon (C), and Sketch (S). It contains 9,944 images, including 1,792 real photos, 2,048 art paintings, 2,344 cartoon images, and 2,760 sketches. Each image is labeled with one of seven animal categories.

3) Office_caltech_10 [49] is a commonly used domain adaptation dataset that consists of four domains: Amazon (A), Webcam (W), DSLR (D), and Caltech (C). It contains 9,000 images, with 2,533 images from Amazon, 795 images from Webcam, 498 images from DSLR, and 4,174 images from Caltech. Each image is labeled with one of ten categories.

B. Experimental Settings

All experiments were implemented on the PyTorch platform [54] and deployed on the same device. For a fair comparison, the same data pre-processing routines and the same model architecture were adopted in all experiments conducted on the same dataset. Specifically, the 3 conv-2 fc network [10], [25], the ResNet18 network, and the ResNet101 network are employed as the feature extractors $f(\cdot)$ on Digits-5, PACS [48], and Office_caltech_10 [49], respectively. The weights of the 3 conv-2 fc network are randomly initialized before training. Both the ResNet18 network and the ResNet101 network are pre-trained on ImageNet. For all experiments, the classification models are conducted with two fully connected layers, in which the dimension of feature embedding is $d \rightarrow K$ (d is the feature embedding dimension and K is the number of categories). The number of training epochs Max_epoch is set to 100 to ensure that the algorithm has converged [10], [25]. The temperature parameter τ and the trade-off parameter γ are both set as 1.0. To reduce noisy activations at the early stages of training, α is gradually transitioned from 0 to 1 according to a progressive schedule [19]: $\alpha = \frac{2}{1 + \exp(-10t/Max_epoch)} - 1$, where t indicates the model training up to the t -th epoch. All experiments are repeated five times under the identical configuration. Similar schemes have been broadly used in [25], [55], [56] to avoid accidental errors.

TABLE I
CLASSIFICATION ACCURACY (MEAN \pm STD%) ON DIGITS-5 DATASET

Standards	Models	$\rightarrow mt$	$\rightarrow mm$	$\rightarrow syn$	$\rightarrow sv$	$\rightarrow up$	Avg
Single Best	Source Only	97.2 \pm 0.6	59.1 \pm 0.6	84.6 \pm 0.8	77.7 \pm 0.7	84.7 \pm 1.0	80.8
	DAN [57]	96.3 \pm 0.5	63.8 \pm 0.7	85.4 \pm 0.8	62.5 \pm 0.7	94.2 \pm 0.9	80.4
	CORAL [61]	97.2 \pm 0.8	62.5 \pm 0.7	82.8 \pm 0.7	64.4 \pm 0.7	93.5 \pm 0.8	80.1
	DANN [19]	97.6 \pm 0.8	71.3 \pm 0.6	85.4 \pm 0.8	63.5 \pm 0.8	92.3 \pm 0.9	82.0
	ADDA [59]	97.9 \pm 0.8	71.6 \pm 0.5	86.5 \pm 0.6	75.5 \pm 0.5	92.8 \pm 0.7	84.8
Source Combined	Source Only	90.2 \pm 0.8	63.4 \pm 0.8	82.4 \pm 0.7	62.9 \pm 0.9	88.8 \pm 0.8	77.5
	DAN [57]	97.5 \pm 0.6	67.9 \pm 0.8	86.9 \pm 0.5	67.8 \pm 0.6	93.5 \pm 0.8	82.7
	DANN [19]	97.9 \pm 0.7	70.8 \pm 0.8	87.4 \pm 0.9	68.5 \pm 0.5	93.5 \pm 0.8	83.6
	JAN [58]	97.2 \pm 0.7	65.9 \pm 0.7	86.6 \pm 0.6	75.3 \pm 0.7	95.4 \pm 0.8	84.1
	MCD [60]	96.2 \pm 0.8	72.5 \pm 0.7	87.5 \pm 0.7	78.9 \pm 0.8	95.3 \pm 0.7	86.1
Multi-Source	MDAN [27]	98.0 \pm 0.9	69.5 \pm 0.3	87.4 \pm 0.5	69.2 \pm 0.6	92.4 \pm 0.7	83.3
	DCTN [41]	96.2 \pm 0.8	70.5 \pm 1.2	86.8 \pm 0.8	77.6 \pm 0.4	92.8 \pm 0.3	84.8
	DRT [56]	99.3\pm0.1	81.0 \pm 0.3	93.8 \pm 0.3	77.6 \pm 0.4	98.4 \pm 0.1	91.8
	M^3SDA [25]	98.4 \pm 0.7	72.8 \pm 1.1	89.6 \pm 0.6	81.3 \pm 0.9	96.1 \pm 0.8	87.7
	Ltc-MSDA [10]	99.0 \pm 0.4	85.6 \pm 0.8	93.0 \pm 0.5	83.2 \pm 0.6	98.3 \pm 0.4	91.8
	CMSS [55]	99.0 \pm 0.1	75.3 \pm 0.6	93.7 \pm 0.2	88.4\pm0.5	97.7 \pm 0.1	90.8
	MLAN [29]	98.6 \pm 0.0	86.3 \pm 0.3	93.0 \pm 0.3	82.8 \pm 0.1	97.0 \pm 0.2	91.6
	PCMDA (ours)	99.1 \pm 0.1	91.4\pm0.2	94.2\pm0.2	82.4 \pm 0.2	98.8\pm0.2	93.2

The best results are annotated in bold font.

C. Compared Method

We compared our PCMDA model with the following three comparative standards:

- *Single Best*: The best results of the single-source unsupervised domain adaptation algorithm are reported among all domains.
- *Source Combine*: All the source domain data are integrated into a large-scale source domain. We report the result of single-source unsupervised domain adaptation algorithms on this dataset.
- *Multi-Source*: The multi-source unsupervised domain adaptation algorithms transfer the knowledge of multiple sources to a target domain.

We compare our method with the following competitors, i.e., DAN [57], JAN [58], DANN [19], ADDA [59], MCD [60], CORAL [61], MEDA [62], Meta-MCD [45], DRT [56], Ltc-MSDA [10], M^3SDA [25], DCTN [41], CMSS [55], MDAN [27], MDDA [26], and MLAN [29].

