Signal, Image and Video Processing
https://doi.org/10.1007/s11760-019-01587-1

ORIGINAL PAPER

Joint local and statistical discriminant learning via feature alignment

Elahe Gholenji[1] · **Jafar Tahmoresnezhad**[1]

Received: 11 May 2019 / Revised: 14 September 2019 / Accepted: 15 October 2019

Abstract
Image processing has attracted increasing attention in recent researches to solve domain shift problem where machine learning algorithms are applied to sets of unseen images. Domain shift problem occurs when the training (source domain) and test (target domain) sets are collected in different environmental conditions but in related domains. In this way, the adaptation across data distributions of the source and target datasets are suggested as domain adaptation framework to overcome the performance degradation. In this paper, a novel domain adaptation method referred as joint local and statistical discriminant learning via feature alignment (LSA), is proposed to find a cross-domain subspace by matching cross-domain distribution shift and adapting the class structures of the local and statistical distributions across the source and target domains, during the dimensionality reduction. Specifically, LSA projects samples into an embedded subspace in which the distances across the samples from same class are minimized and the distances across samples from different classes are maximized, at each local and statistical area, during alignment of marginal and conditional distributions. Furthermore, the class densities of samples based on manifold structure in different classes are preserved to provide more separability across various classes. To evaluate the proposed method, comprehensive experiments have been conducted on benchmark cross-domain object and digit recognition datasets. Experimental results have verified the superiority of LSA with a large margin in average classification accuracy against several state-of-the-art distribution matching and discriminant learning methods of domain adaptation. Moreover, the results have demonstrated the effectiveness of our proposed representation learning. Our source code is available at https://github.com/jtahmores/LSA.

Keywords Visual domain adaptation · Discriminative feature learning · Statistical structure · Local structure · Distribution matching

1 Introduction

In recent years, researchers have paid great attention to visual domain adaptation (DA) problems [1]. Until now, domain adaptation methods have been used in visual problems such as face detection, document classification, sentiment analysis, handwritten digit recognition and event recognition. The main challenge of visual domain adaptation methods is to reduce the distance across the distributions of the training and test samples. In visual classification, the image sets are collected under different real-world conditions. For example, the source samples might be the images captured by digital camera, while the target set consists the images captured by a webcam. Different cameras capture images with different qualities and statistics (including different lightings, backgrounds, resolutions and poses), causing cross-domain distribution shift [2].

Overall, DA problems are studied in two tasks: unsupervised and semi-supervised. Unsupervised domain adaptation problems consist of the labeled source samples and unlabeled target samples. In semi-supervised DA problems, labeled source samples are available, while a limited number of the target samples are labeled, too. This paper has focused on unsupervised DA problems since no labeled data are available in target domain. The main idea of most unsupervised DA methods is to reduce the distribution shift via domain invariant feature learning [3–11]. Feature learning methods find a "good" feature representation for the source and target domains which matches the distributions of domains and preserves the variance of original data, simultaneously.

✉ Jafar Tahmoresnezhad
j.tahmores@it.uut.ac.ir

Elahe Gholenji
elahe.gholenjy@it.uut.ac.ir

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

Published online: 04 November 2019

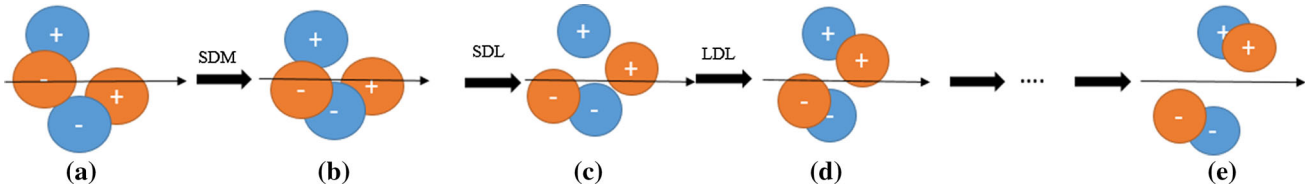

Fig. 1 The main idea of LSA. Red and blue circles illustrate source and target samples with two different classes, and black arrow illustrates the optimal classifier onto embedded subspace. **a** Source and target data in embedded subspace without distribution alignment. **b** Statistical distribution matching based on maximum mean discrepancy (MMD) measurements. **c** Statistical discriminative learning by minimizing the inter-class distances and maximizing the intra-class distances based on statistical structures. **d** Local discriminative learning by minimizing the distances across the pair samples with same manifold and label and maximizing the distances across the pair samples with same manifold but different labels. **e** Optimal embedded subspace with maximum statistical and local margin between various classes by performing an iterative procedure of distribution and discriminative learning

This paper proposes a novel feature learning method called joint local and statistical discriminant learning via feature alignment (LSA). LSA finds a common representation of the source and target data by exploiting the local and statistical structures of data during the subspace learning. The main contributions of LSA are as follows: (1) statistical distribution matching (SDM), (2) local discriminant learning (LDL) and (3) statistical discriminant learning (SDL) where the statistical distributions shift is minimized and the local and statistical discriminant structures of data are preserved, simultaneously.

