Iran Journal of Computer Science (2019) 2:219–230
https://doi.org/10.1007/s42044-019-00037-y

ORIGINAL ARTICLE

Discriminative and domain invariant subspace alignment for visual tasks

Samaneh Rezaei[1] · **Jafar Tahmoresnezhad**[1]

Received: 4 December 2018 / Accepted: 24 May 2019 / Published online: 3 June 2019

Abstract
Transfer learning and domain adaptation are promising solutions to solve the problem that the training set (source domain) and the test set (target domain) follow different distributions. In this paper, we investigate the unsupervised domain adaptation in which the target samples are unlabeled whereas the source domain is fully labeled. We find distinct transformation matrices to transfer both the source and the target domains into the disjointed subspaces where the distribution of each target sample in the transformed space is similar to the source samples. Moreover, the marginal and conditional probability disparities are minimized across the transformed source and target domains via a non-parametric criterion, i.e., maximum mean discrepancy. Therefore, different classes in the source domain are discriminated using the between-class maximization and within-class minimization. In addition, the local information of the source and target data including geometrical structures of the data are preserved via sample labels. The performance of the proposed method is verified using various visual benchmarks experiments. The average accuracy of our proposed method on three standard benchmarks is 70.63%. We compared our method against other state-of-the-art domain adaptation methods where the results prove that it outperforms other domain adaptation methods with 22.9% improvement.

Keywords Unsupervised domain adaptation · Global adaptation · Local adaptation · Distinct transformation · Maximum mean discrepancy

1 Introduction

Machine learning (ML) algorithms train a classification model on labeled training data for labeling the unknown test data. These algorithms show high performance where the amount of the training data is enough and the distributions of the training and test data are similar. However, in visual classification tasks, gaining sufficient labeled data is hard due to the difference in positions and angels of cameras and image statistics. This disparity is known as distribution difference, domain shift or cross-domain problem in visual applications. Thus, a trained ML classifier with the first group of images, would perform poorly to predict the labels of the second group of images [1,2].

A naive solution for cross-domain problem is the manual labeling of the samples to retrain the classifier, but in most cases, it is labor intensive and expensive. Domain adaptation is a recent solution to overcome cross-domain problems. Domain adaptation aims to decrease the distribution difference across the training and test datasets, to improve the model performance. Based on the existence of the labeled target data, domain adaptation is divided into semi-supervised and unsupervised problems. In semi-supervised problems, there is a little labeled data of target domain while the source domain is fully labeled. In unsupervised problems, the source data are fully labeled, and the target data are completely unlabeled. In most of real-world visual problems, no labeled data from target domains are available, therefore, we focus on unsupervised domain adaptation (DA) in this paper.

DA methods obey from three types of strategies [3,4]. Model-based methods adapt the learned parameters of the source-based model for the target domain [5]. Instance-based methods reweight the source samples, which are distributed most similarly to the target ones [6]. Feature-based methods change the feature space of domains to make the source and target domains closer to each other [7].

✉ Jafar Tahmoresnezhad
j.tahmores@it.uut.ac.ir; tahmores@gmail.com

[1] Faculty of IT and Computer Engineering, Urmia University of Technology, Urmia, Iran

Feature-based methods are divided into two lines of approaches including data-alignment and subspace-alignment [3]. Data-alignment methods seek a unified projection matrix to transfer the source and target domains into a shared subspace where the distribution discrepancy across domains is degraded. These methods use shared features across the source and target domains to find a common subspace with minimum distribution difference. The second line of approaches seek a unified projection matrix for each domain using shared and domain-specific features.

In this paper, we propose a subspace-alignment method that adapts the source and target domains while preserving the geometrical and statistical information of data, namely Discriminative and Invariant Subspace Alignment for visual domains (DISA). DISA finds a unified independent subspace for each domain via statistical and geometrical adaptation. (1) In the statistical adaptation, DISA reduces the marginal and conditional distribution differences across the source and target domains via a popular non-parametric method, namely maximum mean discrepancy (MMD) [8], (2) In addition, the statistical information is used to increase the separability and decision region between different classes such that the intra-class distances are decreased and the inter-class distances are increased, (3) In the geometrical adaptation, DISA uses the class and domain manifold information to improve the discrimination across different classes [9]. Via reducing the distances between samples with the local structures and same labels, DISA adapts the geometrical distributions and discriminative structures across the source and target domains.

Our results on 32 cross-domain visual classification tasks show that DISA outperforms other state-of-the-art domain adaptation methods by adapting the geometrical and statistical distributions, simultaneously.

In the rest of this paper, Sect. 2 reviews the related work of DA field which are published in recent years. Sect. 3 presents our proposed method. Sect. 4 discusses experiments and results. Sect. 5 provides conclusion and future works.

2 Related work

Feature-based methods, as a wide category of DA, change the original feature space of data into the latent feature space on which the distribution discrepancy between the source and target domains is reduced. In this paper, we focus on feature-based methods with data-alignment [10–16] and subspace-alignment trends [17,18].

Visual domain adaptation (VDA) [10] is a data-alignment approach that finds a common latent subspace that jointly reduces the marginal and conditional distribution discrepancies between the source and target domains. VDA uses a domain invariant clustering to maximize the between-class distances and discriminate across various classes. Nevertheless, VDA ignores to preserve the geometrical properties of data in the latent subspace.

Yong et al. proposed low-rank and sparse representation (LRSR) [11] to find an embedded subspace in which each target sample can be reconstructed by source domain samples, linearly. In LRSR, each sample in either source or target domains can be reconstructed by its neighborhoods [19]. So whenever the source and target domains are transferred into a latent subspace with the same distribution, instead of each target sample's neighbors, the same neighbors in the source domain can be used for data reconstruction. In this way, LRSR uses a reconstruction matrix with low-rank and sparse constraint and an error matrix for avoiding negative transfer. However, LRSR does not minimize the distribution discrepancy between the source and target domains. Close yet discriminative domain adaptation (CDDA) [12] seeks a common space which both the marginal and the conditional probability distribution discrepancies are reduced and the inter-class samples are repulsed. Robust data structure aligned closed yet discriminative domain adaptation (RSA-CDDA) [13] improves CDDA to reconstruct the target domain by the source domain using the reconstruction matrix with low-rank and sparse constraint in a unified framework. Liu et al. proposed coupled local-global adaptation (CLGA) [14] which adapts the distribution shift across the source and target domains with global and local adaptations. CLGA for global adaptation, reduces the marginal and conditional distribution disparities across the source and target domains using MMD. CLGA builds a graph structure which benefits from the label and manifold structures of domains. Finally, both the local and the global adaptations formulated in a unified optimization problem.

Joint geometrical and statistical alignment (JGSA) [17] is a subspace alignment approach that aims to find a unified subspace for each domain by exploiting the shared and domain-specific features. JGSA reduces the marginal and conditional probability disparities across the embedded subspaces while the source domain information is preserved. In despite of the source samples' information preservation, JGSA does not use the source labels during the geometrical distribution adaptation. Subspace distribution alignment (SDA) [18] aims to transfer the source and target domains into a respective feature space by principal component analysis (PCA) [20]. PCA changes the variance of each dimension in new subspaces, where the source and target domains have different variances along the new bases. SDA uses a subspace transformation matrix to map the source subspace onto the target subspace, using a matrix to align the distributions across domains. SDA describes each distribution by its mean and variance and intends means to zero and aligns the variances between the source and target domains. SDA fails when the covariate shift across domains is large.

In this paper, DISA as a subspace-based method is proposed to transfer the source and target domains into the related embedded subspaces. DISA globally adapts both domains via minimizing the conditional and marginal distribution discrepancies across domains. Moreover, DISA discriminates different classes via minimizing the inter-class and maximizing the intra-class distances. For local adaptation, DISA preserves geometrical information of both domains. Then, the model trained on the source domain is used to predict the target domain labels.

3 Discriminative and invariant subspace-alignment

3.1 Motivation

The main idea of DISA is demonstrated in Fig. 1. DISA iteratively adapts the source and target domains by preserving the statistical and geometrical information of samples.