D. Result

1) *Experiments on Digits-five*: As shown in Table I, we compared our proposed method PCMDA with other methods on Digits-five. Source Only was used as the baseline, representing the model trained with only the source data. The results show that our PCMDA method achieves the highest average accuracy of 93.2% and outperforms the state-of-the-art method DRT [56] by 1.4%. Our approach achieves an average accuracy gain of 19.9% over the baseline. Especially on the hard-to-transfer “ $\rightarrow mm$ ” task [56], our approach obtains a remarkable performance improvement of 5.1% compared to MLAN [29]. These results demonstrate the efficacy of the PCMDA model. This success can be attributed to our approach, which adeptly combines category-discriminative semantic information that is beneficial for cross-domain image classification. Meanwhile, it can be observed that our PCMDA is not optimal on the “ $\rightarrow sv$ ” task. This is due to the fact that a significant proportion of the image samples on SVHN comprise multiple digits (e.g., samples in Fig. 3). It is difficult to allocate pseudo-labels to these samples through [22] (i.e., a pseudo-labeling strategy



Fig. 3. Some multiple-digit images on SVHN.

TABLE II
CLASSIFICATION ACCURACY (MEAN \pm STD%) ON PACS DATASET

Models	$\rightarrow A$	$\rightarrow C$	$\rightarrow P$	$\rightarrow S$	Avg
Source Only	75.8 \pm 0.7	73.2 \pm 0.9	92.1 \pm 0.6	64.3 \pm 1.2	76.3
MDAN [27]	79.1 \pm 0.4	76.0 \pm 0.7	91.4 \pm 0.9	72.0 \pm 0.8	79.6
DCTN [41]	84.7 \pm 0.7	86.7 \pm 0.6	95.6 \pm 0.8	71.8 \pm 1.0	84.7
MDDA [26]	86.7 \pm 0.6	86.2 \pm 0.7	93.9 \pm 0.7	77.6 \pm 0.9	86.1
M^3SDA [25]	89.3\pm0.4	89.9\pm1.0	97.3\pm0.3	76.7 \pm 2.9	88.3
Meta-MCD [45]	87.4 \pm 0.7	86.2 \pm 0.9	97.1 \pm 0.5	78.3 \pm 0.8	87.2
PCMDA (ours)	89.3\pm 0.1	88.9 \pm 0.2	97.0 \pm 0.2	82.1\pm0.5	89.3

The best results are annotated in bold font.

based on classification probabilities). This severely limits the capacity of our prototype combination mechanism to combine category-discriminative semantic information.

2) *Experiments on PACS*: The results of various methods on PACS are presented in Table II. From the Table II, it can be observed that PCMDA achieves an 89.3% average accuracy, outperforming the best competitor M^3SDA [25] by 1.0%. In particular, our method achieves a performance gain of 3.8% compared to the Meta-MCD [45] method on the “ $\rightarrow S$ ” classification task. PCMDA achieves the highest average accuracy in three of the four transfer tasks. Our approach does not obtain the highest average accuracy on the “ $\rightarrow P$ ” task, which is mainly ascribed to performance saturation. It is worth noting that the Source Only model achieved a performance exceeding 92.0% on the “ $\rightarrow P$ ” task. The performance gap among PCMDA, M^3SDA and Meta-MCD is below 0.3% when they are evaluated on the “ $\rightarrow P$ ” task. Our PCMDA method remains the best method among the competitors, consistently verifying the efficacy of our PCMDA model.

TABLE III
CLASSIFICATION ACCURACY (MEAN \pm STD%) ON OFFICE_CALTECH_10 DATASET

Models	$\rightarrow W$	$\rightarrow D$	$\rightarrow C$	$\rightarrow A$	Avg
Source Only	99.0	98.3	87.8	86.1	92.8
MEDA [62]	99.3	99.2	91.4	92.9	95.7
DCTN [41]	99.4	99.0	90.2	92.7	95.3
JAN [58]	99.4	99.4	91.2	91.8	95.5
M^3SDA [25]	99.4	99.2	91.5	94.1	96.1
MCD [60]	99.5	99.1	91.5	92.1	95.6
PCMDA (ours)	99.5\pm0.2	100.0\pm0.0	94.6\pm0.0	95.4\pm0.1	97.4

The best results are annotated in bold font.

3) *Experiments on Office_caltech_10*: The results on Office_caltech_10 are reported in Table III. On Office_caltech_10, our method remains the best performer, which obtains a 1.3% performance gain compared to M^3SDA . The result of three datasets demonstrates the stability and generalizability of PCMDA. Remarkably, on the “ $\rightarrow D$ ” task, our approach achieved a 100.0% accuracy, which is the peak performance. However, our PCMDA does not obtain significant superiority on this dataset, which can be attributed to two primary factors. Firstly, all domain adaptation models show saturated performance on the “ $\rightarrow D$ ” and the “ $\rightarrow W$ ” tasks, where the Source-only model’s performance exceeds 98%. Such a phenomenon indicates that there is little scope for improvement through domain adaptation. Secondly, the Webcam domain, the DSLR domain, and the Caltech domain are extremely similar, so the domain shift among these domains is tiny. The high similarity of these source domains restrains the superiority brought by the combination of category features on the “ $\rightarrow A$ ” task.

Additionally, we conducted statistical significance analysis experiments on the three datasets to prove the improvement that we claimed. We compare our PCMDA algorithm with two methods, i.e., DRT [56] (i.e., the best competitor of Digits-5) and M^3SDA [25] (i.e., the best competitor of PACS and Office_caltech_10). To ensure the reliability and statistical significance of the results, we repeat the experiments thirty times [63] under the identical configuration. The paired t-test [64] is adopted to compare the performance differences of different models on the same dataset. We formulate two hypotheses: the null hypothesis H_0 (i.e., there is no significant difference between the models.) and the alternative hypothesis H_1 (i.e., there is a significant difference between the models.). As shown in Table VI, all p-values of average accuracies are less than the significance level (i.e., 0.05), indicating that H_0 should be rejected and that the performance difference is statistically significant. These results substantiate the improvement that we claimed.

