Knowledge and Information Systems (2021) 63:2157–2188
https://doi.org/10.1007/s10115-021-01586-0

REGULAR PAPER

Cross- and multiple-domains visual transfer learning via iterative Fischer linear discriminant analysis

Mehri Mardani[1] · **Jafar Tahmoresnezhad**[1]

Received: 5 September 2017 / Revised: 7 June 2021 / Accepted: 12 June 2021
/ Published online: 3 July 2021

Abstract
The standard machine learning tasks often assume that the training (source domain) and test (target domain) data follow the same distribution and feature space. However, many real-world applications suffer from the limited number of training labeled data and benefit from the related available labeled datasets to train the model. In this way, since there is the distribution difference across the source and target domains (i.e., domain shift problem), the learned classifier on the training set might perform poorly on the test set. To address the shift problem, domain adaptation provides variety of solutions to learn robust classifiers to deal with distribution mismatch across the source and target domains. In this paper, we put forward a novel domain adaptation approach, referred to as cross- and multiple-domains visual transfer learning via iterative Fischer linear discriminant analysis (CIDA) to tackle shift problem across domains. CIDA transfers the source and target domains into a shared low-dimensional Fischer linear discriminant analysis (FLDA)-based subspace in an unsupervised manner. CIDA benefits joint FLDA and domain adaptation criterions to reduce the distribution mismatch across the training and test sets. Moreover, CIDA employs an adaptive classifier to build a robust model against data drift across different domains. Also, CIDA generates the intermediate pseudotarget labels to utilize the target data in training process. In this way, CIDA refines the pseudolabels using an iterative manner to converge the model. Our extensive experiments illustrate that CIDA significantly outperforms the baseline machine learning and other state-of-the-art transfer learning methods on nine visual benchmark datasets under different difficulties.

Keywords Machine learning · Transfer learning · Domain shift · Fischer linear discriminant analysis · Feature- and model-based domain adaptation

1 Introduction

The machine learning and pattern recognition tasks often assume that the training and test data come from the similar distributions and feature spaces [1]. However, this assumption

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

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

is unrealistic for many real-world applications where we have to benefit from other existing and related domains due to the lack of labeled training data. In this situation, because of the distribution mismatch across the training and test data, the trained model might perform poorly on the test data [2]. For example, in sentiment classification task, the reviews on the books have significant distribution difference against the reviews on electronic devices [3]. However, when the label is not available for the test data, we have to adapt the learning data from other related domains. The distribution difference across the training and test sets is known as domain shift problem.

To address domain shift issue, *domain adaptation (DA)* [4] and *transfer learning (TL)* [5] have led to major solutions in recent years. In DA, the knowledge from an already trained machine learning model is transferred to a different but related problem. In fact, DA tries to improve the generalization in one task via employing what has been learned in another task. DA learns a robust classifier to deal with the distribution mismatch across the source and target domains. DA approaches according to the available information in the target domain are divided into two general categories as follows: unsupervised DA where there are no labeled data in the target domain [3,6–10], and semi-supervised DA where the target domain contains a small amount of labeled data [11–13]. However, both the unsupervised and the semi-supervised DA can benefit from either single source domain [14–17] or multi-source domains [18–21] to transfer knowledge across domains.

Since the source and target domains have different distributions, the key of having a prosperous adaptation is the reduction of distribution divergence. To this end, the existing DA approaches are summarized into the following three different categories: (1) instance-based transfer learning approaches on which the source domain samples are reweighted to have similar distribution with target samples [22–25], (2) feature-based transfer learning approaches that project the source and target data into a common subspace with shared features [8,17,26–31] and (3) model-based transfer learning approaches in which an adaptive classifier is modeled using joint parameters and priors of learned model [32–35]. In this paper, our focus is on feature-based and model-based transfer learning approaches. However, there are two important challenges in existing works, i.e., *defective transformation* and *unevaluated discriminant analysis*.

Defective transformation means that both feature learning and model learning approaches can only reduce, but not remove the distribution mismatch [36,37]. Particularly, the feature learning [17,28–31] conducts feature transformation to have better feature representation. However, the feature mismatch is not removed after transformation [38] since the feature transformation only employs the manifold and structure of data, but ignores to perform strengthen the model against cross domain changes. Also, the model learning usually adapts the priors and parameters of model in the original feature space, where the features are often mismatched, which makes it difficult to minimize the discrepancy across domains. Therefore, it is essential to benefit both the feature learning and the model learning to further facilitate DA.

