ORIGINAL ARTICLE

Zeroth- and first-order difference discrimination for unsupervised domain adaptation

Jie Wang[1,2] · Xing Chen[1,2] · Xiao-Lei Zhang[1,2]

Abstract

Unsupervised domain adaptation transfers empirical knowledge from a label-rich source domain to a fully unlabeled target domain with a different distribution. A core idea of many existing approaches is to reduce the distribution divergence between domains. However, they focused only on part of the discrimination, which can be categorized into optimizing the following four objectives: reducing the intraclass distance between domains, enlarging the interclass distances between domains, reducing the intraclass distances within domains, and enlarging the interclass distances within domains. Moreover, because few methods consider multiple types of objectives, the consistency of data representations produced by different types of objectives has not yet been studied. In this paper, to address the above issues, we propose a zeroth- and first-order difference discrimination (ZFOD) approach for unsupervised domain adaptation. It first optimizes the above four objectives simultaneously. To improve the discrimination consistency of the data across the two domains, we propose a first-order difference constraint to align the interclass differences across domains. Because the proposed method needs pseudolabels for the target domain, we adopt a recent pseudolabel generation method to alleviate the negative impact of imprecise pseudolabels. We conducted an extensive comparison with nine representative conventional methods and seven remarkable deep learning-based methods on four benchmark datasets. Experimental results demonstrate that the proposed method, as a conventional approach, not only significantly outperforms the nine conventional comparison methods but is also competitive with the seven deep learning-based comparison methods. In particular, our method achieves an accuracy of 93.4% on the Office+Caltech10 dataset, which outperforms the other comparison methods. An ablation study further demonstrates the effectiveness of the proposed constraint in aligning the objectives.

Introduction

It is known that machine learning benefits from large-scale manually labeled data. However, manual labeling is often time-consuming and labor-intensive. Dealing with scenarios where manually labeled data are insufficient is a difficult task. A natural thought for this problem is to transfer the labeled to unlabeled data knowledge. However, the mismatch between the data distributions may lead to catastrophic results. To address mismatched distribution problem, much efforts have been devoted to unsupervised domain adaptation (UDA) [1–4], which transfers empirical knowledge from a label-rich source domain to an unlabeled target domain with a different distribution.

The core UDA issue is to reduce the distribution gap between the source domain and the target domain [5–9]. A common thought for this problem is to find a shared subspace where the source and target data distributions are similar [10–13]. To learn the subspace, a metric is first needed to evaluate the distribution difference. Common metrics include

✉ Xiao-Lei Zhang
xiaolei.zhang@nwpu.edu.cn

Jie Wang
wangjie2017@mail.nwpu.edu.cn

Xing Chen
xing.chen@mail.nwpu.edu.cn

1 School of Marine Science and Technology, Northwestern Polytechnical University, 127 Youyi West Road, Xi'an 710072, Shaanxi, China

2 Research and Development Institute of Northwestern Polytechnical University in Shenzhen, Shenzhen 518063, Guangdong, China

Fig. 1 Principle of the proposed ZFOD. **a** Zeroth-order discrimination of ZFOD. **b** Discrimination shift problem of the zeroth-order discrimination. **c** First-order difference discrimination for remedying the discrimination shift problem, where the length of the dotted lines represents the interclass distance within the domains
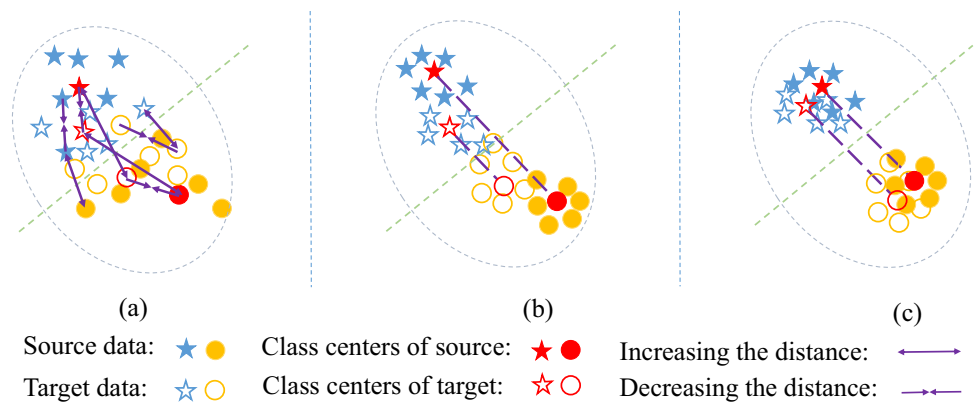
A-distance [14], KL divergence [15], and maximum mean discrepancy (MMD) [16]. Then, an optimization objective defined on the metric with continuous constraints is used to align the source and target domains. For example, [17] reduced the mean deviation of the domains by learning a linear projection matrix with MMD.

Recently, many methods have aimed to further improve the discrimination of the source and target data as an addition to simply aligning the distributions since the latter may weaken data discrimination [12, 18]. The optimization objectives of discriminative learning methods can be roughly divided into four types: reducing the intraclass distances between domains [19, 20], enlarging the interclass distances between domains [21], reducing the intraclass distances within domains [22, 23], and enlarging the interclass distances within domains [24, 25]. For example, [19] aligned the conditional distributions of the source and target domains by reducing the mean deviating between a class in the source domain and the same class in the target domain, which can be regarded as a method of reducing the intraclass distance between domains. [21] proposed a contrastive domain discrepancy to reduce the intraclass distance and enlarge the interclass distance between domains. [25] takes the maximum distance of the data pairs in the same class and the minimum distance of the data pairs in different classes as regularization terms to align the source and target domains, which can be regarded as a mixture of reducing the intraclass distance between and within domains as well as enlarging the interclass distance within domains. Each type of objective shows effectiveness in improving the discrimination of the data representations to some extent; however, to our knowledge, none of the existing methods consider all four objectives together.

Moreover, if multiple types of discriminative optimization objectives are used together, the effects of different objectives on the data representations may be dramatically different, particularly for tasks with large domain shifts. Figures 1a and 1b illustrate this problem, where Fig. 1a demonstrates the principles of multiple types of discriminative optimiza-

tion objectives, and Fig. 1b shows the new data distribution generated by the objectives. From Fig. 1b, we can see that the discrimination of the data distributions is significantly improved compared to that in Fig. 1a; however, the discrimination of the source and target data is different. We call this inconsistent domain alignment the *discrimination shift*, which will finally lead to unsatisfied transfer performance. This problem seems insufficiently explored.

In this paper, we propose a novel UDA algorithm, named *zeroth- and first-order difference discrimination* (ZFOD), to address the above problems. It first optimizes all four discriminative optimization objectives jointly. They are implemented as follows: (i) The minimization of the intraclass distances between domains is implemented by minimizing the distance between the means of the same class across domains. (ii) The maximization of the interclass distance between domains is implemented by maximizing the distance between the mean of a class in the source domain and the mean of another class in the target domain. (iii) The minimization of the intraclass distance within domains is implemented by minimizing the distance of every two samples in the same class for the source and target domains. (iv) The maximization of the interclass distance within domains is implemented by maximizing the distance of every two samples from different classes of either the source domain or the target domain.

To align the components of the ZFOD objective for first-order difference discrimination of data across the source and target domains, motivated by [26], we propose a novel *first-order difference constraint*. As illustrated in Fig. 1c, when ZFOD maximizes the interclass distance within domains, the first-order difference constraint aims to constrain the distances in the source and target domains to be the same.

Notably, optimizing the objective of ZFOD needs to obtain pseudolabels for the target domain. However, the accumulation of pseudolabel errors during the iterative optimization process degrades the performance significantly. To alleviate this difficulty, we use a simple and effective method, named mining target domain intraclass similarity to remedy the pseudolabels (TSRP) [27] and improve accuracy. To

summarize, the novelty and contribution of the paper is summarized as follows:

- We propose jointly optimizing all four discriminative optimization objectives, which is the zeroth-order discrimination of ZFOD.
- We propose first-order difference discrimination to align the objective components. The optimization of the discrimination term in each iteration is formulated as a generalized eigenvalue decomposition problem, which has a simple closed-form solution.
- We conducted an extensive comparison with nine representative conventional methods [10, 19, 20, 24, 25, 28–31] and seven remarkable deep learning-based methods [32–39] on four benchmark datasets, including Office+Caltech10 [29], Office-31 [40], ImageCLEF-DA [41], and Office-Home [42]. Experimental results demonstrate the competitiveness of the proposed method with not only conventional comparison methods but also deep learning-based comparison methods.

The remainder of this paper is organized as follows: In Sect. "Related work", we review some related works. In Sect. "Methods", we present the proposed ZFOD. The experimental results are reported in Sect. "Experiments". Finally, we conclude this paper in Sect. "Conclusion".

Related work

In this section, we first review the methods of learning domain-invariant features and then review the methods that focus on improving data discrimination.

Many domain-invariant feature learning methods have been proposed recently [3, 43], which can be roughly divided into two categories: instance reweighting adaptation methods and feature adaptation methods [3]. Instance reweighting adaptation methods aim to allocate resampling weights directly by feature distribution matching across different domains in a nonparametric manner [22, 44, 45]. For example, [45] proposed an intuitive weighting-based subspace alignment method by reweighting the source samples, which generates a source subspace that is close to a target subspace. [22] reweighted instances by landmark selection so that the pivot samples of the landmarks can be selected as a knowledge transfer bridge, and the outliers can be filtered out. Feature adaptation methods aim to obtain a domain-invariant feature by aligning the domain data distributions [29, 33, 46–50]. For instance, [29] maps both domains into the Grassmann manifold and models the domain shift by constructing geodesic flows. [46] aligned the distributions of source and target domains by aligning their second-order statistics. [47] aligned the distributions indirectly by minimizing the difference between the higher-order central moments of the domains.

In recent years, the idea of generating pseudolabels for the target domain has become popular in UDA [19, 23–25]. However, inaccurate pseudolabels may yield unsatisfactory performance with error accumulation during the optimization process. Therefore, some methods aim to improve pseudolabel accuracy [23, 25, 51]. For example, [51] used three asymmetric classifiers to improve pseudolabel accuracy, where two of the classifiers were used to select confident pseudolabels, and the third learned a discriminative data representation for the target domain. [23] proposed generating accurate pseudolabels by selective pseudolabeling and structured prediction. [52] proposed multistage adaptive label filtering to increase correctly labeled target samples.