E. Ablation Study

To further verify the effectiveness of domain alignment losses (i.e., \mathcal{L}_{reg} and \mathcal{L}_u), the ablation experiments are implemented on Digits-5. Table IV shows the model’s performance under four configurations on Digits-5. Under \mathcal{L}_{cls} constraint alone, the model achieves an accuracy of 83.0%. With \mathcal{L}_{cls} and \mathcal{L}_{reg} , the model only achieves an accuracy of 83.9% because numerous high-quality pseudo-labeled target samples are not utilized for

TABLE IV
ABLATION ANALYSIS OF THE PROTOTYPE COMBINATION REGULARIZATION LOSS AND CONTRASTIVE LOSSES ON DIGITS-5

\mathcal{L}_u	\mathcal{L}_{reg}	$\rightarrow mt$	$\rightarrow mm$	$\rightarrow syn$	$\rightarrow sv$	$\rightarrow up$	Avg
		98.3	65.3	83.5	71.3	96.7	83.0
✓		99.1	89.1	93.0	80.7	98.5	92.0
	✓	98.4	66.2	85.4	74.0	95.3	83.9
✓	✓	99.1	91.4	94.2	82.4	98.8	93.2

TABLE V
ABLATION ANALYSIS OF CLASSIFICATION LOSSES ON DIGITS-5

\mathcal{L}_{sou}	\mathcal{L}_{pro}	$\rightarrow mt$	$\rightarrow mm$	$\rightarrow syn$	$\rightarrow sv$	$\rightarrow up$	Avg
✓		99.1	91.0	93.8	78.9	98.3	92.2
	✓	98.7	84.6	93.5	66.9	98.2	88.4
✓	✓	99.1	91.4	94.2	82.4	98.8	93.2

TABLE VI
STATISTICAL SIGNIFICANCE ANALYSIS OF AVERAGE ACCURACIES FOR PCMDA ON THREE DATASETS

Dataset	Competitor	Test Statistic	P-value
Digits-5	DRT [56]	167	8.63×10^{-45}
PACS	M^3SDA [25]	23.1	9.43×10^{-20}
Office_caltech_10	M^3SDA [25]	52.9	1.30×10^{-29}

TABLE VII
COMPARISON FOR DIFFERENT DISTANCE METRICS ON DIGITS-5

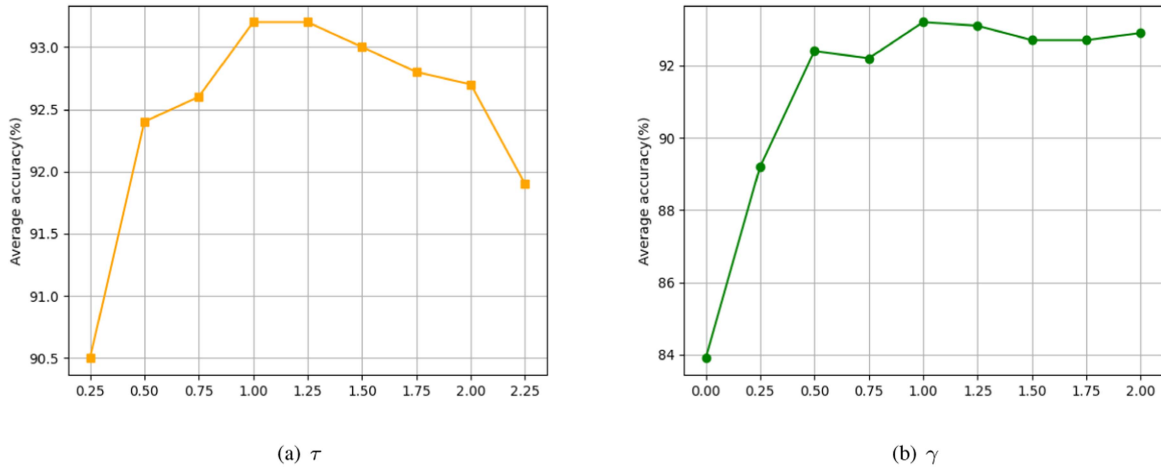
Models	$\rightarrow mt$	$\rightarrow mm$	$\rightarrow syn$	$\rightarrow sv$	$\rightarrow up$	Avg
PCMDA-O	97.3	82.0	87.4	78.2	85.3	86.0
PCMDA-M	98.1	81.7	90.6	83.5	96.7	90.1
PCMDA	99.1	91.4	94.2	82.4	98.8	93.2

modeling training. After considering the \mathcal{L}_u term, an 9.0% performance improvement is achieved. It demonstrates that \mathcal{L}_u is beneficial to the categorical features learning. Furthermore, after simultaneously adding \mathcal{L}_u and \mathcal{L}_{reg} , the complete PCMDA model achieves the highest average accuracy of 93.2%.

Additionally, we conduct component analysis of classification losses. The results are presented in Table V. Without \mathcal{L}_{pro} constraint, the model achieved a 92.2% average accuracy. Without \mathcal{L}_{sou} constraint, the average accuracy of the model has dropped dramatically because the model has yet to learn the supervised knowledge from source samples. After adding the \mathcal{L}_{pro} , the complete PCMDA achieved a 1.0% performance improvement. The results demonstrate that supervised learning of prototypes facilitates the feature discriminability of the model while simultaneously reducing the influence of noisy data. The ablation result shows that each component of PCMDA is positive for performance improvement.

F. Distance Comparison

In this study, we explore the impact of different distance metrics (i.e., \cdot , \cdot in (15)) on constructing category probability distributions. To investigate this, we design two configurations: PCMDA-O and PCMDA-M. PCMDA-O represents the model where category probability distributions are constructed using the Euclidean distance, while PCMDA-M employs the Manhattan distance for the same purpose. From the results presented in Table VII, it is evident that the cosine similarity function consistently outperforms both Euclidean and Manhattan distances.

Fig. 4. Sensitivity of hyperparameters experiments of τ and γ . (All results are evaluated on Digits-5).