Unevaluated discriminant analysis means that the FLDA-based existing works [39,40] only attempt to project the training and test samples into a low-dimensional subspace based on the maximum class discrimination. But, they failed to evaluate the distribution difference across domains during the discriminant analysis. However, the iterative FLDA exploits the pseudotarget labels to customize the FLDA criterions to adapt the multiple source domains with a target domain.

As far as we know, there has been no previous work that tackle these two challenges together. In this work, we propose a novel *cross- and multiple-domains visual transfer learning via iterative Fischer linear discriminant analysis (CIDA)* approach, to address both

challenges of defective transformation and unevaluated discriminant analysis. CIDA learns a domain-invariant classifier in an iterative FLDA-based embedding with empirical risk minimization, while performing hybrid distribution alignment by considering the different importance of criterions on embedded subspace. This work makes the following contributions:

(1) CIDA addresses the challenges of both defective transformation and unevaluated discriminant analysis. CIDA strengthens the model against cross domain changes and minimize the cross domain discrepancies. CIDA benefits both the feature learning and the model learning to tackle challenges ahead.
(2) CIDA focuses on multi-source DA where it exploits multiple knowledge resources to transfer across domains. The experiment results indicate that the existence of multiple related resources facilitate the adaptation tasks.
(3) CIDA employs an iterative FLDA method to estimate the pseudotarget labels for better transformation of data in a hybrid manner. CIDA evaluates the distribution difference across domains during the discriminant analysis. CIDA employs the iterative FLDA and pseudotarget labels to customize the FLDA criterions to adapt the multiple source domains with target domain.
(4) CIDA is evaluated on nine benchmark domain adaptation datasets. The experiments are conducted to assess the robustness and strengthens of CIDA to face with the various situations. However, the results illustrate that CIDA outperforms the baseline machine learning and other state-of-the-art transfer learning approaches.

The rest of paper is organized as follows. In the next section, a short review on DA literature is presented. The proposed method is introduced in Sect. 3. The experimental setup and implementation details are explained in Sect. 4. Section 5 includes the experimental results and discussions. The last section contains the conclusion and future works.

2 Related work

In this section, two lines of related work are discussed as follows: (1) the dimensionality reduction-based transfer learning and (2) multi-source transfer learning to highlight the difference between the proposed algorithm and the available works.

2.1 Dimensionality reduction-based transfer learning

Dimensionality reduction approaches are a well-known case to represent the learning techniques. In general, most of the dimensionality reduction approaches obey from two main frameworks: (1) PCA-based (principal component analysis) framework [29,41,42], which attempts to project data into a low-dimensional space besides the maximum variance preservation on the embedded subspace and (2) FLDA-based framework [43–45], which attempts to project data into a low-dimensional space besides the maximum class discrimination. However, both the PCA- and the FLDA-based frameworks show poor performance in case of domain shift problem where the source and target domains obey from different distributions.

There are several PCA-based approaches such as transfer component analysis (TCA) [29], joint distribution adaptation (JDA) [41] and visual domain adaptation (VDA) [42], which exploit PCA to embed data into a latent subspace. TCA is an efficient feature extraction method that finds the transferred components of input data based on the variance maximization and mismatch minimization. TCA benefits from maximum mean discrepancy (MMD)

[46] to measure the distribution difference of source and target domains. TCA is one of the benchmark approaches in DA literature.

JDA is another novel transfer learning approach that aims to learn a common feature subspace that jointly decreases the marginal and conditional distribution differences between the source and target domains. JDA utilizes MMD to measure the distance among the source and target domains. VDA is a novel framework that constructs a shared feature representation besides the minimizing of joint marginal and conditional distributions across the source and target domains. In fact, VDA preserves the statistical and geometrical structure of input data using the manifold assumptions. In addition, VDA exploits the domain invariant clustering in an embedded subspace to discriminate the various classes of target data.

The main drawback of PCA-based approaches is that most of them embed data in a low-dimensional subspace without considering the class discrimination criteria. In contrast, the FLDA-based approaches consider the class discrimination criteria besides the domain adaptation criterions to adapt the distribution mismatch across domains. Wenting et al. [44] proposed an effective framework that finds a common feature representation such that it maximizes the difference between classes (class-separate objective) and minimizes the difference between domains (domain-merge objective).

Cuong et al. [43] introduced a generalized Fischer-based method for domain shift problem (FIDOS) that constructs a shared feature representation besides the minimizing within-class scatter and maximizing the class discrimination. Zheng et al. [45] proposed the transferred dimensionality reduction (TDA), which is an iterative method that iteratively utilizes the clustering procedure to predict the labels of unlabeled target data. TDA employs the dimensionality reduction and distribution discrepancy minimization across the source and target domains.