3.2 Problem statement

We begin with domain and task definitions and then the problem statement will be expressed.

A domain D is included of two parts i.e., $D = \{X, P(x)\}$, with $X = \{x_1, x_2, \ldots, x_{n_s}\}$ and $x \in X$. n_s is the number of the source samples.

Given a specific domain D, a task is defined as $T = \{Y, f(x)\}$ in which Y is the label set of samples. $f(x)$ is a classifier that assigns the sample's label. $f(x)$ is interpreted as conditional probability distribution for each sample, i.e. $f(x) = P(y \mid x)$ where $y \in Y$. We are given a source domain $D_S = \{x_i^s, y_i^s\}_{i=1}^{n_s}$ with n_s labeled source samples and a target domain $D_T = \{x_i^t\}_{i=1}^{n_t}$ with n_t unlabeled target samples. The assumption is that the distribution of the source and target domains are different, specifically $X_s = X_t$, $Y_s = Y_t$, $P_s(x_s) \neq P_t(x_t)$, $P_s(y_s \mid x_s) \neq P_t(y_t \mid x_t)$. Our problem is to find a latent feature spaces for both the source and the target samples that: (1) the marginal and conditional distribution disparities between the source and target domains are reduced, (2) the between-class distances and inter-class distances are maximized and minimized, respectively, and (3) the geometrical information of the source and target samples are preserved.

3.3 DISA

In this section, DISA is introduced to find the projection matrices P_1 and P_2 for projecting the source and target domains, respectively. DISA decreases the domain shift between the source and target domains in both the statistical and the geometrical adaptations.

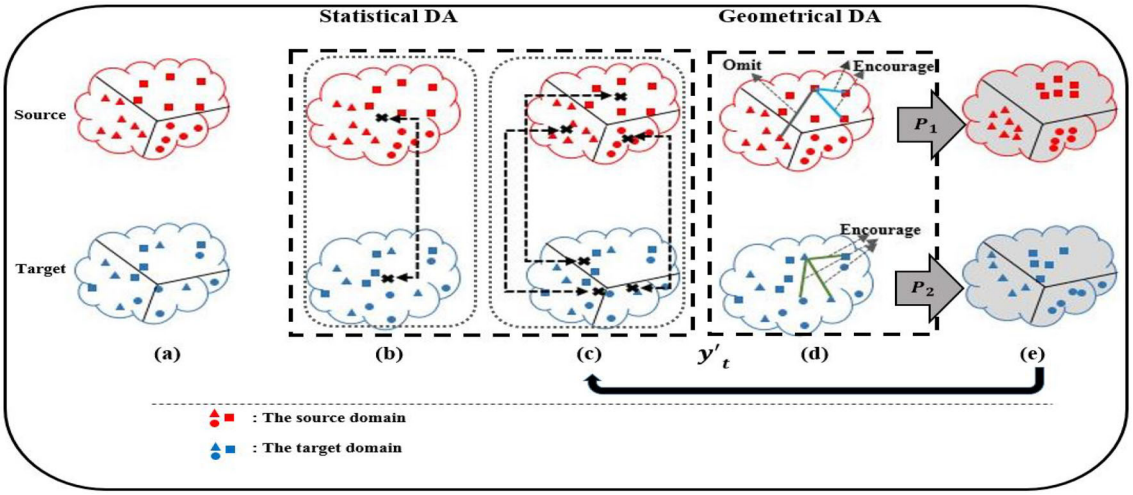

Fig. 1 Illustration of discriminative and invariant subspace alignment method (best viewed in color). **a** Red and blue colors declare the source and target domains, respectively. Different shapes in each domain illustrate different classes. Clouds with white background demonstrate the original feature space. DISA finds two projection matrices, P_1 and P_2 and embeds both domains into the latent subspaces. Gray clouds illustrate latent feature spaces. **b** DISA measures the marginal distribution differences across the source and target domains via MMD. **c** DISA measures class conditional differences across domains via MMD. **d** The local adaptation of source samples is happened by encouraging samples in the same class and also omitting the connections between samples of different classes. For target samples, the local adaptation occurs by encouraging samples of all classes. **e** The latent subspaces is where both the local and the global distributions are aligned and the samples of different classes in each domain are properly isolated each other. Therefore, DISA decreases the domain shift across the source and target domains. A classifier is trained on mapped source samples and is used for updating the labels of the target samples. DISA improves the prediction performance in an iterative algorithm

3.3.1 (Statistical adaptation) Target reconstruction error minimization

We encourage to reduce the mapped target samples' reconstruction error by maximizing the variance of the target domain in the respective embedded subspace. Based on PCA method, the variance maximization can be formulated as follows [20]:

$$\max_{P_2^T P_2 = I} tr(P_2^T X_t H_t X_t^T P_2) \tag{1}$$

where $H_t = I_t - \frac{1}{n_t} 1_t 1_t^T$ is the centering matrix that preserves the target samples' variance information, $1_t \in \mathbb{R}^{n_t}$ is the column one vector and $X_t \in \mathbb{R}^{m \times n_t}$, that m is the original dimension of the target domain. PCA aims to find a projecting matrix P_2 to maximize the data variance.

3.3.2 (Statistical adaptation) Source statistical discriminant learning

Since the source samples in different classes have various structures, the model trained with the source domain likely predicts the imprecise labels for target samples. To improve the performance of the prediction model, we project the source samples into a latent subspace where the different classes are discriminated. Thus, we maximize the between-class distances and minimize the within-class distances according to the following relation:

$$\min_{P_1} \frac{tr\left(P_1^T \left(\sum_{c=1}^{C} n_s^c (m_s^c - \bar{m_s})(m_s^c - \bar{m_s})^T\right)\right)}{tr\left(P_1^T \left(\sum_{c=1}^{C} X_s^c H_s^c (X_s^c)^T\right) P_1\right)} \tag{2}$$

where $X_s^c \in \mathbb{R}^{m \times n_s^c}$ contains the subset of the source samples of class c, $m_s^c = \frac{1}{n_s^c} \sum_{i=1}^{n_s^c} x_i^c$ and n_s^c are the mean of the samples and the number of samples in X_s^c, respectively, $\bar{m_s} = \frac{1}{n_s} \sum_{i=1}^{n_s} x_i$ is the mean of source samples in X_s. $H_s^c = I_s^c - \frac{1}{n_s^c} (1_s^c)^T$ is the centering matrix of samples in class c, $I_s^c \in \mathbb{R}^{n_s^c \times n_s^c}$ and $1_s^c \in \mathbb{R}^{n_s^c}$ are identity matrix and the ones vector, respectively.

3.3.3 (Statistical adaptation) Marginal distribution disparity minimization

By projecting the source and target domains into the respective subspaces, the marginal and conditional distribution disparities across domains are not reduced. To measure the marginal distribution difference, we use the non-parametric method MMD for computing distances between the mapped source and target domains. While the empirical distance between the sample means of the source and target domains in the Hilbert space is computed, DISA finds two projections P_1 and P_2 for each domain as follows:

$$D_1(X_s, X_t) = \left\| \frac{1}{n_s} \sum\nolimits_{x_i \in X_s} P_1^T x_i - \frac{1}{n_t} \sum\nolimits_{x_j \in X_t} P_2^T x_j \right\|^2 \tag{3}$$

The above equation is rewritten in closed form as follows:

$$D_1(X_s, X_t) = min_{P_1, P_2} tr([P_1^T P_2^T] X M_0 X^T \begin{bmatrix} P_1 \\ P_2 \end{bmatrix}) \tag{4}$$

where $X = [X_s, X_t] \in \mathbb{R}^{m \times (n_s + n_t)}$ and $M_0 = \begin{bmatrix} (M_0)_s & (M_0)_{st} \\ (M_0)_{ts} & (M_0)_t \end{bmatrix} \in \mathbb{R}^{(n_s+n_t) \times (n_s+n_t)}$ is MMD coefficient matrix that $(M_0)_s = \frac{1}{n_s^2}$ and $(M_0)_{st} = -\frac{1}{n_s n_t}$ and $(M_0)_{ts} = -\frac{1}{n_t n_s}$ and $(M_0)_t = \frac{1}{n_t^2}$. Also, $tr(.)$ is the trace of a matrix.