G. Influence of Hyperparameters

In this experiment, we evaluate the sensitivity of PCMDA to the temperature parameter τ (i.e., τ in (13), (14), (16), and (17).) and the trade-off parameter γ (i.e., γ in (24)). In the contrasting losses, τ determines the degree of the model attention paid to negative samples. If the temperature parameter τ is set high, the logit distribution becomes smoother, and the contrastive loss treats all samples equally, leading to the model learning without prioritizing. Conversely, when τ is small, the losses become more sensitive to differences between positive samples and negative samples, resulting in poor generalization to unseen data. Therefore, it is necessary to find a suitable parameter τ . As illustrated in Fig. 4(a), different τ values are selected for the PCMDA model on Digits-5. The best average performance is achieved when τ is around 1.0. In addition, our PCMDA achieves competitive results robustly when the parameter τ is varied in the interval $[0.25, 2.25]$. On Digits-5, we evaluate the sensitivity of γ that balances the objectives for \mathcal{L}_u , \mathcal{L}_{reg} , and \mathcal{L}_{cls} . As shown in Fig. 4(b), the best average performance is achieved when γ is around 1.0. The performance of PCMDA is significantly diminished when γ is set to 0, which indicates that there is no category-wise domain alignment (\mathcal{L}_u constraint). Additionally, PCMDA is not sensitive to γ when the parameter γ is varied in the interval $[0, 2.0]$, which further verifies the stability of PCMDA.

H. Quantitative Analysis for Intra-Class Differences

In this part, we conduct a quantitative analysis of the impact of intra-class differences on algorithm performance. We evaluate the PCMDA algorithm under a new configuration PCMDA-equal, which stands for the model optimized according to $w^{(k)} = \frac{1}{N}\mathbf{1}$, where $\mathbf{1} \in \mathbb{R}^N$. This indicates that PCMDA-equal treats multiple source domains as equal. As demonstrated in Tables VIII, IX, and X, the results highlight the efficacy of our prototype combination mechanism. Due to the intra-class differences, the transfer abilities of source domains are varied. Our prototype combination mechanism effectively assigns optimal weights to source prototypes, ensuring that the resulting

TABLE VIII
QUANTITATIVELY ANALYSIS ON DIGITS-5

Models	$\rightarrow mt$	$\rightarrow mm$	$\rightarrow syn$	$\rightarrow sv$	$\rightarrow up$	Avg
PCMDA-equal	98.9	89.0	92.9	77.8	98.2	91.4
PCMDA	99.1	91.4	94.2	82.4	98.8	93.2

TABLE IX
QUANTITATIVELY ANALYSIS ON OFFICE_CALTECH_10

Models	$\rightarrow A$	$\rightarrow C$	$\rightarrow P$	$\rightarrow S$	Avg
PCMDA-equal	99.4	100.0	91.3	93.6	96.1
PCMDA	99.5	100.0	94.6	95.4	97.4

TABLE X
QUANTITATIVELY ANALYSIS ON PACS

Models	$\rightarrow W$	$\rightarrow D$	$\rightarrow C$	$\rightarrow A$	Avg
PCMDA-equal	87.5	88.1	96.4	80.3	88.1
PCMDA	89.3	88.9	97.0	82.1	89.3

prototype combinations closely align with their corresponding target prototypes. By leveraging this alignment, the PCMDA model is able to extract and utilize more target-relevant knowledge compared to the PCMDA-equal model. Consequently, the PCMDA model consistently outperforms PCMDA-equal, demonstrating its superior capacity to adapt to diverse domain shifts.

I. Effect of the Prototype Combination Regularization

We further investigate the effect of the prototype combination regularization in the PCMDA model. To conduct a comprehensive investigation, we design a prototype combination regularization loss $\mathcal{L}_{reg}(M, \hat{D}_{\mathcal{T}}; f)$ (30) to validate whether the prototype combination regularization is applicable to all target samples.

$$\mathcal{L}_{reg}(M, \hat{D}_{\mathcal{T}}; f) = \left\| \mathbb{E}_{b^{(k)} \in M} \phi(b^{(k)}) - \mathbb{E}_{X_{\mathcal{T}} \in \hat{D}_{\mathcal{T}}} \phi(f(X_{\mathcal{T}})) \right\|^2. \quad (30)$$

In this experiment, we evaluate the prototype combination regularization under two configurations on Digits-5. Target-All

TABLE XI
EFFECT OF THE PROTOTYPE COMBINATION REGULARIZATION ON DIGITS-5

Models	$\rightarrow mt$	$\rightarrow mm$	$\rightarrow syn$	$\rightarrow sv$	$\rightarrow up$	Avg
Target-All	98.7	91.0	94.2	78.2	97.0	91.8
Target-Low	98.4	66.2	85.4	74.0	95.3	83.9
PCMDA	99.1	91.4	94.2	82.4	98.8	93.2

stands for the model optimized by (25) and (31), which enhances the distributional alignment between the prototype combinations and all target samples. Target-Low stands for the model optimized by (25) and (32), which enhances the distributional alignment between the prototype combinations and the low-confidence pseudo-labeled target samples.











$$\min_f \mathcal{L}_{cls}(C, f) + \alpha \mathcal{L}_{reg}(M, \hat{D}_T; f). \quad (31)$$

$$\min_f \mathcal{L}_{cls}(C, f) + \alpha \mathcal{L}_{reg}(M, \hat{D}_{T2}; f). \quad (32)$$









Target-Low is trained with low-confidence pseudo-labeled target samples rather than all target samples. Therefore, we design Target-All to evaluate the effect of the prototype combination regularization. Target-All selects the source semantic information that is beneficial for cross-domain image classification through our prototype combination mechanism. It then aligns the distribution between the prototype combinations and all target samples by prototype combination regularization. As shown in Table XI, Target-All and Target-Low achieve 91.8% and 83.9% average accuracies, respectively. Compared to the methods shown in Table I, Target-All obtains an average accuracy that is surpassed only by our PCMDA. This result demonstrates that the prototype combination regularization can effectively reduce the domain shift between source and target domains. However, in Table XI, it can be observed that PCMDA outperforms the Target-All model across all transfer tasks on Digits-5. This is due to the fact that the prototype combination regularization (i.e., domain-wise alignment approach) is unable to alleviate the intra-class differences among all domains, which results in the misclassification of some target samples at the decision boundary. As demonstrated in Tables XI and IV, the prototype combination regularization approach can further reduce the domain shift and enhance classification accuracy across all transfer tasks on Digits-5.