With the increasing usage of pseudolabels, an increasing number of methods are considering how to improve the discrimination of a data representation while maintaining its domain-invariant property. Here, we list some representative methods that transform the unsupervised target domain into a supervised target domain with pseudolabels. [24] minimizes the distance of each pair of samples in a class and maximizes the distance of any two samples that belong to different classes for the source and target domains. [5] obtained the pseudolabels of the target domain by clustering and then aligned the class centers of the source and target domains for a domain-invariant subspace. [12] proposed a supervised discriminative MMD with pseudolabels to mitigate the degradation of feature discriminability incurred by MMD. [10] developed an ensemble model by a clustering-promoting technology and obtained the final decision of unlabeled target data via majority voting. [23] utilized supervised locality-preserving projection to reduce the distances between class samples across the source and target domains. [53] proposed cross-domain contrastive learning to make samples within the same category close to each other, while samples from different classes lie far apart, regardless of which domain they come from.

The methods mentioned above only consider part of the discrimination. In this paper, we divide the UDA methods for improving data discrimination into four categories. With this observation, we propose a zeroth-order discrimination objective to further improve the discrimination of source and target data and keep the data distributions of the two domains aligned. Inspired by [26], which constrained the distance of any two samples belonging to different classes to a constant, we propose first-order difference discrimination for UDA. Unlike [26], the proposed first-order difference discrimination constrains the interclass distances of the source domain to be as equal as possible to that of the target domain, thereby mitigating the discriminative differences between the two domains.

Methods

In this section, we first present the proposed ZFOD framework in Sect. "Optimization objective", and then describe its components, which include the interdomain discrimination, intradomain discrimination, first-order difference discrimination, regularization of the projection model, and pseudolabel generation algorithm respectively from Sect. "Interdomain discrimination" to Sect. "Improving the pseudolabels given the solution in (1)", in detail.

Optimization objective

Assume the source domain contains n_s labeled data points $\{(\mathbf{x}_s^i, y_s^i)\}_{i=1}^{n_s}$ and the target domain contains n_t unlabeled data points $\{\mathbf{x}_t^j\}_{j=1}^{n_t}$ where $\mathbf{x}_s^i, \mathbf{x}_t^j \in \mathbb{R}^m$ and $y_s^i \in \{1, 2, \ldots, C\}$ with m denoted as the feature dimension and C denoted as the number of classes. We denote $\mathbf{X}_s = [\mathbf{x}_s^1, \ldots, \mathbf{x}_s^{n_s}]$ and $\mathbf{X}_t = [\mathbf{x}_t^1, \ldots, \mathbf{x}_t^{n_t}]$. The whole data matrix is $\mathbf{X} = [\mathbf{X}_s, \mathbf{X}_t] \in \mathbb{R}^{m \times n}$ with $n = n_s + n_t$. In this paper, we generate and optimize pseudolabels for the target domain. We denote the pseudolabel of the target data $\mathbf{x}_t^j$ as $\hat{y}_t^j$. Our goal is to find a common feature subspace, defined by a projection matrix $\mathbf{P} \in \mathbb{R}^{m \times d}$, such that the new feature representations of the two domains in the new subspace, i.e., $\mathbf{z}_s^i = \mathbf{P}^\top \mathbf{x}_s^i$ and $\mathbf{z}_t^j = \mathbf{P}^\top \mathbf{x}_t^j$, can be effectively aligned, where d is the feature dimension of the subspace.

The core idea of our ZFOD is (i) the zeroth-order discrimination, which consists of four discriminative optimization objectives and can be roughly divided into two categories: the interdomain discrimination $\mathcal{L}_{\text{interD}}$ and intradomain discrimination $\mathcal{L}_{\text{intraD}}$ for improving the data discrimination, and (ii) the first-order difference discrimination $\mathcal{L}_{\text{FD}}$ for aligning the interclass discrimination of the source and target domains. ZFOD is formulated as the following optimization problem:

$$\min_{\mathbf{P}} \mathcal{L}_{\text{interD}} + \alpha \mathcal{L}_{\text{intraD}} + \beta \mathcal{L}_{\text{FD}} + \gamma \|\mathbf{P}\|_F^2 \tag{1}$$

where the operator $\|\cdot\|_F^2$ is the Frobenius norm, and $\|\mathbf{P}\|_F^2$ is a regularization term of $\mathbf{P}$ to avoid overfitting, α, β and γ are three tradeoff parameters, and $\mathcal{L}_{\text{interD}}$, $\mathcal{L}_{\text{intraD}}$, and $\mathcal{L}_{\text{FD}}$ are the functions of the variables $\mathbf{P}$ and $\{\hat{y}_t^j\}_{j=1}^{n_t}$. For simplicity, we fix $\alpha = 1$, which should yield good performance after an empirical investigation on some benchmark datasets.

Interdomain discrimination

The interdomain discrimination objective $\mathcal{L}_{\text{interD}}$ aligns the distributions of the source samples and target samples, so that the discrimination of the source samples can be transferred to the unlabeled target samples. Given the pseudolabels for

the target domain, the interdomain discrimination can be further divided into two parts: interdomain intraclass distance discrimination and interdomain interclass distance discrimination, which are described respectively as follows:

Interdomain intraclass distance discrimination

Interdomain interclass distance discrimination aligns the data distributions of each single class across the two domains as similarly as possible in the subspace, which can be considered a set of C independent UDA problems. We evaluate the interdomain divergence of a class across domains by MMD [16]. The alignment over all classes is formulated as a *minimization* problem of the following objective:

$$
\begin{aligned}
\mathcal{L}_{\text{interD}}^{\text{intraC}} &= \sum_{k=0}^{C} \left\| \frac{1}{n_s^k} \sum_{\mathbf{x}_s^i \in \mathbf{X}_s^k} \mathbf{z}_s^i - \frac{1}{n_t^k} \sum_{\mathbf{x}_t^j \in \mathbf{X}_t^k} \mathbf{z}_t^j \right\|^2 \\
&= \sum_{k=0}^{C} \text{Tr}\left(\mathbf{P}^\top \mathbf{X} \mathbf{M}_k \mathbf{X}^\top \mathbf{P} \right) \\
&= \text{Tr}\left(\mathbf{P}^\top \mathbf{X} \mathbf{M}^{\text{intraC}} \mathbf{X}^\top \mathbf{P} \right)
\end{aligned}
\tag{2}
$$

where $\mathbf{X}_s^k$ denotes the source samples of class k and $\mathbf{X}_t^k$ denotes the target samples with their pseudoclass labels set to k, and n_s^k and n_t^k are the number of the source and target samples of class k, respectively; particularly, the class $k = 0$ denotes the data of the whole source or target domain; $\mathbf{M}_k$ is the intraclass matrix of class k across domains with its element $(\mathbf{M}_k)_{ij}$ defined as:

$$
(\mathbf{M}_k)_{ij} = \begin{cases}
\frac{1}{n_s^k n_s^k}, & \forall \mathbf{x}^i, \mathbf{x}^j \in \mathbf{X}_s^k \\
\frac{1}{n_t^k n_t^k}, & \forall \mathbf{x}^i, \mathbf{x}^j \in \mathbf{X}_t^k \\
-\frac{1}{n_s^k n_t^k}, & \forall \mathbf{x}^i \in \mathbf{X}_s^k, \mathbf{x}^j \in \mathbf{X}_t^k \\
-\frac{1}{n_s^k n_t^k}, & \forall \mathbf{x}^j \in \mathbf{X}_s^k, \mathbf{x}^i \in \mathbf{X}_t^k \\
0, & \text{otherwise}
\end{cases}
\tag{3}
$$

and $\mathbf{M}^{\text{intraC}} = \sum_{k=0}^{C} \mathbf{M}_k$. Note that $\mathbf{M}_k$ is also known as the class-conditional MMD matrix for class k.

Interdomain interclass distance discrimination

Interdomain interclass distance discrimination improves cross-domain data discrimination such that each single class in a domain is far apart from the other classes in the other domain. The interdomain divergence is evaluated by MMD as well. Unlike the problem in Sect. "Interdomain intraclass distance discrimination", the interdomain interclass distance

discrimination is formulated as a *maximization* of the following objective:

$$\mathcal{L}_{\text{interD}}^{\text{interC}} = \sum_{k=1}^{C-1} \sum_{q=k+1}^{C} \left\| \frac{1}{n_s^k} \sum_{\mathbf{x}_s^i \in \mathbf{X}_s^k} \mathbf{z}_s^i - \frac{1}{n_t^q} \sum_{\mathbf{x}_t^j \in \mathbf{X}_t^q} \mathbf{z}_t^j \right\|^2$$

$$= \sum_{k=1}^{C-1} \sum_{q=k+1}^{C} \text{Tr} \left(\mathbf{P}^\top \mathbf{X} \mathbf{M}_{kq} \mathbf{X}^\top \mathbf{P} \right)$$

$$= \text{Tr} \left(\mathbf{P}^\top \mathbf{X} \mathbf{M}^{\text{interC}} \mathbf{X}^\top \mathbf{P} \right) \tag{4}$$

where $\mathbf{M}_{kq}$ is the interclass matrix of class k in the source domain and pseudoclass q in the target domain:

$$\left(\mathbf{M}_{kq} \right)_{ij} = \begin{cases} \frac{1}{n_s^k n_s^k}, & \forall \mathbf{x}^i, \mathbf{x}^j \in \mathbf{X}_s^k \\ \frac{1}{n_t^q n_t^q}, & \forall \mathbf{x}^i, \mathbf{x}^j \in \mathbf{X}_t^q \\ -\frac{1}{n_s^k n_t^q}, & \forall \mathbf{x}^i \in \mathbf{X}_s^k, \mathbf{x}^j \in \mathbf{X}_t^q \\ -\frac{1}{n_s^k n_t^q}, & \forall \mathbf{x}^j \in \mathbf{X}_s^k, \mathbf{x}^i \in \mathbf{X}_t^q \\ 0, & \text{otherwise} \end{cases} \tag{5}$$

and $\mathbf{M}^{\text{interC}} = \sum_{k=1}^{C-1} \sum_{q=k+1}^{C} \mathbf{M}_{kq}$.