In fact, SDM aligns the statistical distribution mismatch by minimizing the discrepancy across the marginal and conditional distributions of the source and target domains. LSA employs Maximum Mean Discrepancy (MMD) [12], as a nonparametric distribution measurement method, to compute the cross-domain marginal and conditional distributions distances, statistically. LDL analyzes the local and labeling information of samples to discriminate across various classes via manifold regularization. Therefore, the distances across pair samples with same manifold and label are minimized, while the distances across pair samples with same manifold but different labels are maximized along with class density preservation. To increase more separability across classes, LSA uses the statistical and discriminant information of data obtained by MMD. SDL minimizes the distances across the samples from their class mean while maximizes the distances mean of one class from means of other classes for both the source and target domains. The main idea of LSA is illustrated in Fig. 1.

We conduct comprehensive experiments on benchmark object and digit datasets, to evaluate and compare the performance of LSA. Our empirical results demonstrate that LSA outperforms other recent and related feature and deep-based domain adaptation methods by finding an adaptable and discriminant representation of the source and target data. The rest of the paper is organized as follows: Section 2 presents the previous works related to the proposed method. Section 3 describes the proposed method in detail. Section 4 introduces the experiments results and discussions. Finally, Sect. 5 draws the conclusion and future works.

Table 1 The notations of LSA method

Notation	Description
D_s/D_t	Source/target domain
X	Input data matrix
n_s/n_t	# Source/target samples
m, k	# Shared features /subspace bases
C, p	# Shared classes/nearest neighbors
D_s^c/D_t^c	Source/target samples in class c
n_s^c/n_t^c	# Source/target samples in class c
A, M, H	Projection/MMD/centering matrix
W_w/W_b	Within-/between-class weight matrix
L_w/L_b	Within-/between-class Laplacian matrix
S_w/S_b	Statistical within-/between-class matrix
λ, σ, θ	Regularization parameters

2 Related work

In this section, we discuss previous DA methods that are related to our proposed approach in detail. Recent DA methods often have focused their attention on dimensionality reduction to align the source and target domains. One of the base dimensionality reduction method is principal component analysis (PCA) [13] that projects the source and target domains into low-dimensional subspace via variance preservation of original data, without cross-domain distribution matching. However, cross-domain adaptation methods attempt to design a shared latent subspace via feature matching during the dimensionality reduction task, while in the obtained low-dimensional subspace, the similarity across the projected source and target domains is supposed to be increased.

The cross-domain distribution matching is reduced in terms of the marginal, the conditional or both. Transfer

component analysis (TCA) [3] is a basic DA method that improves PCA by marginal distribution matching along with variance preservation across domains. Joint distribution adaptation (JDA) [4] improves TCA by exploiting the conditional distribution matching without discrimination across various classes. Some feature-based DA methods [5–7] improve JDA by preserving the statistical discriminative structures of the domains in new representation. These methods were proposed to align dissimilarity across the source and target domains, statistically. In line with cross-domain geometrical alignment, a few recent DA methods propose a framework to improve the performance adaptation models via manifold regularization. The manifold-based methods reduce the shift across domains statistically and geometrically [9,10]. Overall, the previous methods only exploit the local or statistical structures of data to improve the separability and matching of domains. In this paper, we propose LSA which exploits joint local and statistical discriminant learning on both the source and the target domains during the marginal and conditional distributions matching.

3 Proposed method

In this section, the proposed method is presented for unsupervised DA problems, in detail. LSA finds a projection matrix $A \epsilon \mathbb{R}^{(m \times k)}$ where m and k are the original space size and subspace size, respectively, to map the source and target data into a low-dimensional representation by (1) statistical distribution matching, (2) local discriminant learning and (3) statistical discriminant learning. The used notations in LSA are summarized in Table 1. LSA is designed to solve the unsupervised DA problems. Since the labeled target samples are unavailable in unsupervised problems, we build nearest neighbor (NN) classifier on the source domain to predict the pseudo-labels of target samples. These labels are required to measure the conditional distribution and scatter distances of target samples. At first, the majority of pseudo-labels are incorrect, yet we employ the proposed method in an iterative procedure to update the pseudo-labels of target samples and improve the accuracy of NN-classifier in label prediction.

3.1 Statistical distribution matching (SDM)

LSA aligns joint marginal and conditional distributions across domains based on MMD measures. MMD measures the distances across the means of the source and target domains and distances across the source and target subdomains in all classes. The marginal and conditional distributions (namely as $marginal(D_s, D_t)$ and $conditional(D_s, D_t)$, respectively) are computed as follows:

$$
\begin{aligned}
marginal(D_s, D_t) &= \left\| \frac{1}{n_s} \sum_{i=1}^{n_s} Ax_i - \frac{1}{n_t} \sum_{j=n_s+1}^{n_s+n_t} Ax_j \right\|^2 \\
&= tr(A^T X M_0 X_T A), \\
conditional(D_s, D_t) &= \left\| \frac{1}{n_s^c} \sum_{x_i \epsilon X_s^c} Ax_i - \frac{1}{n_t^c} \sum_{x_i \epsilon X_t^c} Ax_j \right\|^2 \\
&= tr(A^T X M_c X^T A) \qquad (1)
\end{aligned}
$$

where

$$
M_{0ij} = \begin{cases} \frac{1}{n_s n_s} & (x_i, x_j) \epsilon D_s \\ \frac{1}{n_t n_t} & (x_i, x_j) \epsilon D_t \\ -\frac{1}{n_t n_t} & \text{otherwise} \end{cases},
$$

$$
M_{cij} = \begin{cases} \frac{1}{n_s^c n_s^c} & (x_i, x_j) \epsilon D_s^c \\ \frac{1}{n_t^c n_t^c} & (x_i, x_j) \epsilon D_t^c \\ -\frac{1}{n_s^c n_t^c} & \begin{cases} (x_i \epsilon D_t^c, x_j \epsilon D_s^c) \\ (x_i \epsilon D_s^c, x_j \epsilon D_t^c) \end{cases} \\ 0 & \text{otherwise} \end{cases} \qquad (2)
$$