J. Visualization









1) *Visualization of Feature Combination*: In this experiment, we investigate whether PCMDA is capable of selecting and combining category features that are beneficial for cross-domain image classification. We randomly sampled 1000 training samples per domain on Digits-5, 8 training samples (i.e., the number of “mug” samples in DSLR domain) per domain on Office_caltech_10, and 80 training samples (i.e., the number of “house” samples in SKETCH domain) per domain on PACS. Through our prototype combination mechanism, we can obtain the weights of source prototypes. It’s worth noting that 8 and 80 are obtained by $n = \min_{i \in \{1, 2, \dots, N\}, k \in \{1, 2, \dots, K\}} |D_{si}^{(k)}|$, where

class	mt	up	sv	syn	mm
4	 0.90	 -0.20	 0.35	 0.05	
0	 0.61	 0.54	 0.07	 -0.01	

(a) Prototype weight distribution on the “ $\rightarrow mm$ ” task in Digits-5

class	W	D	C	A
mug	 -0.32	 0.41	 0.87	
headphones	 0.21	 -0.13	 1.26	

(b) Prototype weight distribution on the “ $\rightarrow A$ ” task in Office_caltech_10

class	C	P	S	A
elephant	 -0.12	 0.39	 0.73	
house	 0.37	 0.67	 0.31	

(c) Prototype weight distribution on the “ $\rightarrow A$ ” task in PACS

Fig. 5. Visualization of prototype weight distribution. All weights are obtained at inference time using the PCMDA model that has been trained for two epochs.

$|D_{si}^{(k)}|$ is the number of the samples belonging to the k -th category in D_{si} . To facilitate comprehension, we provide illustrative examples, as shown in Fig. 5.

We visualize samples belonging to some fixed categories and the weights of their corresponding prototypes. In Fig. 5, we find that the feature similarity discovered by PCMDA is different for different transfer tasks. As shown in Fig. 5(a), the PCMDA model allocates the high weight to the source prototype whose corresponding samples are of similar handwriting styles to the target samples belonging to the same category. For instance, the samples belonging to the 0-th category in MNIST, USPS, and MNIST-M are analogous to the irregular “0” shape, while the samples belonging to the 0-th category in SynthDigits and SVHN are analogous to the regular “0” shape. One would think that the PCMDA model simply selects similar background features instead of category-discriminative features. In contrast, the

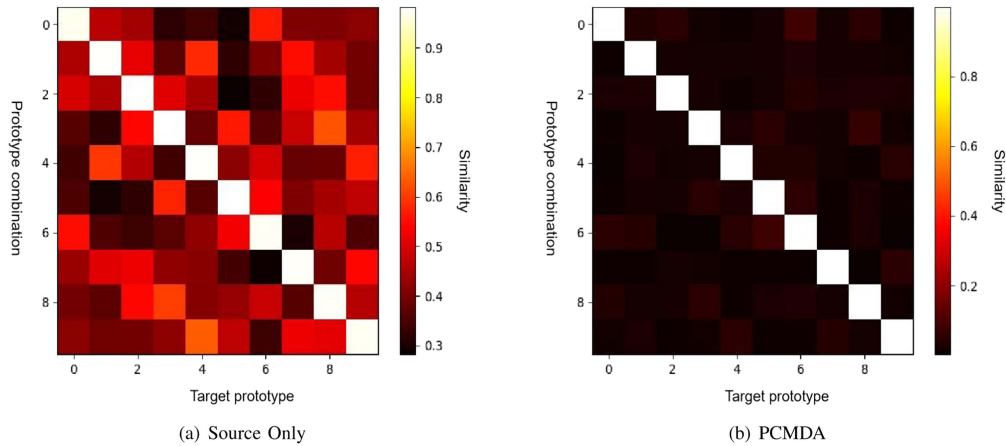


Fig. 6. Visualization of the similarity matrix between the prototype combination and the target prototype. (The results are evaluated on the “ $\rightarrow mm$ ” task.).

PCMDA model prioritizes the selection of the source structural features (i.e., category-discriminative features) that are similar to the target structural features. On the transfer task of 4-th category, the PCMDA model prioritizes the selection of the irregular “4” shape features (e.g., the 4-th category features of MNIST), rather than the similar background features (e.g., the background features of SynthDigits and the background features of SVHN).

Similarly, on Office_caltech_10 and PACS, the PCMDA model prioritizes the selection of source domain structural features that closely resemble those of the target domain. As illustrated in Fig. 5(b) and (c), the PCMDA model emphasizes the “mug” prototype of the Caltech domain on Office_caltech_10 and the “elephant” prototype of the Sketch domain on PACS, demonstrating its ability to dynamically identify and leverage domain-specific semantic features that contribute to effective cross-domain alignment. Compared to [25], [27], [29], PCMDA can automatically select and combine the category-discriminative features rather than relying on simple weight induction.

2) *Visualization of the Similarity Matrix*: Fig. 6 shows the heatmap of the similarity matrix between 10 prototype combinations and 10 target prototypes from on the “ $\rightarrow mm$ ” of Digits-5 under two configurations (Source Only and PCMDA). In Fig. 6, we can observe that each prototype combination is very similar to the corresponding target prototype under both configurations. However, as shown in Fig. 6(a), a significant degree of similarity persists between prototype combinations and target prototypes belonging to disparate categories under the Source Only configuration. This similarity ultimately results in the misclassification of the target samples at the decision boundary. As illustrated in Fig. 6(b), in PCMDA, the similarity between prototype combinations and the target prototypes belonging to different categories is close to zero. This result demonstrates that the decision boundaries of the PCMDA model are more discernible than those of Source Only, thus indicating that the PCMDA model exhibits superior generalization capabilities on the target data.

3) *Visualization of Feature Distributions*: In this part, we visualize the feature distributions on the “ $\rightarrow mm$ ” task of

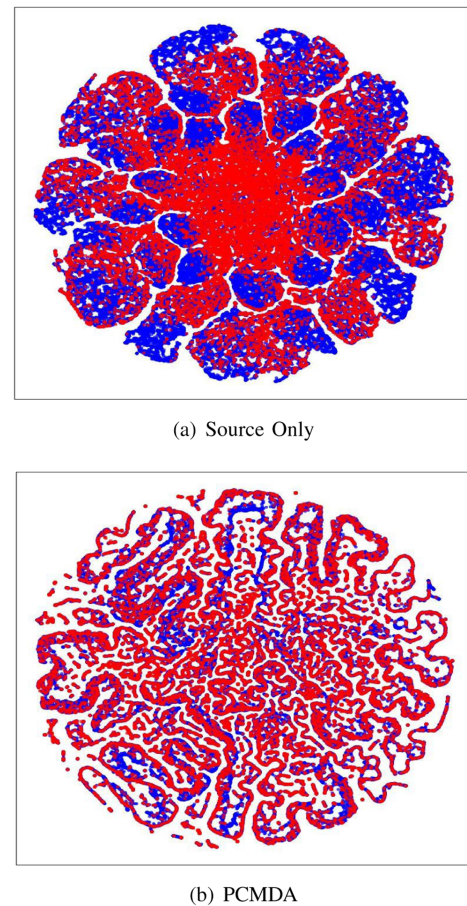


Fig. 7. Feature distributions of source domain (“blue”) and target domain (“red”). (The results are evaluated on the “ $\rightarrow mm$ ” task.).