We regard that intraclass and interclass discrimination as equivalently important. Eventually, the interdomain discrimination is formulated as:

$$\mathcal{L}_{\text{interD}} = \mathcal{L}_{\text{interD}}^{\text{intraC}} - \mathcal{L}_{\text{interD}}^{\text{interC}}$$

$$= \text{Tr} \left(\mathbf{P}^\top \mathbf{X} \mathbf{M} \mathbf{X}^\top \mathbf{P} \right) \tag{6}$$

where $\mathbf{M}$ represents the interdomain discrimination matrix:

$$\mathbf{M} = \mathbf{M}^{\text{intraC}} - \mathbf{M}^{\text{interC}}. \tag{7}$$

Intradomain discrimination

The intradomain discrimination objective $\mathcal{L}_{\text{intraD}}$ improves the data discrimination at each domain, which can be divided into two parts: intradomain intraclass distance discrimination and intradomain interclass distance discrimination.

Intradomain intraclass discrimination

Intradomain intraclass discrimination reduces the intraclass distance in each domain. Here we define the intraclass distance as the average distance of pairwise samples in a class, which results in a *minimization* problem of the following objective:

$$\mathcal{L}_{\text{intraD}}^{\text{intraC}} = \sum_{k=1}^{C} \frac{n_s}{n_s^k} \sum_{\mathbf{x}_s^i, \mathbf{x}_s^j \in \mathbf{X}_s^k} \left\| \mathbf{z}_s^i - \mathbf{z}_s^j \right\|^2$$

$$+ \sum_{k=1}^{C} \frac{n_t}{n_t^k} \sum_{\mathbf{x}_t^i, \mathbf{x}_t^j \in \mathbf{X}_t^k} \left\| \mathbf{z}_t^i - \mathbf{z}_t^j \right\|^2 \tag{8}$$

$$= \text{Tr} \left(\mathbf{P}^\top \mathbf{X}_s \mathbf{W}_s^{\text{intraC}} \mathbf{X}_s^\top \mathbf{P} \right) + \text{Tr} \left(\mathbf{P}^\top \mathbf{X}_t \mathbf{W}_t^{\text{intraC}} \mathbf{X}_t^\top \mathbf{P} \right)$$

$$= \text{Tr} \left(\mathbf{P}^\top \mathbf{X} \mathbf{W}^{\text{intraC}} \mathbf{X}^\top \mathbf{P} \right)$$

where $\mathbf{W}_s^{\text{intraC}}$ and $\mathbf{W}_t^{\text{intraC}}$ are the intraclass matrices of the source domain and target domain respectively:

$$\left(\mathbf{W}_s^{\text{intraC}} \right)_{ij} = \begin{cases} n_s, & \text{if } i = j \\ -\frac{n_s}{n_s^k}, & \text{if } i \neq j \\ 0, & \text{otherwise} \end{cases}, \quad \forall \mathbf{x}_s^i, \mathbf{x}_s^j \in \mathbf{X}_s^k \tag{9}$$

$$\left(\mathbf{W}_t^{\text{intraC}} \right)_{ij} = \begin{cases} n_t, & \text{if } i = j \\ -\frac{n_t}{n_t^k}, & \text{if } i \neq j \\ 0, & \text{otherwise} \end{cases}, \quad \forall \mathbf{x}_t^i, \mathbf{x}_t^j \in \mathbf{X}_t^k \tag{10}$$

and $\mathbf{W}^{\text{intraC}} = \text{diag} \left(\mathbf{W}_s^{\text{intraC}}, \mathbf{W}_t^{\text{intraC}} \right)$.

Intradomain interclass discrimination

Intradomain interclass discrimination enlarges the interclass distance in each domain. Here, we define the interclass distance as the average of the distances of any pair of samples that belong to different classes in a domain, which results in a *maximization* problem of the following objective:

$$\mathcal{L}_{\text{intraD}}^{\text{interC}} = \sum_{y_s^i \neq y_s^j} \left\| \mathbf{z}_s^i - \mathbf{z}_s^j \right\|^2 + \sum_{\hat{y}_t^i \neq \hat{y}_t^j} \left\| \mathbf{z}_t^i - \mathbf{z}_t^j \right\|^2$$

$$= \text{Tr} \left(\mathbf{P}^\top \mathbf{X}_s \mathbf{W}_s^{\text{interC}} \mathbf{X}_s^\top \mathbf{P} \right) + \text{Tr} \left(\mathbf{P}^\top \mathbf{X}_t \mathbf{W}_t^{\text{interC}} \mathbf{X}_t^\top \mathbf{P} \right)$$

$$= \text{Tr} \left(\mathbf{P}^\top \mathbf{X} \mathbf{W}^{\text{interC}} \mathbf{X}^\top \mathbf{P} \right) \tag{11}$$

where $\mathbf{W}_s^{\text{interC}}$ and $\mathbf{W}_t^{\text{interC}}$ are the interclass matrices of the source domain and target domain, respectively:

$$\left(\mathbf{W}_s^{\text{interC}} \right)_{ij} = \begin{cases} n_s - n_s^k, & \text{if } i = j \text{ and } y_s^i = k \\ -1, & \text{if } i \neq j \text{ and } y_s^i \neq y_s^j \\ 0, & \text{otherwise} \end{cases} \tag{12}$$

$$\left(\mathbf{W}_t^{\text{interC}} \right)_{ij} = \begin{cases} n_t - n_t^k, & \text{if } i = j \text{ and } \hat{y}_t^i = k \\ -1, & \text{if } i \neq j \text{ and } \hat{y}_t^i \neq \hat{y}_t^j \\ 0, & \text{otherwise} \end{cases} \tag{13}$$

and $\mathbf{W}^{\text{interC}} = \text{diag} \left(\mathbf{W}_s^{\text{interC}}, \mathbf{W}_t^{\text{interC}} \right)$.

Finally, we formulate the intradomain discrimination from (8) and (11) as follows:

$$
\begin{aligned}
\mathcal{L}_{\text{intraD}} &= \mathcal{L}_{\text{intraD}}^{\text{intraC}} - \rho \mathcal{L}_{\text{intraD}}^{\text{interC}} \\
&= \text{Tr}\left(\mathbf{P}^\top \mathbf{X} \mathbf{W}^{\text{intraC}} \mathbf{X}^\top \mathbf{P}\right) - \rho \, \text{Tr}\left(\mathbf{P}^\top \mathbf{X} \mathbf{W}^{\text{interC}} \mathbf{X}^\top \mathbf{P}\right) \\
&= \text{Tr}\left(\mathbf{P}^\top \mathbf{X} \mathbf{W} \mathbf{X}^\top \mathbf{P}\right)
\end{aligned}
\tag{14}
$$

where ρ is a hyperparameter for balancing the two terms, and $\mathbf{W}$ is the intradomain discrimination matrix:

$$
\mathbf{W} = \mathbf{W}^{\text{intraC}} - \rho \mathbf{W}^{\text{interC}}
\tag{15}
$$

First-order difference discrimination

Definition

The first-order difference discrimination objective $\mathcal{L}_{FD}$ constrains the distance between any two classes of a domain to be similar to the distance of the same pair of classes of another domain, which is formulated as a *minimization* problem of the following objective:

$$
\begin{aligned}
\mathcal{L}_{\text{FD}} &= \sum_{k=1}^{C-1} \sum_{q=k+1}^{C} \left\| \frac{1}{n_s^k} \sum_{\mathbf{x}_s^i \in \mathbf{X}_s^k} \mathbf{z}_s^i - \frac{1}{n_s^q} \sum_{\mathbf{x}_s^j \in \mathbf{X}_s^q} \mathbf{z}_s^j \right. \\
&\quad \left. - \frac{1}{n_t^k} \sum_{\mathbf{x}_t^i \in \mathbf{X}_t^k} \mathbf{z}_t^i + \frac{1}{n_t^q} \sum_{\mathbf{x}_t^j \in \mathbf{X}_t^q} \mathbf{z}_t^j \right\|^2 \\
&= \sum_{k=1}^{C-1} \sum_{q=k+1}^{C} \text{Tr}\left(\mathbf{P}^\top \mathbf{X} \mathbf{S}_{kq} \mathbf{X}^\top \mathbf{P}\right) \\
&= \text{Tr}\left(\mathbf{P}^\top \mathbf{X} \mathbf{S} \mathbf{X}^\top \mathbf{P}\right)
\end{aligned}
\tag{16}
$$

where

$$
(\mathbf{S}_{kq})_{ij} =
\begin{cases}
\frac{1}{n_s^k n_s^k}, & \forall \mathbf{x}^i, \mathbf{x}^j \in \mathbf{X}_s^k \\
\frac{1}{n_s^q n_s^q}, & \forall \mathbf{x}^i, \mathbf{x}^j \in \mathbf{X}_s^q \\
\frac{1}{n_t^k n_t^k}, & \forall \mathbf{x}^i, \mathbf{x}^j \in \mathbf{X}_t^k \\
\frac{1}{n_t^q n_t^q}, & \forall \mathbf{x}^i, \mathbf{x}^j \in \hat{\mathbf{X}}_t^q \\
\frac{1}{n_s^k n_s^q}, & \forall \mathbf{x}^i \in \mathbf{X}_s^k, \mathbf{x}^j \in \mathbf{X}_t^q \text{ or } \forall \mathbf{x}^j \in \mathbf{X}_s^k, \mathbf{x}^i \in \mathbf{X}_t^q \\
\frac{1}{n_s^q n_s^k}, & \forall \mathbf{x}^i \in \mathbf{X}_s^q, \mathbf{x}^j \in \mathbf{X}_t^k \text{ or } \forall \mathbf{x}^j \in \mathbf{X}_s^q, \mathbf{x}^i \in \mathbf{X}_t^k \\
-\frac{1}{n_s^k n_s^q}, & \forall \mathbf{x}^i \in \mathbf{X}_s^k, \mathbf{x}^j \in \mathbf{X}_s^q \text{ or } \forall \mathbf{x}^j \in \mathbf{X}_s^k, \mathbf{x}^i \in \mathbf{X}_s^q \\
-\frac{1}{n_s^k n_t^k}, & \forall \mathbf{x}^i \in \mathbf{X}_s^k, \mathbf{x}^j \in \mathbf{X}_t^k \text{ or } \forall \mathbf{x}^j \in \mathbf{X}_s^k, \mathbf{x}^i \in \mathbf{X}_t^k \\
-\frac{1}{n_s^q n_t^q}, & \forall \mathbf{x}^i \in \mathbf{X}_s^q, \mathbf{x}^j \in \mathbf{X}_t^q \text{ or } \forall \mathbf{x}^j \in \mathbf{X}_s^q, \mathbf{x}^i \in \mathbf{X}_t^q \\
-\frac{1}{n_t^k n_t^q}, & \forall \mathbf{x}^i \in \mathbf{X}_t^k, \mathbf{x}^j \in \mathbf{X}_t^q \text{ or } \forall \mathbf{x}^j \in \mathbf{X}_t^k, \mathbf{x}^i \in \mathbf{X}_t^q \\
0, & \text{otherwise}
\end{cases}
\tag{17}
$$

Algorithm 1: ZFOD.