LSA minimizes the marginal and conditional distributions shift by finding the projection matrix A via solving the following equation:

$$
\min_{AX^T HXA^T = I} \left(\sum_{c=0}^{C} AX^T M_c X A^T + \lambda \|A\|_F^2 \right) \qquad (3)
$$

where $\|.\|_F$ is the Frobenius norm. Via maximizing $AX^T HXA^T$, the variance of the original data is preserved in the embedded representation, where $H = I - (1/(n_s + n_t))\mathbb{1}\mathbb{1}^T$ is the centering matrix of the source and target samples ($I \epsilon \mathbb{R}^{(n_s+n_t)\times(n_s+n_t)}$ is identity matrix and $\mathbb{1}$ is the ones vector) . Second term is added to objection function to avoid trivial solutions. Therefore, SDM finds a feature subspace generation by increasing the similarity across the marginal and conditional distributions of the source and target samples.

3.2 Local discriminant learning (LDL)

SDM minimizes the statistical cross-domain shift, while domains are embedded in complex manifold structures consisting the vital intrinsic information of domains. To increase the margin across various classes, LSA improves the SDM by local discriminant learning. LDL enforces the embedded subspace for geometrical distribution and discriminant adaptation based on the following steps. To use the local and class manifold structures, at first a NN-graph G with $n_s + n_t$ nodes is structured on domains. LDL benefits from two following criteria on graph G:

1. LDL puts an edge between two samples i and j if x_i is one of p nearest neighbors of x_j and also, x_i and x_j have same labels.
2. LDL puts an edge between two samples i and j if x_i is one of p nearest neighbors of x_j but x_i and x_j have different labels.

Criteria 1 and 2 build local within-class graph G_w and between-class graph G_b. The weight matrix for computing weights between the connected samples of local graphs is defined as $W_{ij} = \sum_{i,j=1}^{n_s+n_t} = cos(x_i, x_j)$ where $cos(.,.)$ is the cosine function. Two within-class and between-class weight matrices W_w and W_b are computed on graph G_w and G_b, respectively. Based on manifold theorem [14], LDL aims to find a projection matrix A, while the distance between pair samples in G_w is minimized and the distance between pair samples in G_b is maximized, simultaneously. These goals are formulated as follows:

$$\min_A \sum_{i,j=1}^{n_s+n_t} (Ax_i - Ax_j) W_{w_{ij}}, \max_A \sum_{i,j=1}^{n_s+n_t} (AX_i - Ax_j) W_{b_{ij}} \tag{4}$$

Equation 4 can be rewritten in closed form as follows:

$$\begin{aligned} &\min_A (AX^T D_w X A^T - AX^T W_w X A^T), \\ &\max_A (AX^T D_b X A^T - AX^T W_b X A^T) \end{aligned} \tag{5}$$

where $D_{w_{ii}} = \sum_{j=1}^{n_s+n_t} W_{w_{ij}}$ and $D_{b_{ii}} = \sum_{j=1}^{n_s+n_t} W_{b_{ij}}$ are diagonal matrices. Also, D_{w_x} indicates the density of class containing x. To preserve densities of the classes during the subspace generation, we define $AX^T D_w X A^T = I$ as a constant [15]. Thus, Equation 5 is rewritten as follows:

$$\begin{aligned} &\min_A (I - AX^T W_w X A^T) = \max_A AX^T W_w X A^T, \\ &\max_A AX^T D L_b X A^T \end{aligned} \tag{6}$$

where $L_b = D_b - W_b$ is the Laplacian matrix computed on G_b. The objective function of LDL is defined as follows:

$$\max_{AX^T D_w X A^T = I} (AX^T (\sigma L_b + (1-\sigma) W_w) X A^T) \tag{7}$$

3.3 Statistical discriminant learning (SDL)

SDL computes the statistical within-class scatter matrix on both the source and the target domains as follows:

$$\begin{aligned} S_{w_{source}} &= \sum_{c=1}^{C} \left(\sum_{i=1}^{n_s^c} (x_i^c - \overline{m}_c)(x_i^c - \overline{m}_{cs}) \right), \\ S_{w_{target}} &= \sum_{c=1}^{C} \left(\sum_{i=1}^{n_t^c} (x_i^c - \overline{m}_c)(x_i^c - \overline{m}_{ct}) \right) \end{aligned} \tag{8}$$

where $\overline{m}_{cs} = 1/n_s \sum_{i=1}^{n_s} x_i$ and $\overline{m}_{ct} = 1/n_t \sum_{i=n_s+1}^{n_s+n_t} x_i$ are the sample means in class c for the source and target samples, respectively. The overall statistical within-class scatter matrix is $S_w = [S_{w_{source}}; S_{w_{target}}]$. Indeed, SDM minimizes the distance between samples from their class mean on mapped data to cluster the samples with same label in new representation.