3.3.4 (Statistical adaptation) Conditional distribution disparity minimization

Decreasing the marginal distribution disparity across domains, does not guarantee that the class-conditional distribution disparity across the source and target domains is also reduced. We alleviate class-conditional distribution across domains by computing the empirical distances between the source and target domains in each class. Thus, DISA finds two projections P_1 and P_2 for each domain as follows:

$$D_2(X_s, X_t) = \sum_{c=1}^{C} \left\| \frac{1}{n_s^c} \sum\nolimits_{x_i \in X_s^c} P_1^T x_i - \frac{1}{n_t^c} \sum\nolimits_{x_j \in X_t^c} P_2^T x_j \right\|^2 \tag{5}$$

where n_s^c and n_t^c are the number of the source and target samples belonging to class c, respectively. Eq. 5 can be formulated in close form as follows:

$$D_2(X_s, X_t) = min_{P_1, P_2} tr([P_1^T P_2^T] X M_c X^T \begin{bmatrix} P_1 \\ P_2 \end{bmatrix}) \tag{6}$$

where $M_c = \begin{bmatrix} (M_c)_s & (M_c)_{st} \\ (M_c)_{ts} & (M_c)_t \end{bmatrix} \in \mathbb{R}^{(n_s+n_t) \times (n_s+n_t)}$ is MMD coefficient matrix that it is computed as follows, $(M_c)_s = \frac{1}{(n_s^c)^2}$ and $(M_c)_{st} = -\frac{1}{n_s^c n_t^c}$ and $(M_c)_{ts} = -\frac{1}{n_t^c n_s^c}$ and $(M_c)_t = \frac{1}{(n_t^c)^2}$. Because the target labels are unknown, we utilize the pseudo-labels of the target samples, which can be labeled via a base classifier, i.e., nearest neighbor (NN) [21]. We train an NN classifier on the labeled source samples to label the target samples, initially. As the source and target samples have different structures, at first, target samples are imprecise where DISA in an iterative manner refines them.

3.3.5 (Geometrical adaptation) Discriminant learning through source samples

Statistical information of samples, i.e., the sample mean and variance are not sufficient for minimizing the domain shift across the source and target domains due to the existence of shift in intrinsic structures of the mapped domains. Thus, DISA uses the local information of samples for geometrical distribution adaptation [22]. Based on the manifold theorem [23], samples with near geometrical distribution, have same labels. Therefore, DISA adapts the geometrical distribution across domains.

Since the source samples are fully labeled, DISA benefits from the manifold and label structures of the source samples, simultaneously, to preserve the information of the intrinsic structure of data and increase the separability of the source samples in the latent space. In this way, we create a graph structure with n_s nodes where each node corresponds to one sample in X_s. And then, for each node, we consider its p nearest neighbors. If the sample pair has the same class and same manifold, we connect them by an edge with a corresponding weight. The weight between the sample pair, i and j, is measured by $W_{ij} = e^{-\|x_i - x_j\|^2}$ where $W \in \mathbb{R}^{n_s \times n_s}$ is the weight matrix that contains the Euclidean distance between each sample pairs. $L = D - W$ is the graph Laplacian matrix, where $D_{ii} = \sum_{j=1}^{n_s} W_{ij}$ is a diagonal matrix. To discriminate the mapped source samples in the embedded space, DISA finds P_1 to minimize distances between the source samples in each class based on the manifold structure as follows:

$$min_{P_1} \sum_{i,j=1}^{n_s} (P_1^T x_i - P_1^T x_j) W_{s_{ij}} = min_{P_1} tr(P_1^T X_s L_s X_s^T P_1) \tag{7}$$

L_s, as the Laplacian matrix of the source data, contains geometrical information of the source samples. Thus, P_1 maps the source samples with similar geometrical distribution into the related subspace and close to each other.

3.3.6 (Geometrical adaptation) Manifold learning through target samples

To preserve the geometrical structure of the target samples, DISA builds a nearest neighbor graph with n_t nodes on the target domain. Since there is no target sample with corresponding label, we connect each node with its p nearest neighbors, based on manifold theorem. So, DISA finds P_2 as follows:

$$min_{P_2} = \sum_{i,j=1}^{n_t} (P_2^T x_i - P_2^T x_j) W_{t_{ij}} = min_{P_2} tr(P_2^T X_t L_t X_t^T P_2) \tag{8}$$

The Laplacian matrix of the target samples, L_t, includes geometrical information of the target data. Indeed, P_2 transforms the target samples into the related latent subspaces, though, the local information of samples are preserved.

3.3.7 (Geometrical adaptation) Divergence minimization along subspace generation

DISA finds couple projections for the source and target domains via statistical and geometrical learning. To divergence minimization of data in the embedded subspaces, DISA makes closer both latent subspaces P_1 and P_2. Thus, both the local and the global information of domains are preserved during subspace generation [17]. This aim is formulated as follows:

$$\min_{P_1, P_2} \|P_1 - P_2\|_F^2 \tag{9}$$

where $\|.\|_F^2$ is the Frobenius norm.

3.4 Optimization problem

DISA finds the source and target projections, P_1 and P_2, via combination of equations (1) through (9). We develop the final model for aligning the cross-domain shift based on Rayleigh quotient [24] as follows:

$$min_{P_1, P_2} \frac{tr(P^T \begin{bmatrix} \beta S_w + \gamma X_s L_s X_s^T + \lambda I & -\lambda I \\ -\lambda I & \gamma X_t L_t X_t^T + (\lambda + \mu) I \end{bmatrix} + X(\sum_{c=0}^{C} M_c) X^T) P)}{tr(P^T \begin{bmatrix} \beta S_b & 0(m,m) \\ 0(m,m) & \mu S_t \end{bmatrix} P)} \tag{10}$$

where $S_w = \sum_{c=1}^{C} n_s^c (m_s^c - \bar{m_s})(m_s^c - \bar{m_s})^T$ and $S_b = \sum_{c=1}^{C} X_s^c H_s^c (X_s^C)^T$ learn source domain discriminative information. $\sum_{c=0}^{C} M_c$ for $c = 0$ computes MMD coefficient matrix for marginal distribution and for $c = 1$ through C, computes MMD coefficient matrix in conditional distribution mode. $X = \begin{bmatrix} X_s & 0(m, n_t) \\ 0(m, n_s) & X_t \end{bmatrix}$ and $I \in \mathbb{R}^{m \times m}$ is the identity matrix. In addition, $S_t = X_t H_t X_t^T$ searches for matrix P_2 to maximize the target domain variances in the relative embedded subspace. $P = \begin{bmatrix} P_1 \\ P_2 \end{bmatrix} \in \mathbb{R}^{2m \times k}$ where P_1 and P_2 are given by eigenvalue decomposition of Eq. 10. Algorithm 1 is designed for DISA's strategy implementation.