Input: Labeled source data $\{(\mathbf{x}_s^i, y_s^i)\}_{i=1}^{n_s}$; unlabeled target data $\{(\mathbf{x}_t^j)\}_{j=1}^{n_t}$; hyperparameters β, ρ and γ; dimension of the subspace d.

Output: Projection matrix $\mathbf{P}$, and the final classifier $f_{\text{final}}(\cdot)$.

1 **while** *not converge* **do**

2   Compute the interdomain discrimination matrix $\mathbf{M}$ in (6);

3   Compute the intradomain discrimination matrix $\mathbf{W}$ in (14);

4   Compute the first-order difference discrimination matrix $\mathbf{S}$ by (16);

5   Update $\mathbf{P}$ by solving the generalized eigenvalue problem in (21);

6   Use TSRP to refine pseudolabels and obtain the final classifier $f_{\text{final}}(\cdot)$;

---

and $\mathbf{S}$ represents the first-order difference discrimination matrix:

$$
\mathbf{S} = \sum_{k=1}^{C-1} \sum_{q=k+1}^{C} \mathbf{S}_{kq}
\tag{18}
$$

Discussion

The first-order difference discrimination $\mathcal{L}_{\text{FD}}$ is designed to remedy the weakness of the interdomain discrimination $\mathcal{L}_{\text{interD}}$.

Specifically, when the data distributions are nonuniform, the large intraclass distance components in $\mathcal{L}_{\text{interC}}^{\text{intraC}}$ and small interclass distance components of $\mathcal{L}_{\text{interC}}^{\text{interC}}$ contribute to the main reduction of the objective value $\min_{\mathbf{P}} \mathcal{L}_{\text{interD}}$ during the optimization process, which makes the solution $\mathbf{P}$ biased toward these components, and against the other components. As a result, the classes that do not benefit much from $\min_{\mathbf{P}} \mathcal{L}_{\text{interD}}$ tend to be not aligned well across domains. In other words, the knowledge of those classes in the source domain may not transfer well to the target domain.

If we calculate the components of $\mathcal{L}_{\text{FD}}$, we find that the classes that are not well aligned tend to make the values of the corresponding components large, while the classes that are well aligned yield small values. Therefore, we propose $\mathcal{L}_{\text{FD}}$ as a supplement to $\mathcal{L}_{\text{interD}}$.

Optimization algorithm

The optimization of ZFOD alternately conducts the following steps: (i) solving problem (1) assigning the pseudolabels and (ii) improving the pseudolabels with the solution of (1). See the following subsections for the two steps. See Algorithm 1 for a summarization of ZFOD.

Solving problem (1) given the pseudolabels

Substituting (6), (14), and (16) into (1) results in the following optimization problem:

$$\min_{\mathbf{P}} \operatorname{Tr} \left(\mathbf{P}^\top \mathbf{X} \left(\mathbf{M} + \mathbf{W} + \beta \mathbf{S} \right) \mathbf{X}^\top \mathbf{P} \right) + \gamma \left\| \mathbf{P} \right\|_F^2$$
$$\text{s.t. } \mathbf{P}^\top \mathbf{X} \mathbf{H} \mathbf{X}^\top \mathbf{P} = \mathbf{I}_d \tag{19}$$

where the constraint maximizes the embedded data variance as [24] and [19] did, $\mathbf{I}_d$ is an identify matrix of dimension d and $\mathbf{H} = \mathbf{I}_{(n_s+n_t)} - (1/(n_s + n_t))\mathbf{1}_{(n_s+n_t)\times(n_s+n_t)}$.

The Lagrangian function of problem (19) is:

$$\mathcal{L}(\mathbf{P}, \Theta) = \operatorname{Tr} \left(\mathbf{P}^\top \left(\mathbf{X} \left(\mathbf{M} + \mathbf{W} + \beta \mathbf{S} \right) \mathbf{X}^\top + \gamma \mathbf{I}_m \right) \mathbf{P} \right)$$
$$+ \operatorname{Tr} \left(\left(\mathbf{I}_d - \mathbf{P}^\top \mathbf{X} \mathbf{H} \mathbf{X}^\top \mathbf{P} \right) \Theta \right) \tag{20}$$

where $\Theta = \operatorname{diag}(\theta_1, \theta_2, ..., \theta_d) \in \mathbb{R}^{d\times d}$ is a diagonal matrix with Lagrange multipliers. Solving $\partial \mathcal{L}(\mathbf{P}, \Theta)/\partial \mathbf{P} = 0$ derives the following optimal solution of (19):

$$\mathbf{P}^\star = \left(\mathbf{X} \left(\mathbf{M} + \mathbf{W} + \beta \mathbf{S} \right) \mathbf{X}^\top + \gamma \mathbf{I}_m \right)^{-1} \mathbf{X} \mathbf{H} \mathbf{X}^\top \mathbf{P} \Theta \tag{21}$$

which is a generalized eigenvalue decomposition problem. In practice, we select the generalized eigenvectors of the right side of (21) corresponding to the d-smallest eigenvalues as the final $\mathbf{P}^\star$.

Note that problem (19) can be easily generalized into the following kernel form through a kernel mapping $\phi : \mathbf{X} \to \phi(\mathbf{X})$:

$$\min_{\mathbf{P}} \operatorname{Tr} \left(\mathbf{P}^\top \mathbf{K} \left(\mathbf{M} + \mathbf{W} + \beta \mathbf{S} \right) \mathbf{K}^\top \mathbf{P} \right) + \gamma \left\| \mathbf{P} \right\|_F^2$$
$$\text{s.t. } \mathbf{P}^\top \mathbf{K} \mathbf{H} \mathbf{K}^\top \mathbf{P} = \mathbf{I}_d \tag{22}$$

where $\mathbf{K} = \phi(\mathbf{X})^\top \phi(\mathbf{X})$. It has a similar solution to (21).

Improving the pseudolabels given the solution in (1)

Inaccurate pseudolabels of the target domain result in suboptimal performance. To alleviate this problem, we use our previous work, named TSRP [27], to correct the errors of the pseudolabels. For the integrity of the paper, we present the TSRP algorithm briefly as follows:

As shown in Fig. 2, the TSRP first generates coarse pseudolabels for the target domain by a classifier $f_{\text{init}}(\cdot)$ trained on the source samples $\{(\mathbf{z}_s^i, y_s^i)\}_{i=1}^{n_s}$, and calculates the similarity of the samples in each pseudoclass as in Fig. 2a. Then, for each pseudoclass, it removes the samples with low con-
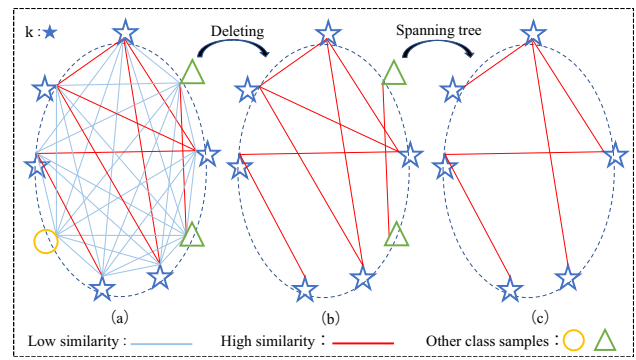
Fig. 2 TSRP principle [27]. TSRP consists of two steps: (i) Deleting, which deletes the samples with low pairwise similarity scores, as shown in Fig. (a) to Fig. (b), and (ii) spanning tree, which selects samples with highly confident pseudolabels by spanning trees, as shown in Fig. (b) to Fig. (c)

fidence, as in Fig. 2a, b, where the confidence is defined as

$$S_{i,j}^k = \begin{cases} \frac{\langle \mathbf{z}_t^i, \mathbf{z}_t^j \rangle}{\|\mathbf{z}_t^i\| \|\mathbf{z}_t^j\|}, & i \neq j, \hat{y}_t^i = \hat{y}_t^j = k \\ 0, & \text{otherwise} \end{cases}, \forall k = 1, 2, \ldots, C \tag{23}$$

Next, as shown as in the operation step from Fig. 2b–c, for each pseudoclass, TSRP connects the samples by spanning trees and then selects the samples of a spanning tree whose root sample has the maximum degree. In this way, the TSRP can further eliminate the negative effect of highly confident misclassified samples. Finally, it uses the selected target-domain samples with high confidence and the source-domain samples together to train a final classifier $f_{\text{final}}(\cdot)$, which is used to generate the refined pseudolabels of the remaining samples with low confidence.

Computational complexity

Assume ZFOD needs T iterations to converge; then the overall computational complexity of ZFOD is $\mathcal{O}(T((C^2+4)(n_s + n_t)^2 + dm^2))$, which is proved as follows:

As shown in Algorithm 1, for each iteration, the computational complexity of ZFOD mainly consists of the following five parts:

- Calculating the interdomain discrimination matrix $\mathbf{M}$ spends $\mathcal{O}(0.5(C^2 + C + 2)(n_s + n_t)^2)$ time.
- Calculating the intradomain discrimination matrix $\mathbf{W}$ spends $\mathcal{O}(2(n_s + n_t)^2)$ time.
- Calculating the first-order difference discrimination matrix $\mathbf{S}$ spends $\mathcal{O}(0.5(C^2 - C)(n_s + n_t)^2)$ time.
- Solving the generalized eigen-decomposition problem takes $\mathcal{O}(dm^2)$ time.

- TSRP costs $\mathcal{O}((n_s + n_t)^2)$.