To compute the statistical between-class scatter matrix S_b, we compute the distance between mean of one class (related to the source or target domains) from means of other classes in both the source and the target domains [11]. LSA maximizes the statistical between-class scatter matrix computed based on MMD measures as follows:

$$\begin{aligned} \max_A \Bigg(&\left\| \frac{1}{n_s^c} \sum_{x_i \epsilon X_s^c} Ax_i - \frac{1}{\sum_{r \epsilon \{1,\ldots,C\}-\{c\}} n_s^r} \sum_{x_j \epsilon X_s^r} Ax_j \right\|^2 \\ &+ \left\| \frac{1}{n_s^c} \sum_{x_i \epsilon X_s^c} Ax_i - \frac{1}{\sum_{r \epsilon \{1,\ldots,C\}-\{c\}} n_t^r} \sum_{x_j \epsilon X_t^r} Ax_j \right\|^2 \\ &+ \left\| \frac{1}{n_t^c} \sum_{x_i \epsilon X_t^c} Ax_i - \frac{1}{\sum_{r \epsilon \{1,\ldots,C\}-\{c\}} n_t^r} \sum_{x_j \epsilon X_t^r} Ax_j \right\|^2 \\ &+ \left\| \frac{1}{n_t^c} \sum_{x_i \epsilon X_t^c} Ax_i - \frac{1}{\sum_{r \epsilon \{1,\ldots,C\}-\{c\}} n_s^r} \sum_{x_j \epsilon X_s^r} Ax_j \right\|^2 \Bigg) \end{aligned} \tag{9}$$

where Equation 9 can be rewritten in closed form as follows:

$$\max_A \; AX^T (M_{ss} + M_{st} + M_{ts} + M_{tt}) X A^T \tag{10}$$

where M_{ss}, M_{st}, M_{ts} and M_{tt} are the between-class scatter matrices defined on the source to source samples, source to target samples, target to source samples and target to target samples, respectively, and are computed as follows:

$$(M_{ss})_{ij} = \begin{cases} \frac{1}{n_s^c n_s^c} & (x_i, x_j) \epsilon D_s^c \\ \frac{1}{n_s^r n_s^r} & (x_i, x_j) \epsilon D_s^r \\ -\frac{1}{n_s^c n_s^r} & \begin{cases} (x_i \epsilon D_s^c, x_j \epsilon D_s^r) \\ (x_i \epsilon D_s^r, x_j \epsilon D_s^c) \end{cases} \\ 0 & \text{otherwise} \end{cases},$$

$$(M_{st})_{ij} = \begin{cases} \frac{1}{n_s^c n_s^c} & (x_i, x_j) \epsilon D_s^c \\ \frac{1}{n_t^r n_t^r} & (x_i, x_j) \epsilon D_t^r \\ -\frac{1}{n_s^c n_t^r} & \begin{cases} (x_i \epsilon D_s^c, x_j \epsilon D_t^r) \\ (x_i \epsilon D_t^r, x_j \epsilon D_s^c) \end{cases} , \\ 0 & \text{otherwise} \end{cases}$$

$$(M_{tt})_{ij} = \begin{cases} \frac{1}{n_t^c n_t^c} & (x_i, x_j) \epsilon D_t^c \\ \frac{1}{n_t^r n_t^r} & (x_i, x_j) \epsilon D_t^r \\ -\frac{1}{n_t^c n_t^r} & \begin{cases} (x_i \epsilon D_t^c, x_j \epsilon D_t^r) \\ (x_i \epsilon D_t^r, x_j \epsilon D_t^c) \end{cases} , \\ 0 & \text{otherwise} \end{cases}$$

$$(M_{ts})_{ij} = \begin{cases} \frac{1}{n_t^c n_t^c} & (x_i, x_j) \epsilon D_t^c \\ \frac{1}{n_s^r n_s^r} & (x_i, x_j) \epsilon D_s^r \\ -\frac{1}{n_t^c n_s^r} & \begin{cases} (x_i \epsilon D_t^c, x_j \epsilon D_s^r) \\ (x_i \epsilon D_s^r, x_j \epsilon D_t^c) \end{cases} \\ 0 & \text{otherwise} \end{cases} \quad (11)$$

where D_s^r and D_t^r , respectively, are the source and target samples and also, n_s^r and n_t^r, respectively, are the number of the source and target samples in class $r \epsilon \{\{1, .., C\} - \{c\}\}$. Indeed, LSA increases the margin between various classes on the cross-domain, source and target subdomains, simultaneously.

Algorithm 1 presents the complete flow of LSA.