3.5 Time complexity

In this section, we analyze the computational complexity of Algorithm 1, as follows: For steps 3 and 4, the time complexity of training a classifier on the source data and predicting the pseudo labels of the target samples is $O(mn_s)$ and $O(mn_t)$, respectively. For steps 5 and 6, constructing the marginal and conditional matrices as the global adaptation stages needs $O((n_s + n_t)^2)$. As the local adaptation stages in steps 7 and 8, computing the Laplacian matrices for the source and target samples, needs $O((n_s)^2)$ and $O((n_t)^2)$, respectively. Time

Algorithm 1 Discriminative and Invariant Subspace Alignment for visual domains (DISA)

Input: source data X_s, target data X_t, source domain labels Y_s, regularization parameters: λ, μ, k, T, β
Output: projection matrices P_1 and P_2, target domain labels y_t
learn 1-NN classifier, f on (X_s, Y_s)
predict the pseudo labels in target domain (X_t), Y_{t_0}, by classifier f
construct M_0 by Eq. 4
construct M_c by Eq. 5
compute L_s by Eq. 7
compute L_t by Eq. 8
repeat until convergence
    solve Eq. 10 and select the k smallest eigenvectors as P_1 and P_2
    learn the classifier f on $(P_1^{\mathrm{T}} X_s, Y_s)$
    update pseudo labels, Y_{t_0}, on $(P_2^{\mathrm{T}} X_t)$
    update matrix M_c according to Eq. 5
end repeat
learn final classifier f' on $(P_1^{\mathrm{T}} X_s, Y_s)$
predict labels of X_t using f'
return target domain labels y_t determined by classifier f'

complexity to compute the coupled transformation matrices in Step 10 is $O(k^3)$, where k is the number of the latent features. For steps 11 and 12, training an NN classifier on the mapped source samples and predicting the mapped target samples need $O(kn_s)$ and $O(kn_t)$, in turn. In summary, the computational complexity of DISA is $O((n_s + n_t)^2 + k^3)$.

4 Experiments

In this section, we evaluate the effectiveness of our proposed method in comparison with baseline methods and other state-of-the-art domain adaptation methods for image classification tasks.

4.1 Benchmarks description

The Office dataset [25] is the most popular object recognition benchmark, which consists of 4652 real-world images that divided into 3 different domains: Amazon (A: images downloaded from amazon.com), DSLR (D: high-resolution images captured by SLR camera) and Webcam (W: low-resolution images captured by a web camera), each domain has 31 categories, including camera, hat, bike, keyboard, scissors. Caltech-256 [26] consists of 30607 images with 256 categories. We select 10 common classes of four-domains, i.e., calculator, laptop, keyboard, mouse, monitor, projector, headphones, backpack, mug and bike. The configuration of our dataset is the same as [24], where 800-dimensional SURF features [27] are extracted from images and then normalized by z-score as input vectors. Finally, every two different domains are denoted as the source and target domains, where 12 domain adaptation tasks can be constructed, i.e., $C \longrightarrow A, C \longrightarrow W, \ldots, D \longrightarrow W$. CMU-PIE [28] is a face recognition benchmark, which consists of 41368 grayscale images of 68 people with different illumination and poses. We use five poses as the same as [24], P1, P2, P3, P4, and P5 where the images are with left-side, upper-side, down-side, front-side and right-side poses, respectively. Since CMU-PIE dataset consists of five domains, 20 cross-domain tasks can be constructed, i.e., $P1 \longrightarrow P2, P1 \longrightarrow P3, \ldots, P5 \longrightarrow P4$. MNIST [29] and USPS [30] are digit recognition benchmarks, which consist of 2000 and 1800 grayscale images, respectively. MNIST (M) images were collected from mixed American high school students and American Census Bureau employees. USPS (U) dataset consist of handwritten digit scanned from envelopes of the US Postal Service images. We use 10 common classes of both datasets as the same as [24]. Since Digit benchmark has two domains, two domain adaptation tasks can be constructed, i.e., $U \longrightarrow M$ and $M \longrightarrow U$ (Fig. 2).

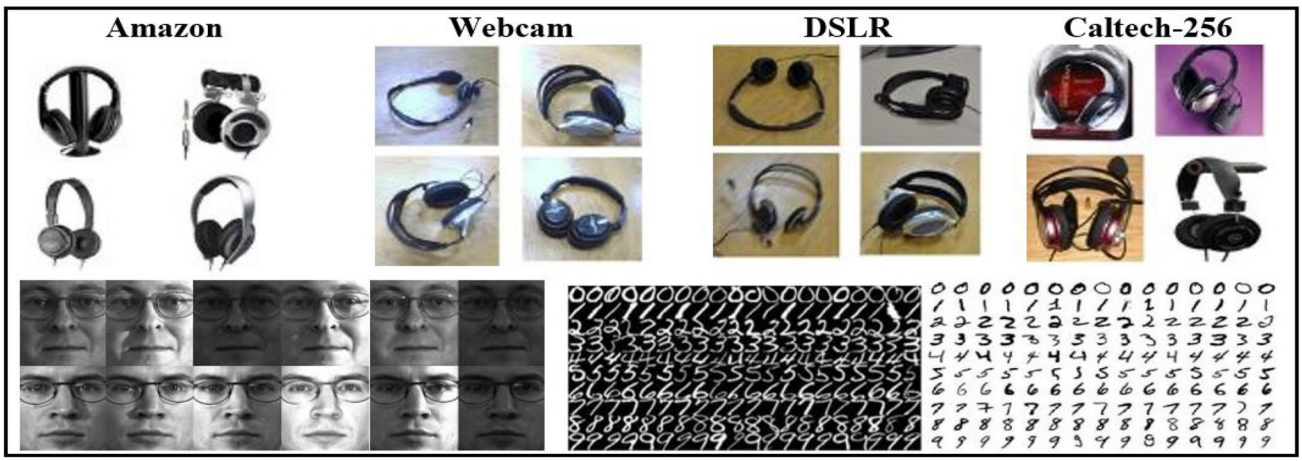

Fig. 2 The first row demonstrates the Office+Caltech-256 dataset, and the second row demonstrates CMU-PIE and MNIST and USPS datasets, respectively (from left to right)

Table 1 Accuracy (%) on 12 pairs of source/target domains on Office+Caltech-256 datasets

Dataset	NN	JDA (2013)	SA (2013)	TJM (2014)	LRSR (2016)	VDA (2017)	CDDA (2017)	CLGA (2018)	DISA
$C \longrightarrow A$	23.60	44.78	49.27	46.76	51.25	46.14	48.33	48.02	**57.93**
$C \longrightarrow W$	25.76	41.69	40.00	39.98	38.64	46.10	44.75	42.37	**49.15**
$C \longrightarrow D$	25.48	45.22	39.49	44.59	47.13	**51.59**	48.41	49.04	49.04
$A \longrightarrow C$	26.00	39.36	39.98	39.45	**43.37**	42.21	42.12	42.3	39.36
$A \longrightarrow W$	29.83	37.97	33.22	42.03	36.61	**51.19**	41.69	41.36	50.51
$A \longrightarrow D$	25.48	39.49	33.76	45.22	38.85	48.41	37.58	36.31	**50.96**
$W \longrightarrow C$	19.86	31.17	**35.17**	30.19	29.83	27.6	31.97	32.95	34.02
$W \longrightarrow A$	22.96	32.78	39.25	29.96	34.13	26.1	37.27	34.57	**42.48**
$W \longrightarrow D$	59.24	89.17	75.16	89.17	82.8	89.18	87.9	**92.36**	90.45
$D \longrightarrow C$	26.27	31.52	34.55	31.43	31.61	31.26	**34.64**	33.66	32.15
$D \longrightarrow A$	28.50	33.09	39.87	32.78	33.19	37.68	33.51	**89.83**	39.35
$D \longrightarrow W$	63.39	89.49	76.95	85.42	77.29	90.85	90.51	35.99	**93.22**
Average	31.37	46.31	44.72	46.42	45.39	49.03	48.22	48.23	**52.38**

Bold results in the Table 1 show the best results in comparison with the results of the other methods in the specified domain adaptation task