2.2 Multi-source transfer learning

In recent years, multi-source transfer learning is of interest to researchers, since there are generally multiple sources available for knowledge transfer in target learning [47]. Although tapping from multi-sources would provide more knowledge, they further result to a challenging domain adaptation issue, since the multiple sources have a large mismatch from each other. To this end, there are dozens of the proposed methods to deal with multi-source problems [48–52].

Transfer learning for multiple-domain sentiment analysis [48] is a Bayesian probabilistic model to handle the multiple source and multiple target domains. The method uses Gibbs sampling for inferring the parameters of model from unlabeled and labeled data. Multi-domain collaborative filtering (MCF) [52] is a probabilistic method, which exploits the probabilistic matrix factorization for modeling of rating problem in various domains. MCF transfers the knowledge across different domains by automatically learning the correlation between domains.

Conditional probability-based multi-source domain adaptation (CP-MDA) [51] is a multi-source domain adaptation method for realizing the different stages of fatigue using the surface electromyography signals, which tackles the distribution differences. CP-MDA employs a novel weighting scheme to address the conditional probability distribution differences across multiple domains. Boosting for transfer learning with multiple sources [47] extends the boosting framework for transferring the knowledge from multiple sources. The proposed approach addresses the negative transfer problem to import the knowledge from multiple sources. Multi-domain adaptation for sentiment classification (MCS) [50] adapts the classi-

fiers for a specific domain via multiple source domains. MCS combines the base classifiers to select the automatically labeled instances from unlabeled data in target data.

Different from the existing models, our CIDA tends to extract an embedded shared subspace in which the within-class scatters and between-class scatters regularizers are developed to couple multiple sources during the knowledge transfer. Compared with [43], our model uses from general criterions to extract the high-ranked subspaces. Furthermore, we propose the feature and model learning regularizers to further strengthen the supervised knowledge from multiple sources and intrinsic information of target.

3 Proposed method

In this section, we introduce our CIDA approach in detail for addressing the unsupervised domain shift problem, efficiently.

3.1 Motivation

In this work, we propose a new FLDA-based framework that projects the input data into an embedded subspace based on the following criteria: (1) the distribution of source and target data obey from similar distribution, (2) a solution based on the representation and model learning, (3) an intermediate pseudolabel prediction to converge the accurate results. In the rest of this section, the preliminaries and problem description are presented with full details.

3.2 Problem description

Definition 1 *(Domain)* A domain $\mathcal{D}$ is comprised of $\{\mathcal{X}, P(X)\}$ where $\mathcal{X}$ is an m-dimensional feature space and $P(X)$ is a marginal probability distribution on $\mathcal{X}$ where $X = \{x_1, \ldots, x_n\} \in \mathcal{X}$. The input data includes two domains, the source domain S and the target domain T. We denote the source domain as $\mathcal{D}_s = \{(x_1, y_1), \ldots, (x_{n_s}, y_{n_s})\}$ where is completely labeled. Similarly, we define the target domain as $\mathcal{D}_t = \{x_{n_s+1}, \ldots, x_{n_s+n_t}\}$ where is fully unlabeled. Also, n_s and n_t are defined as the number of source and target samples, respectively.

Definition 2 *(Task)* Given a specific domain $\mathcal{D}$, a task for domain $\mathcal{D}$ is denoted by $\mathcal{T} = \{\mathcal{Y}, f(x)\}$ where is composed of the following two components: $\mathcal{Y}$ is the set of labels of domain $\mathcal{D}$ and $f(x)$ is a classifier, which can be employed to predict the corresponding labels of data x. From a probabilistic standpoint, $f(x)$ can be expressed as the conditional probability distribution, i.e., $f(x) = Q(y \mid x)$ where $\mathrm{y} \in \mathcal{Y}$.

The domain shift problem is considered with N_s source domains and a single target domain. Therefore, the input is a collection of related source domains as $X^S = \{X^1, X^2, \ldots, X^{N_s}\}$ and the output is a linear mapping, which transforms data into an embedded subspace to predict the labels of target data X^T. Since the distribution difference across the source and target domains degrades the performance of model, in this paper, we are to learn a feature representation in which the marginal distribution difference of source and target domains is reduced, i.e., $P_s(x_s) \approx P_t(x_t)$ where $P_s(x_s)$ and $P_t(x_t)$ are the marginal distribution probability of source and target domains, respectively. Moreover, $\mathcal{X}_s = \mathcal{X}_t$ where $\mathcal{X}_s$ and $\mathcal{X}_t$ are the feature spaces of source and target domains, in turn. In fact, CIDA attempts to learn a shared low-dimensional feature space on which the marginal distribution of source and target domains obeys from the similar distribution.