Algorithm 1. joint Local and Statistical discriminant learning via feature Alignment (LSA)

Input: Source data X_s, Target data X_t, Source label Y_s

Output: Target labels Y_t

1.Initialize pseudo target labels Y_t via 1-NN trained on $\{X_s, Y_s\}$

2.Compute centering matrix $H = I - (1/(n_s + n_t))\mathbb{1}\mathbb{1}^T$ that $I \epsilon \mathbb{R}^{(n_s+n_t)\times(n_s+n_t)}$ and $\mathbb{1}$ is the ones vector

Repeat until convergence

3.Compute MMD matrix $M = \sum_{c=0}^{C} M_c$ via Equation

4.Compute statistical discriminative matrices $S_w = [S_{w_{source}}; S_{w_{target}}]$ and $S_b = M_{ss} + M_{st} + M_{ts} + M_{tt}$ via equations 8 and 11

5. Compute between class Laplacian matrix $L_b = D_b - W_b$ on between class nearest neighbor graph G_b

6. Compute within class manifold matrices W_w and D_w on within class nearest neighbor graph G_w

7. Solve eigendecomposition for Equation 12 and select k smallest eigenvectors as adaptation matrix A

8.Update pseudo target labels Y_t using 1-NN classifier trained on projected source data $\{A^T X_s , Y_s \}$

end repeat

9.Return obtained pseudo target labels Y_t

3.4 Optimization problem

LSA finds a projection matrix A by combination of SDM, LDL and SDL based on Rayleigh quotient [4] as follows:

$$\min_{G=I} \left(AX^T \left(\sum_{c=1}^{C} M_c + \theta D_w \right) XA^T + AS_w A^T + \lambda \|A\|_F^2 \right) \quad (12)$$

where $G = AX^T(H + \sigma L_b + (1-\sigma)W_w + S_b)XA^T$ and $S_b = M_{ss} + M_{st} + M_{tt} + M_{ts}$. The projection matrix A is given by k minimum eigenvalues from solving Equation 12 via eigenvalue decomposition.

3.5 Time complexity

The computational complexity of SDM, SDL and LDL is $O(C(n_s + n_t)^2))$.

4 Experiments

In this section, we conduct our experiments on visual benchmarks to evaluate the performance of LSA.

4.1 Datasets

We evaluate LSA on four benchmark visual datasets: (1)Digit: USPS and MNIST domains contain handwritten digit images. We conduct two cross-domain problems on digit dataset: USPS-vs-MNIST (U-M) is designed by randomly selecting 1800 samples from USPS and 2000 samples from MNIST as the source and target domains, respectively, and MNIST-vs-USPS (M-U) is designed by reversing the source and target domains. (2)COIL20 contains images of grayscale objects with 20 classes that are divided into two domains: COIL1 (images taken in directions [0, 85] and [180, 265]) and COIL2 (images taken in directions [90, 175] and [270, 355]). Two cross-domain problems are conducted on COIL20: COIL1-vs-COIL2 (C1-C2) by selecting COIL1 as the source domain and COIL2 as the target domain and COIL2-vs-COIL1 (C2-C1) by selecting COIL2 as the source domain and COIL1 as the target domain. (3)Office+Caltech(Surf [2]): Office dataset contains three domains: Amazon (A) consists images collected from Amazon.com, Webcam (W) consists low-resolution images taken by webcam, and DSLR (D) consists high-resolution images taken by digital SLR. Caltech-256 dataset contains object images from 31 categories. We build 12 cross-domain problems on ten common classes from Office+Caltech datasets by selecting one domain as the source domains and selecting another domain as the target domain. (4)Office+Caltech(Decaf6 [16]): We follow deep-

Table 2 The optimal values of parameters for LSA (subspace size k and regularization parameters θ, σ and λ)

	k	λ	σ	θ
Office(surf)	100	0.5	0.3	0.1
Digit	220	0.0001	0.1	0.00005
COIL	20	0.0005	0.8	0.001
Office(decaf)	40	0.0005	0.1	0.1

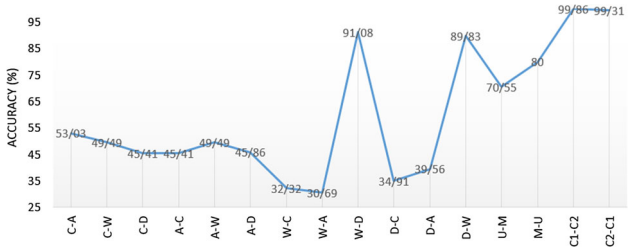

Fig. 2 Classification accuracy (%) of LSA on Office(surf), COIL and Digit datasets

based DA benchmark dataset, Office+Caltech dataset with decaf features represented in []. Deep convolutional activation features (Decaf) are the deep features computed by different pretrained convolutional neural network (CNN) models on ImageNet datasets [16]. This dataset contains four domains: Amazon (A), Webcam (W), DSLR (D) and Caltech (C). We build 12 cross-domain problems similar to Office+Caltech(Surf).

Overall, 28 cross-domain experiments are conducted on visual domains to evaluate LSA.

4.2 Experimental setup

LSA is compared with three types of distribution matching methods (TCA [4], JDA [5], BDA [8], MBDA [17] and TIT [18]), discriminant learning methods (SCA [19], VDA [5], CDDA [7], RSA-CDDA [20], TDCC [6], CLGA [10], DGA-DA [11], DICE [21], JGSA [9] and DICD [22]) and deep-based DA methods (AlexNet [23], TAISL [24], PUnda [25], ELM [26], AELM [26], DDC [27] and DAN [28]). The mentioned methods are NN-based, and we compare LSA with best reported results of the mentioned methods.