Table 2 Accuracy (%) on 20 pairs of source/target domains on CMU-PIE dataset

Dataset	NN	JDA (2013)	SA (2013)	TJM (2014)	LRSR (2016)	VDA (2017)	CDDA (2017)	CLGA (2018)	DISA
$P1 \longrightarrow P2$	26.09	58.81	41.62	23.87	65.87	72.99	60.22	67.83	**77.29**
$P1 \longrightarrow P3$	26.59	54.23	49.45	28.86	64.09	61.64	58.7	63.85	**74.69**
$P1 \longrightarrow P4$	30.67	84.5	64.52	43.37	82.03	90.12	83.48	88.95	**91.35**
$P1 \longrightarrow P5$	16.67	49.75	39.4	19.3	54.9	42.4	54.17	61.76	**63.30**
$P2 \longrightarrow P1$	24.49	57.62	43.34	26.14	45.04	72.87	62.33	71.4	**80.25**
$P2 \longrightarrow P3$	46.63	62.93	57.72	37.93	53.49	75.61	64.64	72.98	**81.31**
$P2 \longrightarrow P4$	54.07	75.82	67.47	50.53	71.43	83.6	79.9	86.24	**90.90**
$P2 \longrightarrow P5$	26.53	39.89	41.3	21.63	47.97	57.72	44	51.23	**69.91**
$P3 \longrightarrow P1$	21.37	50.96	44.48	28.66	52.49	58.76	58.46	70.17	**81.12**
$P3 \longrightarrow P2$	41.01	57.95	57.58	35.97	55.56	74.65	59.73	73.48	**81.52**
$P3 \longrightarrow P4$	46.53	68.45	66.99	51.97	77.5	87.53	77.2	89.31	**93.45**
$P3 \longrightarrow P5$	26.23	39.95	48.41	25.31	54.11	52.63	47.24	55.51	**75.37**
$P4 \longrightarrow P1$	32.95	80.58	66.69	45.71	81.54	92.35	83.1	89.56	**94.18**
$P4 \longrightarrow P2$	62.68	82.63	76.67	57.58	58.39	92.27	82.26	92.94	**95.89**
$P4 \longrightarrow P3$	73.22	87.25	80.88	71.63	82.23	90.38	86.64	93.08	**95.04**
$P4 \longrightarrow P5$	37.19	54.66	54.72	30.94	72.61	69.98	58.33	71.63	**82.60**
$P5 \longrightarrow P1$	18.49	46.46	32.74	27.13	52.19	49.91	48.02	57.68	**63.90**
$P5 \longrightarrow P2$	24.19	42.05	37.08	22.65	49.41	62.31	45.61	55.43	**73.36**
$P5 \longrightarrow P3$	28.31	53.31	44.12	28.86	58.45	61.27	52.02	58.03	**76.78**
$P5 \longrightarrow P4$	31.24	57.01	48.48	32.59	64.31	71.19	55.99	71.85	**76.74**
Average	34.76	60.24	53.18	35.53	63.53	71	63.1	72.15	**80.95**

Bold results in the Table 2 show the best results in comparison with the results of the other methods in the specified domain adaptation task

4.2 Implementation details

The evaluation measurement used for DISA to compare its performance with other state-of-the-art methods is accuracy, as the same with [20] as follows:

$$\text{Accuracy} = \frac{|x : x \in X_t \bigwedge f(x) = y(x)|}{n_t} \tag{11}$$

where $f(x)$ is the predicted label for each target sample x and $y(x)$ is the true label of it. DISA has five parameters: λ, μ, k, T, β and γ. We empirically set $\lambda = 0.5$, $\mu = 5$, $\beta = 0.1$ and

Table 3 Accuracy (%) on 2 pairs of source/target domains on USPS+MNIST datasets

Dataset	NN	JDA (2013)	SA (2013)	TJM (2014)	LRSR (2016)	VDA (2017)	CDDA (2017)	CLGA (2018)	DISA
$U \longrightarrow M$	44.7	59.65	67.78	52.25	54.51	62.95	62.05	58.35	**69.90**
$M \longrightarrow U$	65.94	67.28	48.8	63.28	73.82	74.95	76.22	71.28	**83.89**
Average	55.32	63.47	58.29	57.77	64.17	68.95	69.14	64.81	**76.89**

Bold results in the Table 3 show the best results in comparison with the results of the other methods in the specified domain adaptation task

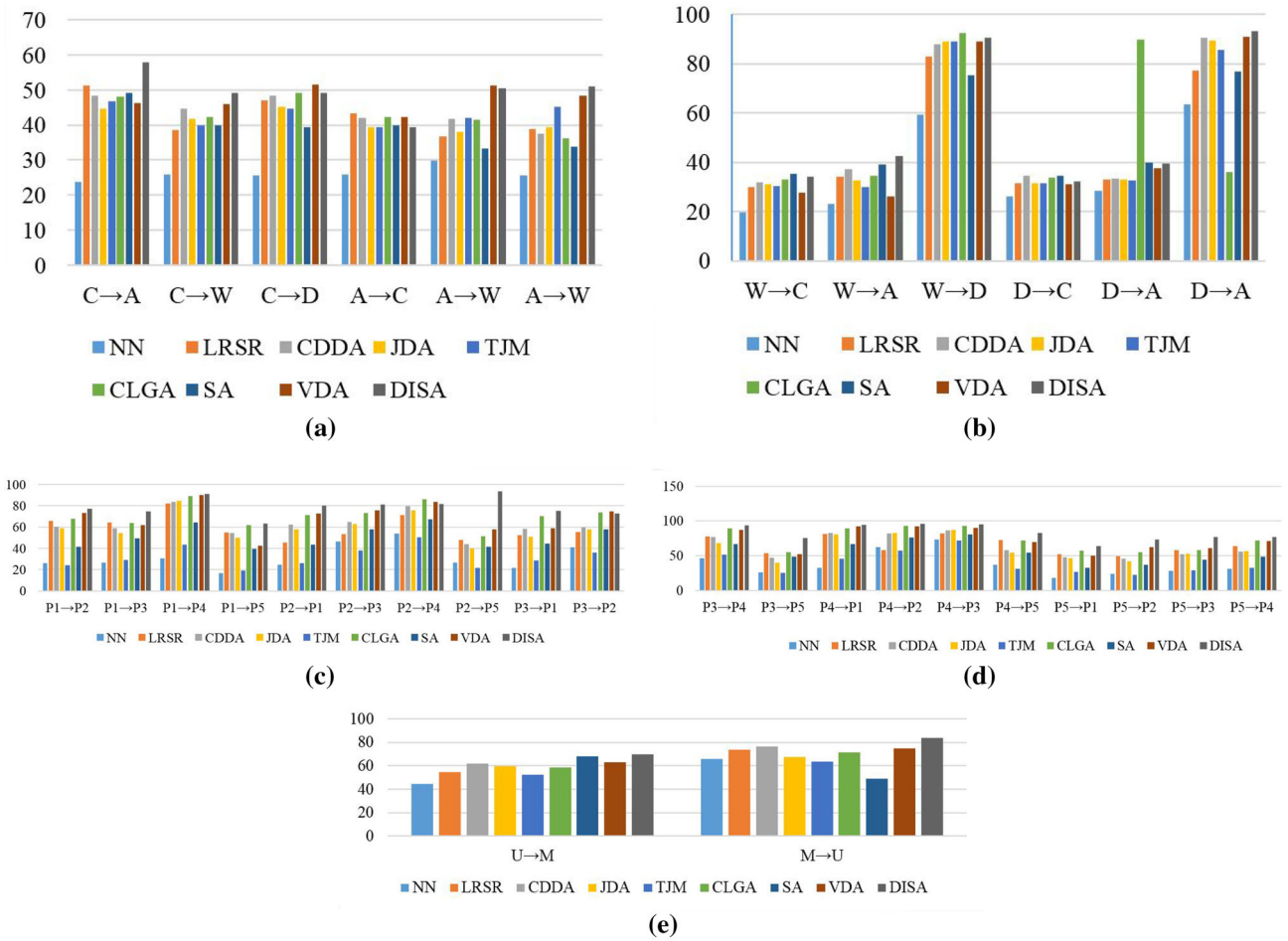

Fig. 3 Accuracy (%) of DISA in comparison against cross-domain approaches. **a**, **b** Office+Caltech-256, **c**, **d** CMU-PIE, **e** USPS+MNIST (best viewed in color)

$\gamma = 0.1$ for Office+Caltech-256. For CMU-PIE dataset, we set $\lambda = 5$, $\mu = 0.5$, $\beta = 0.0005$ and $\gamma = 0.001$ Also, we set $\lambda = 1$, $\mu = 5$, $\beta = 0.001$ and $\gamma = 0.05$ for Digits datasets. The number of subspaces, k, for Office+Clatech-256 datasets is set to 30 and for Digits datasets is set to 110. The number of iterations in both benchmarks, T, for DISA's convergence is set to 10.