As the experiments show, because T and the optimal d are usually small numbers, ZFOD can be solved in a polynomial time with respect to the number of data samples.

Storage complexity

The overall storage complexity of ZFOD is $\mathcal{O}((n_s + n_t)^2)$, which is proved as follows:

- Storing the source and target data, which require an $\mathcal{O}((n_s + n_t))$ space.
- Calculating the interdomain discrimination matrix **M**, which require an $\mathcal{O}((n_s + n_t)^2)$ space.
- Calculating the intradomain discrimination matrix **W**, which require an $\mathcal{O}(n_s^2 + n_t^2)$ space.
- Calculating the first-order difference discrimination matrix **S**, which require an $\mathcal{O}((n_s + n_t)^2)$ space.

Experiments

In this section, we evaluate the performance of the proposed ZFOD on several popular visual cross-domain benchmarks. The source code of ZFOD is available at https://github.com/02Bigboy/ZFOD.

Datasets and cross-domain tasks

The experiments were performed on 4 datasets. The detailed information of the datasets are introduced as follows:

Office+Caltech10 [29] contains four domains: Amazon, Webcam, DSLR and Caltech-256, which share the same 10 classes. The dataset has 2533 images in total.

Office-31 [40] consists of three domains: Amazon (A), Webcam (W) and DSLR (D). It contains 4110 images with 31 categories in total.

ImageCLEF-DA [41] has three domains: Caltech-256 (C), ImageNet ILSVRC 2012 (I), and Pascal VOC 2012 (P). It contains 12 classes, each of which has 50 images from three domains.

Office-Home [42] includes 65 object classes from four domains, i.e., Artistic images (A), Clipart (C), Product images (P) and Real-World images (R). There are a total of 15,588 images.

For the Office+Caltech10 dataset, we used the 4096-dimensional DECAF-6 feature [54]. For the other datasets, we used the 2048-dimensional ResNet-50 feature [32].

Experimental settings

For the proposed ZFOD, we applied the following parameter setting to all comparison experiments. Specifically, we set the hyperparameters $\beta = 1.5$, $\gamma = 1$, and $\rho = 0.1$. We set the dimension of the subspace d to 100, and limited the number of the optimization iterations to be no larger than 10, i.e., $N = 10$. As the ablation study shows, ZFOD is insensitive to the hyperparameter setting.

To evaluate the effectiveness of the proposed algorithm, nine representative conventional methods and seven remarkable deep learning-based methods were compared.

Conventional UDA methods

we compare nine conventional UDA methods with ZFOD:

- Classic UDA methods:
  1. 1-nearest neighbor classifier (1NN) [28].
  2. Geodesic flow kernel for domain adaptation (GFK) [29].
  3. Joint geometrical and statistical alignment (JGSA) [30].
  4. Manifold embedded distribution alignment (MEDA) [20].
- Method of reducing the intraclass distance between domains:
  1. Joint distribution adaptation (JDA) [19].
- Method of reducing the intraclass distance between domains and reducing the intraclass distance within domains.
  1. Domain-irrelevant class clustering (DICE) [10].
- Method of reducing the intraclass distance and enlarging the interclass distances within domains.
  1. Minimum centroid shift (MCS) [31].
- Methods of reducing the intraclass distance between domains, enlarging the interclass distances within domains and reducing the intraclass distances within domains.
  1. Discriminatve transfer feature and label consistency (DTLC) [25].
  2. Domain invariant and class discriminative (DICD) [24].
  3. Target similarity for pseudolabels (TSRP) [27].

Note that the proposed ZFOD is closely related to DICD. When ZFOD removes the interclass distance across domains and first-order difference discrimination, ZFOD degrades into DICD.

Table 1 Computational and storage complexities of the comparison methods

Methods	Computational complexity	Storage complexity
JDA	$\mathcal{O}(Tdm^2 + TC(n_s + n_t)^2 + Tm(n_s + n_t))$	$\mathcal{O}((n_s + n_t)^2)$
DICD	$\mathcal{O}(Tdm^2 + (TC + 3T)(n_s + n_t)^2 + Tdm(n_s + n_t))$	$\mathcal{O}((n_s + n_t)^2)$
DTLC	$\mathcal{O}(Tdm^2 + TC(ns + nt)^2 + Tdm(ns + nt) + Tn_t^3)$	$\mathcal{O}((n_s + n_t)^2)$
DICE	$\mathcal{O}(Tdm^2 + Tmn_s n_t + T(n_s + n_t)^2 + Tmd(n_s + n_t))$	$\mathcal{O}((n_s + n_t)^2)$
ZFOD	$\mathcal{O}(Tdm^2 + T(C^2 + 4)(n_s + n_t)^2)$	$\mathcal{O}((n_s + n_t)^2)$

Table 2 Average classification accuracy (%) of the comparison methods on the target domains of the Office+Caltech10 dataset, where A = Amazon, C = Caltech, D = DSLR And W = Webcam. The highest accuracy on a cross-domain task is marked in bold

Tasks \ Methods	NN	GFK	JDA	JGSA	MEDA	DICE	MCS	TSRP	DTLC	DICD	ZFOD
C → A (DeCAF6)	85.7	87.8	89.8	91.4	93.4	92.3	**93.5**	92.4	92.8	91.0	92.1
C → W (DeCAF6)	66.1	70.2	83.7	86.8	95.6	93.6	85.1	94.2	**98.0**	92.2	93.2
C → D (DeCAF6)	74.5	88.5	86.6	93.6	91.1	93.6	90.4	94.9	93.0	93.6	**98.7**
A → C (DeCAF6)	70.3	75.8	82.3	84.9	87.4	85.9	88.3	87.9	88.2	86.0	**88.4**
A → W (DeCAF6)	57.2	77.0	78.6	81.0	88.1	86.4	91.5	89.5	**93.6**	81.4	90.8
A → D (DeCAF6)	64.9	84.1	80.3	88.5	88.1	89.8	91.7	92.4	87.3	83.4	**95.5**
W → C (DeCAF6)	60.3	75.1	83.5	85.0	93.2	85.3	88.3	87.1	88.1	84.0	**89.3**
W → A (DeCAF6)	62.5	82.9	90.2	90.7	**99.4**	90.7	93.4	90.4	92.0	89.7	91.6
W → D (DeCAF6)	98.7	**100**	**100**	**100**	99.4	**100**	**100**	**100**	**100**	**100**	**100**
D → C (DeCAF6)	52.0	73.1	85.1	86.2	87.5	87.4	89.0	88.3	89.3	86.1	**89.5**
D → A (DeCAF6)	62.7	85.2	91.4	92.0	93.2	92.5	93.5	**93.6**	92.9	92.2	93.4
D → W (DeCAF6)	89.2	90.9	99.0	99.7	97.6	90.0	99.3	99.0	**100**	99.0	98.6
Average	70.3	82.0	87.5	90.0	92.8	91.4	92.0	92.5	93.0	89.9	**93.4**

We list the computational and storage complexities of the proposed methods and some comparison methods in Table 1. The table shows that when the number of classes (C) is small, the complexity of the proposed method is similar to those of the comparison methods.

Deep learning-based UDA methods

They are the deep adaptation networks (DAN) [33], residual transfer networks (RTN) [34], multiadversarial DA (MADA) [35], conditional domain adversarial network(CDAN) [36], joint adaptation networks (JAN) [37], Collaborative and adversarial network (iCAN) [38], and maximum classifier discrepancy (MCD) [39], respectively. Note that we also added Resnet-50 [32], which use the classifier trained on the source domain to the test domain directly without applying a specially designed UDA algorithm as a baseline.

The classification accuracy on the target domain is used as the evaluation metric. For a fair comparison, the reported results of the comparison methods were either from their original papers or produced by the publicly available codes of the methods.

Main results

Table 2 lists the classification accuracy of the comparison methods on the domain adaptation tasks of the Office+ Caltech10 dataset. The table shows that ZFOD yields the highest average accuracy, which achieves an absolute improvement of 0.4% over the best competitor DTLC. For the results on each domain adaptation task, ZFOD achieves the highest accuracy on 6 out of 12 tasks.

In particular, the average accuracy of ZFOD is 3.5% higher than that of the closely related comparison method DICD. If we look at the tasks in detail, we find that ZFOD is superior to DICD for all tasks except for "D→W". For example, the accuracy of ZFOD over DICD is improved from 83.4% to 95.5% on the "A→D" task, and from 93.6% to 98.7% on the "C→D" task.

Table 3 shows the classification accuracy of the comparison methods on the Office-31 and ImageCLEF-DA datasets. From the table, we see that ZFOD reaches an average accuracy of 88.1%, which is 0.2% higher than the runner-up comparison method DTLC. It achieves the best results on 3 out of 12 tasks. It is worthy mentioning that ZFOD is superior

Table 3 Average classification accuracy (%) of the comparison methods on the target domains of the Office-31 and ImageCLEF-DA datasets

Tasks \ Methods	Resnet	DAN	RTN	JAN	MADA	CDAN	iCAN	MEDA	DTLC	DICD	ZFOD
A → D	68.9	78.6	77.5	84.7	87.8	89.8	90.1	85.3	**91.0**	81.7	**91.0**
A → W	68.4	80.5	84.5	85.4	90.1	**93.1**	92.5	86.2	85.3	83.3	90.4
D → A	62.5	63.6	66.2	68.6	70.3	70.1	72.0	72.4	73.5	69.4	**74.8**
D → W	96.7	97.1	96.8	97.4	97.4	98.2	**98.8**	97.2	98.0	97.6	98.1
W → A	60.7	62.8	64.8	70.0	66.4	68.0	69.9	**74.0**	73.5	69.4	72.3
W → D	99.3	99.6	99.4	99.8	99.6	**100**	**100**	99.4	99.8	99.4	99.6
I → P	74.8	74.5	75.6	76.8	75.0	76.7	79.5	80.2	**80.5**	77.5	79.5
P → I	83.9	82.2	86.8	88.0	87.9	90.6	89.7	**91.5**	90.8	81.2	**91.5**
I → C	91.5	92.8	95.3	94.7	96.0	97.0	94.7	96.2	**96.7**	94.5	95.8
C → I	78.0	86.3	86.9	89.5	88.8	90.5	89.9	92.7	**95.0**	91.2	93.5
C → P	65.5	69.2	72.7	74.2	75.2	74.5	78.5	79.1	**80.9**	77.0	77.8
P → C	91.2	89.8	92.2	91.7	92.2	93.5	92.0	**95.8**	90.0	87.2	92.5
Average	78.5	81.4	83.2	85.1	85.6	86.8	87.3	87.5	87.9	84.1	**88.1**