LSA has six hyper-parameters: subspace size k, regularization parameters θ, σ and λ, the nearest neighbor number p and the iteration number T. In our experiments, we set optimal parameters on visual datasets based on Table 2. Also, the optimal values of T and p are set to 10 in all experiments. The evaluation metric to compute the classification accuracy on target samples is as $Accuracy = \frac{|\{x:\, x\epsilon X_t \cap f(x)=y(x)|}{n_t}$, where $y(x)$ and $f(x)$ are the actual and predicted labels, respectively, for sample x from target domain D_T.

4.3 Experimental results and discussions

The results of LSA and the mentioned methods on Office+Caltech (surf), COIL and Digit are shown in Table 3, and the highest accuracy for each cross-domain problem is highlighted in bold. Also, the accuracy of our proposed methods for Office+Caltech (surf), COIL and Digit datasets is visualized in Fig. 2. According to the results, LSA has 1.01% improvement in average accuracy against JGSA (best compared method). LSA gains 99.86% accuracy on C1-C2 and 99.31% on C2-C1. Indeed, LSA predicts incorrect labels only for 1 of 720 samples on C1-C2 and also, for 5 out 720 samples on C2-C1. As is clear, LSA builds an adaptable representation compared to recent feature matching methods, via jointly matching and discriminative learning.

Also, we evaluate the performance of LSA against deep learning-based DA methods. The results of LSA and deep learning-based DA methods on decaf dataset are shown in Table 4. The compared methods are categorized into two categories: feature-based DA methods and deep-based domain adaptation methods. According to obtained results, LSA has 0.49% improvement in average accuracy compared to best feature-based methods (DGA-DA). Also, LSA outperforms some feature-based methods on Office+Caltech with deep features, while has weakly performance on Office+Caltech with surf features against them.

Comparison with deep learning-based methods, LSA has 1.15% improvement in average accuracy compared to best deep-based methods, i.e., DAN, and outperforms it on 9 out of 12 problems, which only employs feature matching strategies.

4.4 Parameter sensitivity

Using Office+Caltech, Digit and COIL datasets, we evaluate the sensitivity of LSA with respect to hyper-parameters. We select optimal values of parameters based on similar trends on most datasets. We evaluate the performance of LSA with respect to four parameters: subspace size k and regularization parameters θ, σ and λ. We perform the experiments with $k\epsilon[20, 220]$, $(\theta, \lambda)\epsilon[0.00001, 10]$ and $\sigma\epsilon[0, 1]$. Overall, the value of subspace size k shows the accuracy of the proposed method to reconstruct the original data in low-dimensional subspace, the value of regularization parameter λ controls the complexity of model, the value of regularization parameter θ preserves the density of classes in new subspace, and the value of regularization parameter σ is set to exploit the local and class manifold of domains. Figure 3 shows the behavior of three visual problems with respect to different values of LSA parameters.

As previously mentioned, we perform an iterative procedure on LSA to improve the accuracy of pseudo-target labels. The accuracy of visual problems with respect to different iter-

Table 3 Classification accuracy (%) of LSA against compared methods on Office+Caltech (surf), COIL and Digit datasets

Dataset	NN	TCA	JDA	SCA	VDA	RSA	CDDA	CLGA	JGSA	TDCC	BDA	MBDA	DGA-DA	**LSA**
C-A	23.7	45.82	44.78	43.74	46.14	45.5	48.33	48.02	**53.15**	47.18	44.89	47.29	52.09	53.03
C-W	25.76	30.51	41.69	33.56	46.1	41.69	44.75	42.37	48.47	48.47	38.64	41.02	47.12	**49.49**
C-D	25.48	35.67	45.22	39.49	**51.59**	49.04	48.41	49.04	48.41	47.13	47.77	49.02	45.86	49.68
A-C	26	40.07	39.36	38.29	42.21	39.09	42.12	42.3	41.5	42.03	40.78	40.78	41.32	**45.41**
A-W	29.83	35.25	37.97	33.9	**51.19**	43.39	41.69	41.36	45.08	43.05	39.32	40	38.31	49.49
A-D	25.48	34.39	39.49	34.21	**48.41**	39.49	37.58	36.31	45.22	38.85	43.31	43.31	38.22	45.86
W-C	19.86	29.92	31.17	30.63	27.6	32.95	31.97	32.95	**33.57**	32.06	28.94	31.17	33.3	32.32
W-A	22.96	28.81	32.78	30.48	26.1	35.28	37.27	34.57	40.81	33.19	32.99	40.08	**41.75**	30.69
W-D	59.24	85.99	89.17	92.36	89.18	**94.9**	87.9	92.36	88.54	88.54	91.72	92.36	89.81	91.08
D-C	26.27	32.06	31.52	32.32	31.26	33.66	34.64	33.66	30.28	33.13	32.5	33.21	33.66	**34.91**
D-A	28.5	31.42	33.09	33.72	37.68	36.01	33.61	33.51	35.99	38.73	33.51	33.09	33.51	**39.56**
D-W	63.39	86.44	89.49	88.81	90.85	90.17	90.51	89.83	**93.22**	**93.22**	90.85	91.86	**93.22**	89.83
U-M	44.7	51.05	59.65	48	62.95	63.2	62.05	58.55	68.15	60.45	59.35	61.4	**70.75**	70.55
M-U	65.94	56.28	67.28	65.11	74.72	77.5	76.22	80.44	75.72	67.89	69.78	74.44	**82.33**	80
C1-C2	83.61	88.47	89.17		99.31	95.42	91.53	93.89	95.4	95	97.22	97.78	**100**	99.86
C2-C1	82.78	85.83	88.47		97.92	95.28	93.89	92.5	93.9	92.64	96.81	97.08	**100**	99.31
Average(Office)	31.37	43.03	46.31	44.29	49.03	48.41	48.22	48.23	50.58	48.16	47.15	48.75	49.02	**50.95**
Average(Digit)	55.32	53.67	63.47	56.56	68.84	70.35	69.14	67.14	74.3	64.17	64.57	67.92	**76.54**	75.28
Average(COIL)	83.2	87.15	88.89		98.62	95.35	92.71	93.19	94.7	93.82	97.02	97.43	**100**	99.58
Average(Overall)	40.84	49.87	53.77		57.7	57.02	56.4	56.21	59.06	55.87	55.56	57.23	58.83	**60.07**