4.3 Experimental results evaluation and discussion

The performance of DISA and other nine compared methods (NN [21], JDA [24], SA [31], TJM [32], LRSR [11], VDA [10], CDDA [12], CLGA [14]) on three benchmark datasets, consist of Office+Caltech-256, CMU-PIE and Digits are shown in Tables 1, 2 and 3, respectively. The highest accuracy for each cross-domain adaptation pairs is highlighted in bold. We compare DISA with the best-reported results of other methods based on the reported results in their papers. All proposed methods use the NN-classifier to train on the labeled source samples and predict the labels of target samples. Based on results shown in Table 1, DISA achieves (3.35%) improvement compared to VDA, best method among state-of-the-art methods, in average accuracy on Office+Caltech-256 and outperforms it in 9 out of 12 cross-domain adaptation tasks.

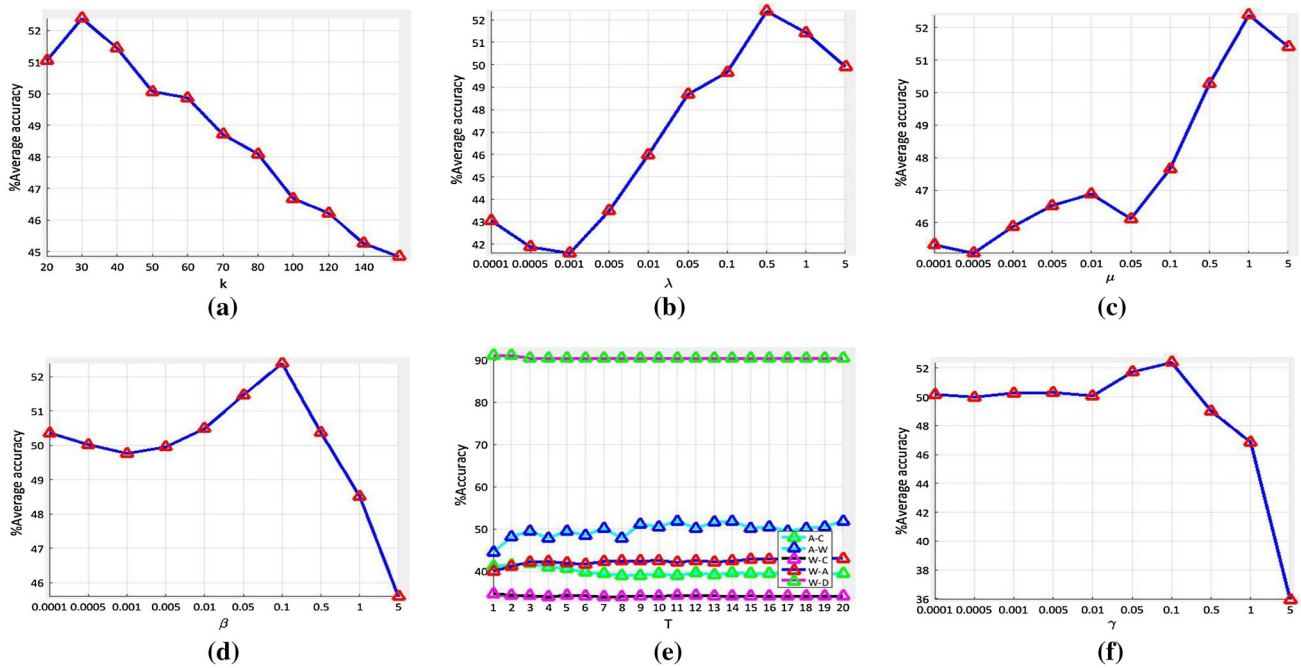

Fig. 4 Parameters sensitivity evaluation on Office+Caltech-256 datasets. **a** The number of subspaces with respect to average accuracy (%), **b** impact of regularization parameter λ with respect to average accuracy (%), **c** sensitivity of parameter μ with respect to average accuracy (%), **d** sensitivity of regularization parameter β with respect to average accuracy (%), **e** the number of iteration, T, with respect to accuracy (%) on $A \longrightarrow C$, $A \longrightarrow W$, $W \longrightarrow C$, $W \longrightarrow A$, $W \longrightarrow D$ experiments, **f** Sensitivity of Laplacian regularization parameter γ with respect to average accuracy (%) on Office+Caltech-256 datasets

DISA has (21.01%) improvement compared to NN which demonstrates that DISA adapts the mismatched source and target domains, effectively.

JDA and TJM map both domains into the shared latent subspace where the marginal distribution discrepancy across domains is decreased. TJM sets weights for source samples with $l_{2,1}$ norm for effectively learn the model and JDA decreases the conditional distribution difference. However, DISA adapts both the marginal and the conditional distribution discrepancies and preserves the local information of samples in the latent subspaces. DISA outperforms JDA with (6.07%), (20.71%) and (13.42%) improvement and TJM with (7.97%), (45.39%) and (19.12%) on Office+Caltech-256, CMU-PIE and Digits databases, respectively.

SA is a subspace-based method that maps the source and target domains into latent subspaces. SA uses a linear transformation matrix to align basis vectors of the mapped source domain with the mapped target domain basis vectors. SA ignores to reduce distribution discrepancy but DISA decreases both the marginal and the conditional distribution differences. DISA gains (7.66%), (27.77%) and (18.6%) improvement against SA on the Office+Caltech-256, CMU-PIE and Digits datasets, respectively.

Although LRSR minimizes the reconstruction error, the distribution discrepancy across domains still remains large. DISA brings both domains closer and its performance against LRSR improves (6.99%), (17.42%) and (12.72%) in average accuracy on the Office+Caltech-256, CMU-PIE and Digits datasets, respectively.

Although VDA adapts domains using statistical information, it ignores to preserve the local information of samples. However, DISA uses both the local and the global information of samples for bringing both domains closer to each other. DISA obtains (3.35%), (9.95%) and (7.94%) improvement against VDA in average accuracy on the Office+Caltech-256, CMU-PIE and Digits datasets, respectively.

CDDA adapts domains statistically and discriminates different classes from each other. DISA decreases the marginal and conditional distribution discrepancies and adapts domains locally. DISA performs (4.16%), (17.85%) and (7.75%) better than CDDA in average accuracy on the Office+Caltech-256, CMU-PIE and Digits datasets, respectively.

CLGA adapts both the source and the target domains in geometrical and statistical adaptations. In the statistical adaptation, CLGA reduces the marginal and conditional distribution differences across domains. In the geometrical adaptation, CLGA uses the manifold and label information of samples to adapt both domains. However, DISA adapts both domains in statistical level, and uses the labels and manifold information of the source domain for local adapta-