Table 4 Average classification accuracy (%) of the comparison methods on the target domains of the Office-Home dataset

Tasks \ Methods	ResNet	DAN	JAN	MCD	CDAN	EasyTL	DICE	JGSA	DICD	ZFOD
Ar → Cl	34.9	43.6	45.9	46.9	49.0	49.8	49.1	51.3	47.4	**52.2**
Ar → Pr	50.0	57.0	61.2	64.1	69.3	72.5	70.7	72.9	67.2	**74.2**
Ar → Rw	58.0	67.9	68.9	77.6	74.5	75.8	73.9	**78.5**	69.5	73.4
Cl → Ar	37.4	45.8	50.4	56.1	54.4	60.7	51.4	**58.1**	55.4	57.1
Cl → Pr	41.9	56.5	59.7	62.4	66.0	69.5	65.9	**72.4**	64.5	70.8
Cl → Rw	46.2	60.4	61.0	65.5	68.4	71.2	65.9	**73.4**	63.8	70.9
Pr → Ar	38.5	44.0	45.8	58.9	55.6	59.0	60.0	62.3	59.9	**62.5**
Pr → Cl	31.2	43.6	43.4	45.8	48.3	47.1	48.6	**50.3**	48.2	49.9
Pr → Rw	60.4	67.7	70.3	**80.0**	75.9	76.4	76.2	79.4	73.7	75.8
Rw → Ar	53.9	63.1	63.9	**73.3**	68.4	64.8	65.4	67.9	64.8	65.8
Rw → Cl	41.2	51.5	52.4	49.8	**55.4**	51.1	53.5	53.4	52.6	52.6
Rw → Pr	59.9	74.3	76.8	**83.1**	80.5	77.3	78.8	80.4	76.4	79.4
Average	46.1	56.3	58.3	63.6	63.8	64.6	63.3	63.3	62.0	**65.4**

to DICD in all tasks, and reaches an average accuracy of 3.5% higher than that of DICD. Especially for the "A→D" task of the Office-31 dataset and the "P→I" task of the ImageCLEF-DA dataset, the accuracy has been increased from 81.7% and 81.2% to 91.0% and 91.5%, respectively. Besides, ZFOD outperforms the deep learning-based methods. Its average accuracy is 0.8% higher than the best deep model iCAN.

Table 4 lists the classification accuracy of the comparison methods on the Office-home dataset. From the table, we observe similar phenomena as those on the other datasets. Specifically, ZFOD reaches the highest average accuracy, which is 0.8% higher than the best referenced method EeasyTL. It achieves the highest accuracy on 3 out of 12 tasks, while EeasyTL wins none of the tasks. Compared to the deep learning-based methods, ZFOD outperforms all deep methods with an absolute accuracy improvement of at least

1.6%. For example, ZFOD outperforms CDAN in 7 out of 12 tasks. The accuracy is improved from 66.0% to 70.8% on the "Cl→Pr" task, and from 55.6% to 62.5% on the "Pr→Ar" task.

The above phenomena, which demonstrate the advantage of ZFOD, can be explained as follows: ZFOD not only maximizes the data discrimination in four aspects but also aligns the data distributions of the two domains by the so-called first-order difference discrimination, which may result in better domain-invariant and discriminative representations than the comparison methods. In contrast, some excellent comparison methods, such as MCS, DTLC, and DICD, only maximize part of the data discrimination. Their interdomain alignment is also limited to the data and do not refer to the levels of high-order data discrimination. See Table 5 for a summary of the differences between ZFOD and some rep-

Table 5 Properties of the comparison methods

Method	Interdomain		Intradomain		First-order difference discrimination
	Intraclass	Interclass	Intraclass	Interclass	
DICE [10]	✓	✗	✓	✗	✗
JDA [19]	✓	✗	✗	✗	✗
DTLC [25]	✓	✗	✓	✓	✗
DICD [24]	✓	✗	✓	✓	✗
MCS [31]	✗	✗	✓	✓	✗
ZFOD (proposed)	✓	✓	✓	✓	✓

resentative conventional methods that also optimize the data discrimination to some extent.

Ablation study

ZFOD contains three novel points: TSRP-based pseudolabel generation, first-order difference discrimination, and zeroth-order discrimination. In this subsection, we conducted an ablation study by removing the novel points from ZFOD one by one. The ZFOD without the TSRP-based pseudolabel generation method is denoted as "*ZFOD without TSRP*". The ZFOD without the first two novel points is denoted as "*ZFOD without TSRP and FOD*". If we remove all three novel points, the most closely related method is DICD [24]. When the zeroth-order discrimination removes the interclass distance across domains, then the remaining three discrimination subitems are the same type of subitems as those in DICD. Therefore, we compared the above ZFOD variants with DICD.

The experiments were conducted on the four UDA datasets. For each dataset, we randomly selected two tasks as representatives. The comparison results are listed in Table 6 and analyzed as follows:

Effect of the TSRP-based pseudolabel generation method on performance

From the comparison results between "ZFOD without TSRP" and ZFOD, we see that the performance of "ZFOD without TSRP" is apparently worse than ZFOD, which indicates that the TSRP-based pseudolabel generation method can improve the accuracy of the pseudolabels, and the correctness of the pseudolabels is very important for ZFOD to learn a domain-invariant and discriminative feature.

Effect of first-order discrimination on performance

From the comparison results between "ZFOD without TSRP" and "ZFOD without TSRP and FOD", we see that the former shows slightly better or at least similar performance with the latter on the tasks of Office+Caltech10, Office-31 and ImageCLEF-DA and significantly outperforms the latter on the tasks of Office-home, which shows the effectiveness of the first-order discrimination and that the first-order discrimination can further help align the distribution of the source domain and target domain.

Effect of zeroth-order discrimination on performance

From the comparison results between "ZFOD without TSRP and FOD" and DICD, we find that the zeroth-order discrimination, which has an extra interdomain interclass distance discrimination subitem beyond DICD, outperforms DICD on Office+Caltech10, Office-31 and ImageCLEF-DA. However, it is significantly worse than DICD on Office-Home, which indicates that the advantage of combining all four subitems of the zeroth-order discrimination over the subset of the subitems is not guaranteed due to the discrimination inconsistency problem analyzed in Sect. "Discussion".

After comparing the results between "ZFOD without TSRP and FOD" and DICD, we find that there is no guarantee that "ZFOD without TSRP and FOD" is always better than DICD due to the discrimination inconsistency problem. However, we further find that "ZFOD without TSRP" outperforms DICD apparently in all tasks, which indicates that the first-order discrimination overcomes the discrimination inconsistency problem of the zeroth-order discrimination.

Effects of hyperparameters on performance

ZFOD has four hyperparameters β, γ, ρ, and d. In the previous sections, we used the same hyperparameter setting in all experiments. To study whether the performance of ZFOD is sensitive to different hyperparameter settings, we conducted a grid search for each parameter. During the grid search of one parameter, we fixed the other parameters to their default values. The experiments were conducted on randomly selected tasks of the four UDA datasets: "C → W (Office-Caltech10)", "P → I (ImageCLEF-DA)", "A → W

Table 6 Ablation study for the components of ZFOD

Dataset	Office-Caltech10		Office-31		ImageCLEF-DA		Office-Home	
Methods\Tasks	C → D	A → D	A → D	D → W	P → I	C → P	Ar → Cl	Cl → Rw
DICD	93.6	83.4	81.7	97.6	81.2	70.0	47.4	63.8
ZFOD without TSRP and FOD	96.2	89.2	84.1	**98.2**	81.2	77.0	41.0	56.6
ZFOD without TSRP	95.5	89.2	84.7	98.1	82.7	**78.2**	49.5	65.4
ZFOD	**98.7**	**95.5**	**91.0**	98.1	**91.5**	77.8	**52.2**	**70.9**
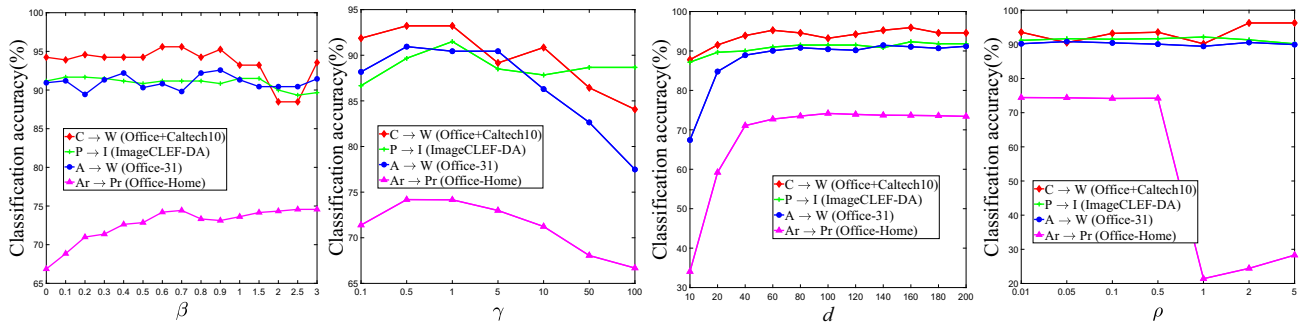
Fig. 3 Effect of the hyperparameters on four domain adaptation tasks. Different colors represent different domain adaptation tasks