Bold indicates the max values

Table 4 Classification accuracy (%) of LSA against compared methods on cross-domain Office+Caltech (decaf) dataset

	NN	JDA	SCA	JGSA	CDDA	DGA-DA	TIT	DICD	ELM	AELM	AlexNet	DAN	DDC	TAISL	PUnDA	**LSA**
C-A	87.06	89.7	89.46	91.44	90.71	91.25	89.5	91.02	89.07	89.46	91.9	92	91.9	90	90.3	**90.07**
C-W	72.2	83.7	85.42	86.78	85.76	**93.56**	92.1	92.2	70.51	79.32	83.7	90.6	85.4	85.3	88.3	**93.56**
C-D	80.89	86.6	87.9	93.63	**91.72**	**91.72**	86.7	93.63	78.98	81.53	87.1	89.3	88.8	906	76.2	90.45
A-C	78.54	82.2	78.81	84.86	85.66	85.2	83.8	86.02	79.61	79.96	83	84.1	85	80.1	82.3	**87.27**
A-W	77.31	78.6	75.93	81.02	78.31	80.98	91.4	81.36	74.56	77.63	79.5	**91.8**	86.1	77.9	82.7	83.39
A-D	80.25	80.2	85.35	88.54	84.08	89.81	89.1	83.44	80.25	85.35	87.4	**91.7**	89	85.1	76.2	89.81
W-C	68.21	80.5	74.18	84.95	86.02	86.46	80.2	83.97	70.61	71.24	73	81.2	78	82.6	82.6	**86.82**
W-A	73.07	88.1	86.12	90.71	89.77	90.81	89.3	89.67	75.37	76.83	83.8	**92.1**	84.9	85.6	86.9	91.72
W-D	**100**	**100**	**100**	**100**	**100**	**100**	94.9	**100**	**100**	**100**	**100**	**100**	**100**	97.7	89.8	**100**
D-C	70.08	80.1	78.09	86.2	86.2	86.2	80.7	86.11	68.21	75.6	79	80.3	81.1	84	69.2	**87.89**
D-A	75.89	89.4	89.98	91.96	91.34	**93.11**	92.5	92.17	80.79	83.19	87.1	90	89.5	87.6	83.1	92.69
D-W	97.97	98.9	98.64	99.66	**100**	**100**	88.1	98.98	98.31	98.98	97.7	98.5	98.2	95	93.4	99.32
Average	80.12	86.5	85.88	89.98	89.13	90.76	88.2	89.88	80.52	83.25	86.1	90.1	88.2	86.87	83.42	**91.25**

Bold indicates the max values

ation numbers is shown in Fig. 3e. As can be seen in figure, LSA convergences in 10 iterations on most datasets.

5 Conclusion and future works

In this paper, we proposed joint local and statistical discriminant learning via feature alignment (LSA) method which exploits three structures of both the source and the target domains: (1) statistical distribution structures, (2) statistical discriminant structures and (3) local discriminant structures, to find the aligned representation of data. Comprehensive experiments conducted on benchmark visual datasets verify the effectiveness and improvement of LSA compared to recent deep- and feature-based methods. We plan to extend our proposed method on deep neural networks to be able to work in other vision problems such as object detection. Also,

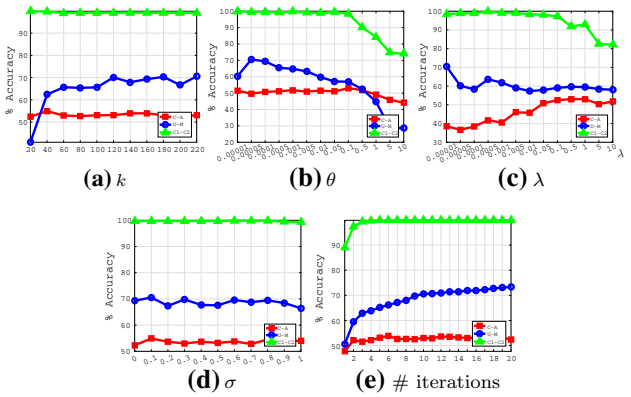

Fig. 3 Evaluation of the classification accuracy (%) of LSA with respect to different values of parameters on three samples of visual experiments

LSA can be extended to online domain adaptation where the online and real-time datasets are used.