Digits-5 by t-SNE [65]. As shown in Fig. 7, the scatter plot of PCMDA indicates that the source and target feature embeddings have more overlapping regions compared to the scatter plot of Source Only. This suggests that the source and target feature distributions are better aligned in the PCMDA model than in the Source Only model. This phenomenon is consistent with our experimental results on the “ $\rightarrow mm$ ” task.

V. CONCLUSION

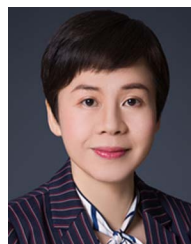
This paper proposes a prototype combination method for the multi-source unsupervised domain adaptation problem (PCMDA). In the proposed method, a prototype combination mechanism is designed to model the target domain as a weighted combination of multiple source domains at the category level. Since the quality of target pseudo-labels is varied, a contrastive prototype adaptation framework and a prototype combination regularization are designed for diverse confidence pseudo-labeled target samples. The synergistic effect between the contrastive prototype adaptation framework and the prototype combination regularization leads to a progressive reduction in the discrepancy between the source and target domains, thus enhancing the capabilities of the PCMDA model in the target domain inference. Additionally, we provide theoretical guarantees for our prototype combination mechanism. We conducted extensive experiments on three benchmarks, i.e., Digits-5, PACS, and Office_caltech_10, on which our method PCMDA achieves 93.2%, 89.3%, and 97.4% average accuracies, respectively. These results demonstrate that PCMDA is capable of selecting and combining the category features that are beneficial for cross-domain image classification. In addition, analytical experiments demonstrate that our prototype combination regularization for low-confidence pseudo-labeled target samples can further reduce the domain shift and enhance classification accuracy.

Though the effectiveness of PCMDA has been demonstrated, there are two notable limitations that cannot be overlooked. First, the PCMDA method relies on the assumption that target category features can be accurately represented through a linear combination of prototypes. This assumption may break down in scenarios where the domain shift exhibits highly non-linear or complex characteristics. Second, using linear weighting constrains the model's representational capacity, as it cannot capture interactions or higher-order relationships between source domains that could be advantageous for adaptation. In future work, we aim to address these limitations by exploring alternative cross-domain combination strategies, including non-linear and pixel-level approaches. For non-linear combination schemes, we plan to design algorithms capable of modeling intricate interactions between source and target domains in a non-linear manner, thereby enhancing the adaptability to complex domain shifts. For pixel-level combination schemes, our goal is to develop methods that align the distributions of source and target domains at a finer granularity, such as the pixel level, to capture structural and contextual details better. These advancements hold the potential to further improve the robustness and flexibility of PCMDA in challenging domain adaptation scenarios.