(a) DICD   (b) ZFOD
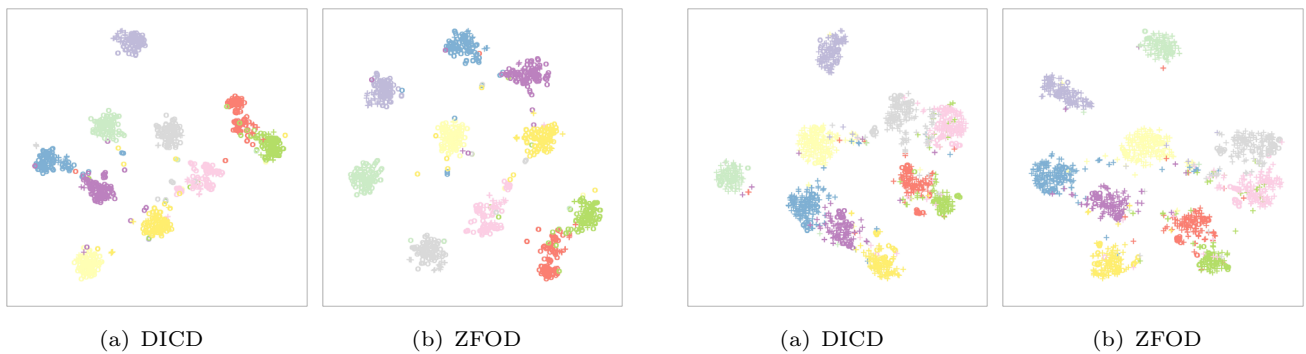
Fig. 4 Visualization of the data distributions produced by DICD and ZFOD on the "A → D (Office+Caltech10)" task. Different colors represent different categories. The source samples are marked by the symbol "o". The target samples are marked by "+"

(a) DICD   (b) ZFOD

Fig. 5 Visualization of the data distributions produced by DICD and ZFOD on the "W → C (Office+Caltech10)" task

(Office-31)" and "Ar → Pr (Office-Home)". The results are shown in Fig. 3 and analyzed as follows:

The hyperparameter β is a hyperparameter for the first-order discrimination. The larger β is, the more important the first-order discrimination behaves in ZFOD. We investigated β in a wide range of [0, 3]. From the result in Fig. 3, we can see that, when we increase β gradually from 0 to 3, the performance of ZFOD improves steadily, especially on the large dataset Office-home. This phenomenon proves that the first-order difference discrimination facilitates the alignment of the source and target domains. Because ZFOD achieves relatively stable results when $\beta \in [0.5, 1.5]$, we chose $\beta = 1.5$ as the default value.

The hyperparameter γ balances the discrimination loss functions and the complexity of the projection matrix **P**. We studied β by a grid search of {0.1, 0.5, 1, 5, 10, 50, 100}. The figure shows that all accuracy curves tend to first rise up and then move down when γ is gradually increased. Eventually, we pick $\gamma = 1$, which tends to yield the best performance for all tasks.

The hyperparameter d defines the dimension of the subspace. We searched d in a wide range of [10, 200]. The figure shows that with the increasing d, the accuracy curves first gradually rise and then tend to be stable. Finally, we chose $d = 100$, which balances the classification performance and computational complexity.

The hyperparameter ρ in (15) balances the intradomain intraclass discrimination and intradomain interclass discrimination. We studied ρ by a grid search of {0.01, 0.05, 0.1, 0.5,

Fig. 6 Cross-domain similarity matrices produced by DICD and ZFOD on the "A → D (Office+Caltech10)" task

(a) DICD

(b) ZFOD
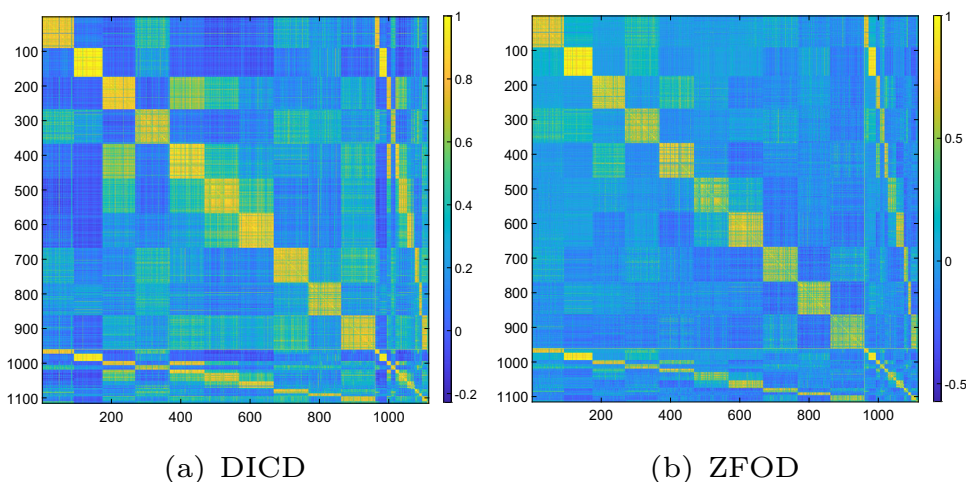
Fig. 7 Cross-domain similarity matrices produced by DICD and ZFOD on the "W → C (Office+Caltech10)" task

(a) DICD

(b) ZFOD
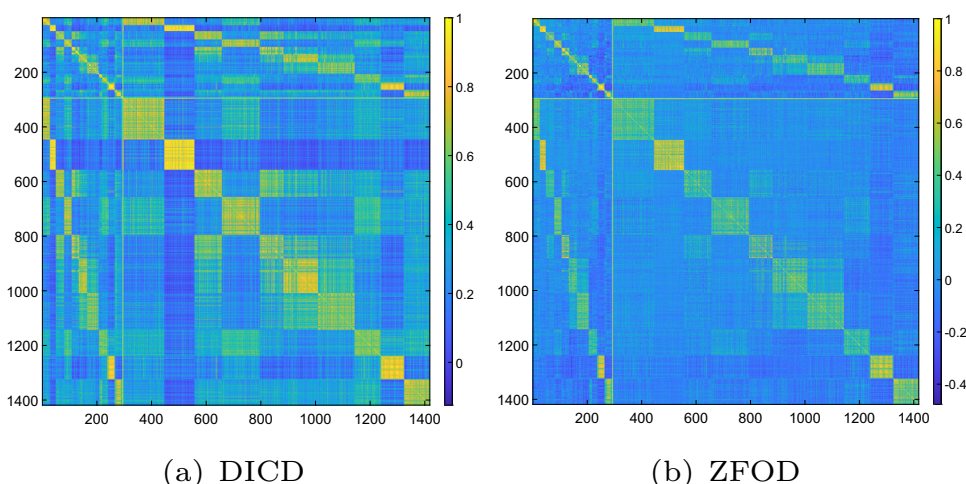
1, 2, 5}. The figure shows that ZFOD performs steadily when $\rho \in [0.01, 0.5]$, and drops sharply on the task of "Ar → Pr (Office-Home)" when $\rho > 0.5$. Therefore, we set $\rho = 0.1$ as a safe default value.

From the above analysis, we conclude that ZFOD is insensitive to the hyperparameter selection. Although the performance of ZFOD can be further improved beyond the results of the previous experimental subsections by carefully selecting a hyperparameter setting per task, we did not do so for the real-world applications of ZFOD.

Data visualization

In this subsection, we visualize the distributions and pairwise similarity matrices of data produced by DICD and ZFOD on two randomly selected tasks, which are the "A → D (Office-Caltech10)" and "W → C (Office-Caltech10)" tasks.

Figures 4 and 5 visualize the data distributions. From the two figures, we see that the data distributions produced by ZFOD have larger interclass distances than those produced by DICD.

Figures 6 and 7 show the similarity matrices of all data across domains, where the similarity between samples is measured by the cosine similarity. A similarity matrix across domains contains three parts: a *source-domain similarity matrix* which is at the upper-left corner of the full similarity matrix, a *target-domain similarity matrix* which is at the lower-right corner, and two *cross-domain similarity matrices* which are at the upper right and lower left corners.

Figures 6a and 7a show that the sample similarity of the cross-domain similarity matrices produced by DICD is quite high for some different classes, which indicates that DICD does not focus enough on domain alignment. In contrast, Figs. 6b and 7b show that ZFOD alleviates this problem, which indicates the effectiveness of the first-order discrimination.

Moreover, Figs. 6a and 7a show that the sample similarity produced by DICD in the source- and target-domain similarity matrices is high for the samples in the same class and not discriminative enough for the samples in different classes. In contrast, Figs. 6b and 7b show that ZFOD not only yields high intraclass similarity as DICD does, but also produces

lower interclass similarity than DICD, which indicates the advantage of the zeroth-order discrimination.

Conclusion

In this paper, we proposed the zeroth- and first-order difference discrimination algorithm for unsupervised domain adaptation. It contains three novel components: zeroth-order discrimination, first-order difference discrimination, and TSRP-based pseudolabel generation. The zeroth-order discrimination consists of interdomain discrimination and intradomain discrimination, each of which is further divided into interclass discrimination and intraclass discrimination. The novelty of zeroth-order discrimination is that it covers four important aspects of data discrimination that have not been considered in the literature to our knowledge. Because the interdomain discrimination only maximizes the cross-domain discrimination at the data level without aligning the interclass distances of the source and target domains, the first-order difference discrimination was proposed to overcome the weakness. Because all of the discrimination terms use pseudolabels to define the pseudoclasses at the target domain, it is important to generate highly accurate pseudolabels. Eventually, the TSRP-based pseudolabel generation method is applied. Its core idea is to iteratively pick the pseudolabels with high confidence to train a strong classifier, which is then used to correct the remaining pseudolabels or improve their confidence. We conducted an extensive comparison with nine state-of-the-art conventional UDA methods and seven representative deep learning-based UDA methods. The comparison results demonstrate the effectiveness of the proposed method. The ablation studies further confirm the effectiveness of each novel component of ZFOD, e.g., in improving the discrimination of the domain-invariant feature and aligning the source and domains well.

Declarations

Conflict of interest The authors declare that they have no conflicts of interests.