References

1. Jhuo, I.H., Liu, D., Lee, D.T., Chang, S.F.: Robust visual domain adaptation with low-rank reconstruction. In: 2012 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), June, pp. 2168–2175. IEEE
2. Saenko, K., Kulis, B., Fritz, M., Darrell, T.: Adapting visual category models to new domains. In: European conference on computer vision, September, pp. 213–226. Springer, Berlin, Heidelberg (2010)
3. Pan, S.J., Tsang, I.W., Kwok, J.T., Yang, Q.: Domain adaptation via transfer component analysis. IEEE Trans. Neural Netw. **22**(2), 199–210 (2011)
4. Long, M., Wang, J., Ding, G., Sun, J., Yu, P.S.: Transfer feature learning with joint distribution adaptation. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 2200–2207 (2013)
5. Tahmoresnezhad, J., Hashemi, S.: Visual domain adaptation via transfer feature learning. Knowl. Inf. Syst. **50**(2), 585–605 (2017)
6. Fan, Y., Yan, G., Li, S., Song, S., Wang, W., Peng, X.: Transfer domain class clustering for unsupervised domain adaptation. In: International Conference on Electrical and Information Technologies for Rail Transportation, October, pp. 827–835. Springer, Singapore (2017)
7. Luo, L., Wang, X., Hu, S., Wang, C., Tang, Y., Chen, L.: Close yet distinctive domain adaptation (2017). arXiv preprint arXiv:1704.04235
8. Wang, J., Chen, Y., Hao, S., Feng, W., Shen, Z.: Balanced distribution adaptation for transfer learning. In: 2017 IEEE International Conference on Data Mining (ICDM), November, pp. 1129–1134. IEEE (2017)
9. Zhang, J., Li, W., Ogunbona, P.: Joint geometrical and statistical alignment for visual domain adaptation (2017). arXiv preprint arXiv:1705.05498
10. Liu, J., Li, J., Lu, K.: Coupled local-global adaptation for multi-source transfer learning. Neurocomputing **275**, 247–254 (2018)
11. Luo, L., Chen, L., Hu, S., Lu, Y., Wang, X.: Discriminative and geometry aware unsupervised domain adaptation (2017). arXiv preprint arXiv:1712.10042
12. Gretton, A., Borgwardt, K., Rasch, M.J., Scholkopf, B., Smola, A.J.: A kernel method for the two-sample problem (2008). arXiv:0805.2368
13. Abdi, H., Williams, L.J.: Principal component analysis. Wiley Interdiscip. Rev. Comput. Stat. **2**(4), 433–459 (2010)
14. Belkin, M., Niyogi, P., Sindhwani, V.: Manifold regularization: a geometric framework for learning from labeled and unlabeled examples. J. Mach. Learn. Res. **7**(Nov), 2399–434 (2006)
15. Cai, D., He, X., Zhou, K., Han, J., Bao, H.: Locality sensitive discriminant analysis. In: IJCAI, January, pp. 1713–1726 (2007)
16. Donahue, J., Jia, Y., Vinyals, O., Hoffman, J., Zhang, N., Tzeng, E., Darrell, T.: Decaf: a deep convolutional activation feature for generic visual recognition. In: International conference on machine learning, January, pp. 647–655 (2014)
17. Li, Y., Cheng, L., Peng, Y., Wen, Z., Ying, S.: Manifold alignment and distribution adaptation for unsupervised domain adaptation. In: 2019 IEEE International Conference on Multimedia and Expo (ICME), July, pp. 688–693. IEEE (2019)
18. Li, J., Lu, K., Huang, Z., Zhu, L., Shen, H.T.: Transfer independently together: a generalized framework for domain adaptation. IEEE Trans. Cybern. **49**(6), 2144–2155 (2018)
19. Ghifary, M., Balduzzi, D., Kleijn, W.B., Zhang, M.: Scatter component analysis: a unified framework for domain adaptation and domain generalization. IEEE Trans. Pattern Anal. Mach. Intell. **39**(7), 1414–30 (2017)
20. Luo, L., Wang, X., Hu, S., Chen, L.: Robust data geometric structure aligned close yet discriminative domain adaptation (2017). arXiv:1705.08620
21. Liang, J., He, R., Sun, Z., Tan, T.: Aggregating randomized clustering-promoting invariant projections for domain adaptation. IEEE Trans. Pattern Anal. Mach. Intell. **41**(5), 1027–1042 (2018)
22. 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)
23. Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. In: Advances in neural information processing systems, pp. 1097–1105 (2012)
24. Lu, H., Zhang, L., Cao, Z., Wei, W., Xian, K., Shen, C., van den Hengel, A.: When unsupervised domain adaptation meets tensor representations. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 599–608 (2017)
25. Gholami, B., Pavlovic, V.: Punda: Probabilistic unsupervised domain adaptation for knowledge transfer across visual categories. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 3581–3590 (2017)
26. Uzair, M., Mian, A.: Blind domain adaptation with augmented extreme learning machine features. IEEE Trans. Cybern. **47**(3), 651–660 (2016)
27. Tzeng, E., Hoffman, J., Zhang, N., Saenko, K., Darrell, T.: Deep domain confusion: maximizing for domain invariance (2014). arXiv preprint arXiv:1412.3474
28. Long, M., Cao, Y., Wang, J., Jordan, M.I.: Learning transferable features with deep adaptation networks (2015). arXiv preprint arXiv:1502.02791