REFERENCES

- [1] S. Khan, H. Abbas, and W. Iqbal, "Verifiable privacy-preserving image retrieval in multi-owner multi-user settings," *IEEE Trans. Emerg. Topics Comput. Intell.*, vol. 8, no. 2, pp. 1640–1655, Apr. 2024.
- [2] J. Jin et al., "A complex-valued variant-parameter robust zeroing neural network model and its applications," *IEEE Trans. Emerg. Topics Comput. Intell.*, vol. 8, no. 2, pp. 1303–1321, Apr. 2024.
- [3] H. Yu et al., "Generative adversarial registration network for multi-contrast liver MRI registration and added value to hepatocellular carcinoma segmentation: A multicentre study," *IEEE Trans. Emerg. Topics Comput. Intell.*, vol. 8, no. 2, pp. 1714–1727, Apr. 2024.
- [4] Y. Liang, H. Huang, Z. Cai, Z. Hao, and K. C. Tan, "Deep infrared pedestrian classification based on automatic image matting," *Appl. Soft Comput.*, vol. 77, pp. 484–496, 2019.
- [5] P. K. Mishra, A. Mihailidis, and S. S. Khan, "Skeletal video anomaly detection using deep learning: Survey, challenges, and future directions," *IEEE Trans. Emerg. Topics Comput. Intell.*, vol. 8, no. 2, pp. 1073–1085, Apr. 2024.
- [6] Y. Zhang, H. Huang, Z. Lin, Z. Hao, and G. Hu, "Running-time analysis of evolutionary programming based on Lebesgue measure of searching space," *Neural Comput. Appl.*, vol. 30, pp. 617–626, 2018.
- [7] J. Yosinski, J. Clune, Y. Bengio, and H. Lipson, "How transferable are features in deep neural networks?," in *Proc. Adv. Neural Inf. Proces. Syst.*, 2014, vol. 27, pp. 3320–3328.
- [8] J. Quinero-Candela, M. Sugiyama, A. Schwaighofer, and N. D. Lawrence, *Dataset Shift in Machine Learning*. Cambridge, MA, USA: MIT Press, 2008.
- [9] L. Zhou, N. Li, M. Ye, X. Zhu, and S. Tang, "Source-free domain adaptation with class prototype discovery," *Pattern Recognit.*, vol. 145, 2024, Art. no. 109974.
- [10] H. Wang, M. Xu, B. Ni, and W. Zhang, "Learning to combine: Knowledge aggregation for multi-source domain adaptation," in *Proc. Lect. Notes Comput. Sci.*, 2020, pp. 727–744.
- [11] D. Patel and K. Amin, "A cross-domain semantic similarity measure and multi-source domain adaptation in sentiment analysis," in *Proc. Int. Conf. Inf. Commun. Technol. Sustain. Dev.*, 2022, pp. 760–764.
- [12] Y. Li, Y. Wu, J. Li, and S. Liu, "Prompting large language models for zero-shot domain adaptation in speech recognition," in *Proc. IEEE Autom. Speech Recognit. Underst. Workshop*, 2023, pp. 1–8.
- [13] C. Ge et al., "Domain adaptation via prompt learning," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 36, no. 1, pp. 1160–1170, Jan. 2025.
- [14] C. Yang, X. Guo, Z. Chen, and Y. Yuan, "Source free domain adaptation for medical image segmentation with fourier style mining," *Med. Image Anal.*, vol. 79, 2022, Art. no. 102457.
- [15] T. M. Buonocore, C. Crema, A. Redolfi, R. Bellazzi, and E. Parimbelli, "Localizing in-domain adaptation of transformer-based biomedical language models," *J. Biomed. Inform.*, vol. 144, 2023, Art. no. 104431.
- [16] S. Kumari and P. Singh, "Deep learning for unsupervised domain adaptation in medical imaging: Recent advancements and future perspectives," *Comput. Biol. Med.*, vol. 170, 2024, Art. no. 107912.
- [17] M. Long, Z. Cao, J. Wang, and M. Jordan, "Conditional adversarial domain adaptation," in *Proc. Adv. Neural Inf. Process. Syst.*, 2018, pp. 1647–1657.
- [18] M. Saffari, M. Khodayar, and S. M. J. Jalali, "Sparse adversarial unsupervised domain adaptation with deep dictionary learning for traffic scene classification," *IEEE Trans. Emerg. Topics Comput. Intell.*, vol. 7, no. 4, pp. 1139–1150, Aug. 2023.
- [19] Y. Ganin and V. Lempitsky, "Unsupervised domain adaptation by back-propagation," in *Proc. Int. Conf. Machin. Learn.*, 2015, pp. 1180–1189.
- [20] J. Li et al., "Physical fitness assessment for cancer patients using multi-model decision fusion based on multi-source data," *IEEE Trans. Emerg. Topics Comput. Intell.*, vol. 7, no. 4, pp. 1290–1300, Aug. 2023.
- [21] 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.
- [22] W. Zhang, W. Ouyang, W. Li, and D. Xu, "Collaborative and adversarial network for unsupervised domain adaptation," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 3801–3809.
- [23] M. Xu, H. Wang, and B. Ni, "Graphical modeling for multi-source domain adaptation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 46, no. 3, pp. 1727–1741, Mar. 2024.
- [24] Y. Zhu, F. Zhuang, and D. Wang, "Aligning domain-specific distribution and classifier for cross-domain classification from multiple sources," in *Proc. AAAI Conf. Artif. Intell.*, 2019, vol. 33, no. 01, pp. 5989–5996.
- [25] X. Peng, Q. Bai, X. Xia, Z. Huang, K. Saenko, and B. Wang, "Moment matching for multi-source domain adaptation," in *Proc. - IEEE/CVF Int. Conf. Comput. Vis. Workshops*, 2019, pp. 1406–1415.
- [26] S. Zhao et al., "Multi-source distilling domain adaptation," in *Proc. AAAI Conf. Artif. Intell.*, 2020, vol. 34, no. 07, pp. 12975–12983.
- [27] H. Zhao, S. Zhang, G. Wu, J. M. Moura, J. P. Costeira, and G. J. Gordon, "Adversarial multiple source domain adaptation," in *Proc. Adv. Neural Inf. Proces. Syst.*, 2018, vol. 31, pp. 8568–8579.