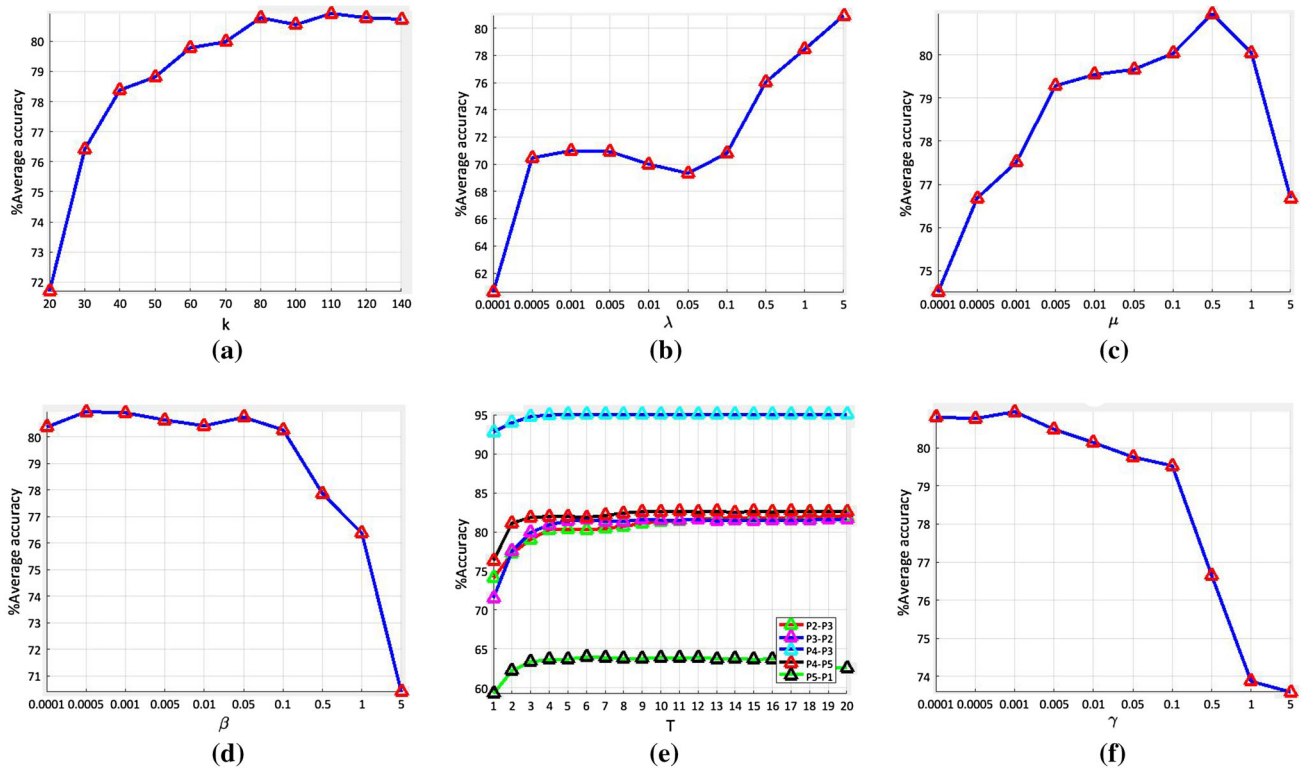

Fig. 5 Parameters sensitivity evaluation on the CMU-PIE dataset. **a** The number of subspaces with respect to average accuracy (%), **b** impact of regularization parameter λ with respect to average accuracy (%), **c** sensitivity of parameter μ with respect to average accuracy (%), **d** sensitivity of regularization parameter β with respect to average accuracy (%), **e** The number of iteration, T, with respect to accuracy (%) on $P2 \longrightarrow P3$, $P3 \longrightarrow P2$, $P4 \longrightarrow P3$, $P4 \longrightarrow P5$ and $P5 \longrightarrow P1$ experiments, **f** Sensitivity of Laplacian regularization parameter γ with respect to average accuracy (%) on CMU-PIE dataset

tion. DISA on Office+Caltech-256 datasets obtains (4.15%) improvement, and in CMU-PIE and Digits datasets obtains (8.80%) and (12.08%) performance improvement against CLGA, respectively.

Based on the Figs. 3a and b, DISA in comparison to VDA (best-compared method on the Office+Caltech-256 datasets), outperforms in 9 out of 12 problems. DISA outperforms NN and JDA (baseline methods) in 12 out of 12 experiments. Figures 3c and d depict that DISA outperforms in 20 out of 20 problems in comparison to CLGA (best-compared method on CMU-PIE datasets). Figure 3e shows that DISA outperforms all other state-of-the-art methods in all cases.

4.4 Impact of parameters

The effectiveness of DISA depends on the optimal values of the parameters. For tuning the optimal value of k, the dimension of latent subspaces, the number of iteration, T, and the regularization parameters, μ, β, λ and γ, we evaluate DISA on each various adaptation tasks. We set the regularization parameters μ, β, λ and γ empirically by searching the wide range [0.0001, 5] and the number of neighbors for computing Laplacian matrix is set to 5. Figure 4a illustrates the experiments on Office+Caltech-256 datasets for evaluating the impact of k. We report the average accuracy of DISA with $k \in [20, 150]$ where $k = 30$ is the optimal dimension for the latent subspaces on Office+Caltech-256 datasets. Figures 4b, c, d and f depict the impact of parameters λ, μ, β and γ on Office+Caltech256 datasets, respectively. In this way, $\lambda = 0.5$, $\mu = 1$, $\beta = 0.1$ and $\gamma = 0.1$ are selected as the optimal parameters for DISA. Figure 4e shows the results of each iteration impact on Office+Caltech-256 datasets. We choose $T = 10$ as the optimal parameter.

Figures 5a, b, c, d, e and f illustrate the impact of different values of parameters $k, \lambda, \mu, \beta, T$ and γ on CMU-PIE dataset, respectively. Thus, $k = 110$, $\lambda = 5$, $\mu = 0.5$, $\beta = 0.0005$, $\gamma = 0.001$ and $T = 10$ are the optimal parameters on CMU-PIE dataset.

Figure 6a shows the results on Digits datasets for evaluating the sensitivity of parameter k. We report the average accuracy of DISA with $k \in [20, 140]$ where $k = 110$ is the optimal dimension for latent subspaces on Digits datasets. Figures 6b, c, d and f show the impact of parameters λ, μ, β and γ on the Digits datasets, respectively. In this way, $\lambda = 1$,

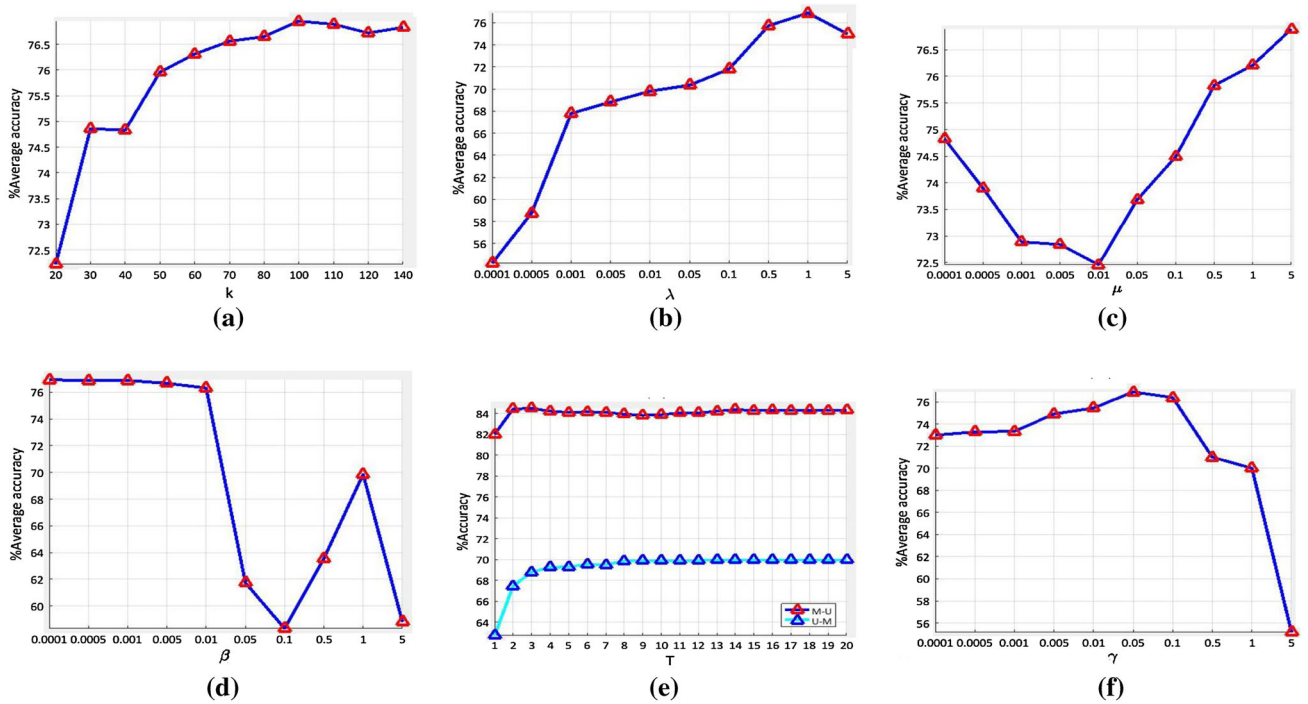

Fig. 6 Parameters sensitivity evaluation on the MNIST+USPS datasets. **a** The number of subspaces with respect to average accuracy (%), **b** Impact of regularization parameter λ with respect to average accuracy (%), **c** Sensitivity of parameter μ with respect to average accuracy (%), **d** Sensitivity of regularization parameter β with respect to average accuracy (%), **e** The number of iteration, T, with respect to accuracy (%) on $M \longrightarrow U$ and $U \longrightarrow M$ experiments, **f** Sensitivity of Laplacian regularization parameter γ with respect to average accuracy (%) on MNIST+USPS datasets