References

1. Pan SJ, Yang Q (2010) A survey on transfer learning. IEEE Trans Knowl Data Eng 22(10):1345–1359. https://doi.org/10.1109/TKDE.2009.191
2. Zhuang F, Qi Z, Duan K, Xi D, Zhu Y, Zhu H, Xiong H, He Q (2020) A comprehensive survey on transfer learning. Proc. IEEE 109(1):43–76. https://doi.org/10.1109/JPROC.2020.3004555
3. Zhang L, Gao X (2022) Transfer adaptation learning: a decade survey. IEEE Trans Neural Netw Learn Syst. https://doi.org/10.1109/TNNLS.2022.3183326
4. Tao H, Qiu J, Chen Y, Stojanovic V, Cheng L (2023) Unsupervised cross-domain rolling bearing fault diagnosis based on time-frequency information fusion. J Franklin Inst 360(2):1454–1477. https://doi.org/10.1016/j.jfranklin.2022.11.004
5. Tian L, Tang Y, Hu L, Ren Z, Zhang W (2020) Domain adaptation by class centroid matching and local manifold self-learning. IEEE Trans Image Process 29:9703–9718. https://doi.org/10.1109/TIP.2020.3031220
6. Long M, Wang J, Ding G, Pan SJ, Philip SY (2013) Adaptation regularization: A general framework for transfer learning. IEEE Trans Knowl Data Eng 26(5):1076–1089. https://doi.org/10.1109/TKDE.2013.111
7. Tang H, Wang Y, Jia K (2022) Unsupervised domain adaptation via distilled discriminative clustering. Pattern Recognition 127:108638. https://doi.org/10.1016/j.patcog.2022.108638
8. Yang H, He H, Zhang W, Bai Y, Li T (2022) Lie group manifold analysis: an unsupervised domain adaptation approach for image classification. Appl Intell 52(4):4074–4088. https://doi.org/10.1007/s10489-021-02564-3
9. Zhou C, Tao H, Chen Y, Stojanovic V, Paszke W (2022) Robust point-to-point iterative learning control for constrained systems: A minimum energy approach. Int J Robust Nonlinear Control 32(18):10139–10161. https://doi.org/10.1002/rnc.6354
10. Liang J, He R, Sun Z, Tan T (2018) Aggregating randomized clustering-promoting invariant projections for domain adaptation. IEEE Trans Pattern Anal Mach Intell 41(5):1027–1042. https://doi.org/10.1109/TPAMI.2018.2832198
11. Pilanci M, Vural E (2020) Domain adaptation on graphs by learning aligned graph bases. IEEE Transactions on Knowledge and Data Engineering. https://doi.org/10.1109/TKDE.2020.2984212
12. Wang W, Li H, Ding Z, Nie F, Chen J, Dong X, Wang Z (2021) Rethinking maximum mean discrepancy for visual domain adaptation. IEEE Trans Neural Netw Learn Syst. https://doi.org/10.1109/TNNLS.2021.3093468
13. Zhuang Z, Tao H, Chen Y, Stojanovic V, Paszke W (2022) An optimal iterative learning control approach for linear systems with nonuniform trial lengths under input constraints. IEEE Transactions on Systems, Man, and Cybernetics: Systems. https://doi.org/10.1109/TSMC.2022.3225381
14. Ben-David S, Blitzer J, Crammer K, Kulesza A, Pereira F, Vaughan JW (2010) A theory of learning from different domains. Mach Learn 79(1):151–175. https://doi.org/10.1007/s10994-009-5152-4
15. Kullback S, Leibler RA (1951) On information and sufficiency. Ann Math Stat 22(1):79–86
16. Gretton A, Borgwardt K, Rasch M, Schölkopf B, Smola A (2006) A kernel method for the two-sample-problem. Advances in neural information processing systems **19**

17. Pan SJ, Tsang IW, Kwok JT, Yang Q (2011) Domain adaptation via transfer component analysis. IEEE Trans Neural Netw 22(2):199–210. https://doi.org/10.1109/TNN.2010.2091281

18. Chen Y, Song S, Li S, Wu C (2019) A graph embedding framework for maximum mean discrepancy-based domain adaptation algorithms. IEEE Trans Image Process 29:199–213. https://doi.org/10.1109/TIP.2019.2928630

19. Long M, Wang J, Ding G, Sun J, Yu PS (2013) Transfer feature learning with joint distribution adaptation. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 2200–2207

20. Wang J, Feng W, Chen Y, Yu H, Huang M, Yu PS (2018) Visual domain adaptation with manifold embedded distribution alignment. In: Proceedings of the 26th ACM International Conference on Multimedia, pp. 402–410

21. Kang G, Jiang L, Yang Y, Hauptmann AG (2019) Contrastive adaptation network for unsupervised domain adaptation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 4893–4902

22. Li J, Jing M, Lu K, Zhu L, Shen HT (2019) Locality preserving joint transfer for domain adaptation. IEEE Trans Image Process 28(12):6103–6115. https://doi.org/10.1109/TIP.2019.2924174

23. Wang Q, Breckon T (2020) Unsupervised domain adaptation via structured prediction based selective pseudo-labeling. Proc AAAI Conf Artificial Intell 34:6243–6250. https://doi.org/10.1609/aaai.v34i04.6091

24. Li S, Song S, Huang G, Ding Z, Wu C (2018) Domain invariant and class discriminative feature learning for visual domain adaptation. IEEE Trans Image Process 27(9):4260–4273. https://doi.org/10.1109/TIP.2018.2839528

25. Li S, Liu CH, Su L, Xie B, Ding Z, Chen CP, Wu D (2020) Discriminative transfer feature and label consistency for cross-domain image classification. IEEE Trans Neural Netw Learn Syst 31(11):4842–4856. https://doi.org/10.1109/TNNLS.2019.2958152

26. Yang B, Yuen PC (2019) Cross-domain visual representations via unsupervised graph alignment. Proc AAAI Conf Artificial Intell 33:5613–5620

27. Wang J, Zhang X-L (2023) Improving pseudo labels with intra-class similarity for unsupervised domain adaptation. Pattern Recognition, 109379 . https://doi.org/10.1016/j.patcog.2023.109379

28. Fukunaga K, Narendra PM (1975) A branch and bound algorithm for computing k-nearest neighbors. IEEE Trans Comput 100(7):750–753

29. Gong B, Shi Y, Sha F, Grauman K (2012) Geodesic flow kernel for unsupervised domain adaptation. In: 2012 IEEE Conference on Computer Vision and Pattern Recognition, pp. 2066–2073. IEEE

30. Zhang J, Li W, Ogunbona P (2017) Joint geometrical and statistical alignment for visual domain adaptation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1859–1867

31. Liang J, He R, Sun Z, Tan T (2019) Distant supervised centroid shift: A simple and efficient approach to visual domain adaptation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 2975–2984

32. He K, Zhang X, Ren S, Sun J (2016) Deep residual learning for image recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 770–778

33. Long M, Cao Y, Wang J, Jordan M (2015) Learning transferable features with deep adaptation networks. In: International Conference on Machine Learning, pp. 97–105. PMLR

34. Long M, Zhu H, Wang J, Jordan MI (2016) Unsupervised domain adaptation with residual transfer networks. Advances in neural information processing systems **29**

35. Pei Z, Cao Z, Long M, Wang J (2018) Multi-adversarial domain adaptation. In: Thirty-second AAAI Conference on Artificial Intelligence

36. Long M, Cao Z, Wang J, Jordan MI (2018) Conditional adversarial domain adaptation. Advances in neural information processing systems **31**

37. Long M, Zhu H, Wang J, Jordan MI (2017) Deep transfer learning with joint adaptation networks. In: International Conference on Machine Learning, pp. 2208–2217. PMLR

38. Zhang W, Ouyang W, Li W, Xu D (2018) Collaborative and adversarial network for unsupervised domain adaptation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 3801–3809

39. Saito K, Watanabe K, Ushiku Y, Harada T (2018) Maximum classifier discrepancy for unsupervised domain adaptation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 3723–3732

40. Saenko K, Kulis B, Fritz M, Darrell T (2010) Adapting visual category models to new domains. In: European Conference on Computer Vision, pp. 213–226. https://doi.org/10.1007/978-3-642-15561-1_16. Springer

41. Caputo B, Müller H, Martinez-Gomez J, Villegas M, Acar B, Patricia N, Marvasti N, Üsküdarlı S, Paredes R, Cazorla M, *et al.* (2014) Imageclef 2014: Overview and analysis of the results. In: International Conference of the Cross-Language Evaluation Forum for European Languages, pp. 192–211. Springer

42. Venkateswara H, Eusebio J, Chakraborty S, Panchanathan S (2017) Deep hashing network for unsupervised domain adaptation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 5018–5027

43. Oza P, Sindagi VA, Sharmini VV, Patel VM (2023) Unsupervised domain adaptation of object detectors: A survey. IEEE Transactions on Pattern Analysis and Machine Intelligence

44. Huang J, Gretton A, Borgwardt K, Schölkopf B, Smola A (2006) Correcting sample selection bias by unlabeled data. Advances in neural information processing systems **19**

45. Chen S, Zhou F, Liao Q (2016) Visual domain adaptation using weighted subspace alignment. In: 2016 Visual Communications and Image Processing (VCIP), pp. 1–4. https://doi.org/10.1109/VCIP.2016.7805516. IEEE

46. Sun B, Feng J, Saenko K (2016) Return of frustratingly easy domain adaptation. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 30

47. Zellinger W, Grubinger T, Lughofer E, Natschläger T, Saminger-Platz S (2017) Central moment discrepancy (cmd) for domain-invariant representation learning. arXiv preprint arXiv:1702.08811

48. Tzeng E, Hoffman J, Zhang N, Saenko K, Darrell T (2014) Deep domain confusion: Maximizing for domain invariance. arXiv preprint arXiv:1412.3474

49. Du Z, Li J, Su H, Zhu L, Lu K (2021) Cross-domain gradient discrepancy minimization for unsupervised domain adaptation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 3937–3946

50. Xu Y, Cao H, Mao K, Chen Z, Xie L, Yang J (2022) Aligning correlation information for domain adaptation in action recognition. IEEE Trans Neural Netw Learn Syst. https://doi.org/10.1109/TNNLS.2022.3212909

51. Saito K, Ushiku Y, Harada T (2017) Asymmetric tri-training for unsupervised domain adaptation. In: International Conference on Machine Learning, pp. 2988–2997. PMLR

52. Du Y, Zhou D, Xie Y, Lei Y, Shi J (2023) Prototype-guided feature learning for unsupervised domain adaptation. Pattern Recognit 135:109154. https://doi.org/10.1016/j.patcog.2022.109154

53. Wang R, Wu Z, Weng Z, Chen J, Qi G-J, Jiang Y-G (2022) Cross-domain contrastive learning for unsupervised domain adap-

tation. IEEE Trans Multimedia. https://doi.org/10.1109/TMM.2022.3146744

54. Donahue J, Jia Y, Vinyals O, Hoffman J, Zhang N, Tzeng E, Darrell T (2014) Decaf: A deep convolutional activation feature for generic visual recognition. In: International Conference on Machine Learning, pp. 647–655. PMLR

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.