- [28] S. J. Pan and Q. Yang, "A survey on transfer learning," *IEEE Trans. Knowl. Data Eng.*, vol. 22, no. 10, pp. 1345–1359, Oct. 2010.
- [29] Y. Xu, M. Kan, S. Shan, and X. Chen, "Mutual learning of joint and separate domain alignments for multi-source domain adaptation," in *Proc. IEEE/CVF Winter Conf. Appl. Comput. Vis. Workshops*, 2022, pp. 1890–1899.
- [30] P. Zhang, B. Zhang, T. Zhang, D. Chen, Y. Wang, and F. Wen, "Prototypical pseudo label denoising and target structure learning for domain adaptive semantic segmentation," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. Workshops*, 2021, pp. 12414–12424.
- [31] Y. Pan, T. Yao, Y. Li, Y. Wang, C. -W. Ngo, and T. Mei, "Transferrable prototypical networks for unsupervised domain adaptation," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. Workshops*, 2019, pp. 2239–2247.
- [32] X. Luo, W. Chen, Z. Liang, C. Li, and Y. Tan, "Adversarial style discrepancy minimization for unsupervised domain adaptation," *Neural Netw.*, vol. 157, pp. 216–225, 2023.
- [33] J. Zhu, H. Bai, and L. Wang, "Patch-mix transformer for unsupervised domain adaptation: A game perspective," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. Workshops*, 2023, pp. 3561–3571.
- [34] C. Su, X. Peng, D. Yang, Z. Li, X. Wu, and W. Zhong, "A two-stage multi-target domain adaptation framework for prediction of key performance indicators based on adversarial network," *IEEE Trans. Emerg. Topics Comput. Intell.*, vol. 8, no. 2, pp. 1772–1787, Apr. 2024.
- [35] M. Wang et al., "Informative pairs mining based adaptive metric learning for adversarial domain adaptation," *Neural Netw.*, vol. 151, pp. 238–249, 2022.
- [36] Y. Yuan, Y. Li, Z. Zhu, R. Li, and X. Gu, "Joint domain adaptation based on adversarial dynamic parameter learning," *IEEE Trans. Emerg. Topics Comput. Intell.*, vol. 5, no. 4, pp. 714–723, Aug. 2021.
- [37] H. Tang, K. Chen, and K. Jia, "Unsupervised domain adaptation via structurally regularized deep clustering," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. Workshops*, 2020, pp. 8725–8735.
- [38] S. Ben-David, J. Blitzer, K. Crammer, A. Kulesza, F. Pereira, and J. W. Vaughan, "A theory of learning from different domains," *Mach. Learn.*, vol. 79, no. 1, pp. 151–175, 2010.
- [39] Y. Mansour, M. Mohri, and A. Rostamizadeh, "Domain adaptation with multiple sources," in *Proc. Adv. Neural Inf. Process. Syst.*, 2008, vol. 21, pp. 1041–1048.
- [40] I.-H. Jhuo, D. Liu, D. Lee, and S.-F. Chang, "Robust visual domain adaptation with low-rank reconstruction," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, 2012, pp. 2168–2175.
- [41] R. Xu, Z. Chen, W. Zuo, J. Yan, and L. Lin, "Deep cocktail network: Multi-source unsupervised domain adaptation with category shift," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 3964–3973.
- [42] M. Ye, X. Zhang, P. C. Yuen, and S.-F. Chang, "Unsupervised embedding learning via invariant and spreading instance feature," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. Workshops*, 2019, pp. 6210–6219.
- [43] K. He, H. Fan, Y. Wu, S. Xie, and R. Girshick, "Momentum contrast for unsupervised visual representation learning," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. Workshops*, 2020, pp. 9729–9738.
- [44] F. Schroff, D. Kalenichenko, and J. Philbin, "Facenet: A unified embedding for face recognition and clustering," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, 2015, pp. 815–823.
- [45] D. Li and T. Hospedales, "Online meta-learning for multi-source and semi-supervised domain adaptation," in *Proc. Lect. Notes Comput. Sci.*, 2020, pp. 382–403.
- [46] S. Yang, H. Huang, F. Luo, Y. Xu, and Z. Hao, "Local-diversity evaluation assignment strategy for decomposition-based multiobjective evolutionary algorithm," *IEEE Trans. Syst. Man Cybern. Syst.*, vol. 53, no. 3, pp. 1697–1709, Mar. 2023.
- [47] A. Gretton, K. M. Borgwardt, M. J. Rasch, B. Schölkopf, and A. Smola, "A kernel two-sample test," *J. Mach. Learn. Res.*, vol. 13, no. 1, pp. 723–773, 2012.
- [48] D. Li, Y. Yang, Y. -Z. Song, and T. M. Hospedales, "Deeper, broader and artier domain generalization," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2017, pp. 5542–5550.
- [49] B. Gong, Y. Shi, F. Sha, and K. Grauman, "Geodesic flow Kernel for unsupervised domain adaptation," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, 2012, pp. 2066–2073.
- [50] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proc. IEEE*, vol. 86, no. 11, pp. 2278–2324, Nov. 1998.
- [51] Y. Ganin et al., "Domain-adversarial training of neural networks," *J. Mach. Learn. Res.*, vol. 17, no. 1, pp. 2096–2030, 2016.
- [52] J. J. Hull, "A database for handwritten text recognition research," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 16, no. 5, pp. 550–554, May 1994.
- [53] N. Yuval, "Reading digits in natural images with unsupervised feature learning," in *Proc. Adv. Neural Inf. Process. Syst.*, 2011, vol. 2011, no. 5, Art. no. 7.
- [54] A. Paszke et al., "Automatic differentiation in pytorch," *NIPS Workshop Autodiff*, 2017. [Online]. Available: <https://openreview.net/forum?id=BJJsrmfCZ>
- [55] L. Yang, Y. Balaji, S. -N. Lim, and A. Shrivastava, "Curriculum manager for source selection in multi-source domain adaptation," in *Proc. Lect. Notes Comput. Sci.*, 2020, pp. 608–624.
- [56] Y. Li, L. Yuan, Y. Chen, P. Wang, and N. Vasconcelos, "Dynamic transfer for multi-source domain adaptation," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. Workshops*, 2021, pp. 10998–11007.
- [57] 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.
- [58] 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.
- [59] E. Tzeng, J. Hoffman, K. Saenko, and T. Darrell, "Adversarial discriminative domain adaptation," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 7167–7176.
- [60] K. Saito, K. Watanabe, Y. Ushiku, and T. Harada, "Maximum classifier discrepancy for unsupervised domain adaptation," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 3723–3732.
- [61] B. Sun and K. Saenko, "Deep coral: Correlation alignment for deep domain adaptation," in *Proc. Lect. Notes Comput. Sci.*, 2016, pp. 443–450.
- [62] J. Wang, W. Feng, Y. Chen, H. Yu, M. Huang, and P. S. Yu, "Visual domain adaptation with manifold embedded distribution alignment," in *Proc. 26th ACM Int. Conf. Multimedia*, 2018, pp. 402–410.
- [63] J. Brownlee, "The magic number 30: Why sample size is often considered sufficient," 2024. [Online]. Available: <https://www.linkedin.com/pulse/magic-number-30-why-sample-size-often-considered-sufficient>
- [64] W. S. Gosset, "The probable error of a mean," *Biometrika*, vol. 6, no. 1, pp. 1–25, 1908.
- [65] L. Van der Maaten and G. Hinton, "Visualizing data using t-SNE," *J. Mach. Learn. Res.*, vol. 9, no. 86, pp. 2579–2605, 2008.



Min Huang received the Ph.D. degree in electrical power engineering from North China Electric Power University, Beijing, China, in 2005. She is currently a Full Professor with the School of Software Engineering, South China University of Technology, Guangzhou, China. Her research interests include the theoretical foundation and application of transfer learning, artificial intelligence, and Big Data processing.



Zifeng Xie (Student_Member) received the B.S. degree in bioengineering in 2021 from the South China University of Technology, Guangzhou, China, where he is currently working toward the M.S. degree with the School of Software Engineering. His research interests include the theoretical foundation of domain adaptation, prototype learning, and computer vision.



Han Huang (Senior Member, IEEE) received the B.S. degree in information management and information system and the Ph.D. degree in computer science from the School of Mathematics, South China University of Technology (SCUT), Guangzhou, China, in 2003 and 2008, respectively. He is currently a Full Professor with the School of Software Engineering, SCUT. His research interests include the theoretical foundation and application of microcomputation, intelligent software engineering, data intelligence engineering, and biochemical computer science. Prof.

Huang is a Distinguished Member of CCF.



Liuqi Zhao received the M.S. degree from the School of Computer Science, Jinan University, Guangzhou, China, in 2019. He is currently with China Southern Power Grid EHV Transmission Company. His research focuses on the application of Big Data processing.



Chang Zhang received the M.S. degree from the School of Software Engineering, South China University of Technology, Guangzhou, China, in 2023. He is currently with the Industrial and Commercial Bank of China. His research interests include the theoretical foundation and application of domain adaptation, and computer vision.



Ziyang Feng received the B.S. degree from the School of Electronic Information Engineering, Wuyi University, Jiangmen, China, in 2008. He is currently with China Southern Power Grid EHV Transmission Company. His research focuses on the application of cloud computing and database.