$\mu = 5$, $\beta = 0.001$ and $\gamma = 0.05$ are chosen as the optimal parameters for DISA. Figure 6e shows the iteration sensitivity on the Digits datasets where $T = 10$ is set as the optimal parameter.

5 Conclusion and future work

In this paper, we proposed a novel visual domain adaptation approach, referred to as Discriminative and Invariant Subspace-Alignment (DISA) to tackle shift problem across domains. DISA finds the correlated features of both domains in the latent subspaces to solve the cross-domain problem where it adapts the transformed source and target domains by minimizing the marginal and conditional distribution distances. DISA exploits the manifold information of domains to preserve the geometrical information of samples. We evaluate DISA on various tasks of common visual datasets and the results express the superiority of the proposed method in comparison with other state-of-the-art visual domain adaptation methods.

We intend to extend DISA as an online method for pretending labels of real-time data in visual problems. As a future work, we consider concentrating to embed DISA in missing modality and deep learning problems.

References

1. Patel, V.M., Gopalan, R., Li, R., Chellappa, R.: Visual domain adaptation: a survey of recent advances. IEEE Signal Process Mag. **32**(3), 53–69 (2015)
2. Pan, S.J., Yang, Q.: A survey on transfer learning. IEEE Trans. Knowl. Data Eng. **22**(10), 1345–1359 (2010)
3. Shao, L., Zhu, F., Li, X.: Transfer learning for visual categorization: a survey. IEEE Trans. Neural Netw. Learn. Syst. **26**(5), 1019–1034 (2015)
4. Weiss, K., Khoshgoftaar, T.M., Wang, D.: A survey of transfer learning. J. Big Data **3**(1), 9 (2016)
5. Luo, L., Chen, L., Hu, S., Lu, Y., Wang, X.: Discriminative and geometry aware unsupervised domain adaptation. arXiv:1712.10042 (2017) (preprint)
6. Jing, M., Li, J., Zhao, J., Lu, K.: Learning distribution-matched landmarks for unsupervised domain adaptation. In: International conference on database systems for advanced applications, pp. 491–508. Springer, Cham (2018)
7. Li, S., Song, S., Huang, G., Ding, Z., Wu, C.: Domain invariant and class discriminative feature learning for visual domain adaptation. IEEE Trans. Image Process. **27**(9), 4260–4273 (2018)
8. Gretton, A., Borgwardt, K.M., Rasch, M.J., Schölkopf, B., Smola, A.: A kernel two-sample test. J. Mach. Learn. Res. **13**(Mar), 723–773 (2012)

9. Tahmoresnezhad, J., Hashemi, S.: A generalized kernel-based random k-samplesets method for transfer learning. Iran. J. Sci. Technol. Trans. Electr. Eng. **39**, 193–207 (2015)
10. Tahmoresnezhad, J., Hashemi, S.: Visual domain adaptation via transfer feature learning. Knowl. Inf. Syst. **50**(2), 585–605 (2017)
11. Xu, Y., Fang, X., Wu, J., Li, X., Zhang, D.: Discriminative transfer subspace learning via low-rank and sparse representation. IEEE Trans. Image Process. **25**(2), 850–863 (2016)
12. Luo, L., Wang, X., Hu, S., Wang, C., Tang, Y., Chen, L.: Close yet distinctive domain adaptation. arXiv:1704.04235 (2017) (preprint)
13. Luo, L., Wang, X., Hu, S., Chen, L.: Robust data geometric structure aligned close yet discriminative domain adaptation. arXiv:1705.08620 (2017) (preprint)
14. Liu, J., Li, J., Lu, K.: Coupled local-global adaptation for multi-source transfer learning. Neurocomputing **275**, 247–254 (2018)
15. Tahmoresnezhad, J., Hashemi, S.: Exploiting kernel-based feature weighting and instance clustering to transfer knowledge across domains. Turk. J. Electr. Eng. Comput. Sci. **25**(1), 292–307 (2017)
16. Ding, Z., Fu, Y.: Robust transfer metric learning for image classification. IEEE Trans. Image Process. **26**(2), 660–670 (2017)
17. Zhang, J., Li, W., Ogunbona, P.: Joint geometrical and statistical alignment for visual domain adaptation. arXiv:1705.05498 (2017) (preprint)
18. Sun, B., Saenko, K.: Subspace distribution alignment for unsupervised domain adaptation. In: BMVC (pp. 24-1) (2015)
19. Shao, M., Kit, D., Fu, Y.: Generalized transfer subspace learning through low-rank constraint. Int. J. Comput. Vis. **109**(1–2), 74–93 (2014)
20. Jolliffe, I.: Principal component analysis. In: International encyclopedia of statistical science (pp. 1094–1096). Springer, Berlin (2011)
21. Cover, T., Hart, P.: Nearest neighbor pattern classification. IEEE Trans. Inf. Theory **13**(1), 21–27 (1967)
22. Li, W., Zhang, Z., Liu, Z.: Action recognition based on a bag of 3D points. In: Proc. IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops. IEEE (2010), pp. 9–14
23. Belkin, M., Niyogi, P., Sindhwani, V.: Manifold regularization: a geometric framework for learning from labeled and unlabeled examples. J. Mach. Learn. Res. **7**, 2399–2434 (2006)
24. Long, M., Wang, J., Ding, G., Sun, J., Philip, S.Y.: Transfer feature learning with joint distribution adaptation. In: IEEE International Conference on Computer Vision (ICCV), 2013, pp. 2200–2207 (2013)
25. Saenko, K., Kulis, B., Fritz, M., Darrell, T.: Adapting visual category models to new domains. In: European Conference on Computer Vision, pp. 213–226. Springer, Berlin (2010)
26. Griffin, G., Holub, A., Perona, P.: Caltech-256 object category dataset (2007)
27. Saenko, K., Kulis, B., Fritz, M., Darrell, T.: Adapting visual category models to new domains. In: Proceedings of the European Conference on Computer Vision, pp. 213–226 (2010)
28. Sim, T., Baker, S., Bsat, M.: The CMU pose, illumination, and expression (PIE) database. In: Fifth IEEE International Conference on Automatic Face and Gesture Recognition, 2002, Proceedings, pp. 53–58 (2002)
29. Lecun, Y., Botton, L., Bengio, Y., Haffner, P.: Gradient based learning applied to document recognition. Proc. IEEE **86**(11), 2278–2324 (1998)
30. Hull, J.: A database for handwritten text recognition research. IEEE Trans. Pattern Anal. Mach. Intell. **16**(5), 550–554 (1994)
31. Fernando, B., Habrard, A., Sebban, M., Tuytelaars, T.: Unsupervised visual domain adaptation using subspace alignment. In: Proceedings of the IEEE International Conference On Computer Vision, pp. 2960–2967 (2013)
32. Long, M., Wang, J., Ding, G., Sun, J., Yu, P.S.: Transfer joint matching for unsupervised domain adaptation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1410–1417 (2014)