3.3 Generating domain invariant representation

In this section, at first we introduce the classical FLDA and then propose our CIDA, which is the based on FLDA.

3.3.1 Feature extraction using classical FLDA

The main objective of FLDA is to model one dependent variable as a linear combination of other variables. In this way, FLDA extracts new features of a domain according to the linear combination of the available features. In fact, FLDA attempts to maximize the class-separate degree to incorporate the following two criterions: (1) FLDA maximizes the between-class scatter matrix (S_b) and (2) minimizes the within-class scatter matrix (S_w), such that the samples in the embedded subspace have maximum discrimination.

S_b and S_w are defined as follows such that K and n_i demonstrate the number of available classes and the number of samples that belongs to class i, respectively:

$$S_b = \sum_{i=1}^{K} p_i(\mu_i - \mu)(\mu_i - \mu)^T \tag{1}$$

$$S_w = \sum_{i=1}^{K} \sum_{j=1}^{n_i} (x_i^j - \mu_i)(x_i^j - \mu_i)^T \tag{2}$$

where $p_i = \frac{n_i}{N}$ shows the prior of class i, N is the total number of samples, $\mu_i = \frac{1}{n_i} \sum_{j=1}^{n_i} x_i^j$ is the mean vector of class i, $\mu = \frac{1}{N} \sum_{i=1}^{N} x^i$ is the overall data mean, x_i^j is the j^{th} sample in i^{th} class. Therefore, the projection matrix of FLDA, i.e., the matrix A, is obtained from maximizing the following optimization problem $J(A)$:

$$J(A) = \frac{A^T S_b A}{A^T S_w A}. \tag{3}$$

The intuition behind maximizing $J(A)$ is to learn a projection matrix $A \in R^{m \times k}$ in order to transform data from the original feature space that composed of m features into a low-dimensional subspace with k features (i.e., $k < m$). The optimization problem $J(A)$ can be solved using the eigenvalue decomposition of ${S_w}^{-1} S_b$ where k eigenvectors of ${S_w}^{-1} S_b$ corresponding to k largest eigenvalues is chosen as matrix A.

3.3.2 CIDA

In recent years, the classical machine learning approaches could not be responsible to most of real-world applications, where the attention to DA has been increased due to the considerable performance of it to deal with the available problems. Thus, we are to tackle the shift problem by integrating the machine learning approaches and DA solutions.

In this paper, the domain shift problem is leveraged based on the multi-source scenario. Thus, the training and test sets are defined as $X^S = \{X^1, X^2, \ldots, X^{N_s}\}$ and X^T, respectively, where X^u denotes the u^{th} source domain and N_s is the total number of source domains. In general, the domain adaptation problems are divided into following categories, heterogeneous and homogeneous. In heterogeneous domain adaptation, the source and target domains are from different feature spaces, while in homogeneous domain adaptation, the source and target

domains are from the same one. Our problem belongs to the homogeneous domain adaptation problem.

Since the various classes might come from different distributions, they are treated differently, and thus, there is dissimilarity among them. Therefore, we enlarge the margins across various classes as much as possible. To this end, the new between-class scatter matrix, S'_B, is defined as follows:

$$S'_B = \frac{1}{N_s^2} \sum_{i,j} \sum_{u,v} p_i^u p_j^v (\mu_i^u - \mu_j^v)(\mu_i^u - \mu_j^v)^T \tag{4}$$

where p_i^u and μ_i^u are the prior and the mean of class i on the subset X^u, respectively. Also p_j^v and μ_j^v are the prior and the mean of class j on the subset X^v, in turn. Moreover, S'_B computes the weighted average of between class-scatter matrices across different subsets from various classes in source domains. In the other words, S'_B minimizes the marginal distribution difference of various classes of source domains such that the learned classifier can accurately predict the labels of target data due to the large margins across various classes of different source domains.

Moreover, we are to minimize the distribution difference across the same classes in different domains to adapt the source and target domains. In this way, the difference among the same classes from source and target domains is minimized. Hence, we shrink the margins among the samples of the same classes of source and target domains in order to well-align the samples. Consequently, S'_W is defined as the new within class scatter matrix as follows:

$$S'_W = \sum_{u=1}^{N_s} \sum_{i=1}^{K} (\mu_i^u - \mu_i^t)(\mu_i^u - \mu_i^t)^T \tag{5}$$

where μ_i^t is the mean of class i that belongs to X^T. In fact, S'_W minimizes the marginal distribution difference between the same classes that belong to the source and target domains.

The intuition behind CIDA is to learn a projection matrix $A \in R^{m \times k}$ that persuades the following three principal objectives: (1) the marginal distribution difference of various classes of source and target domains is maximized (i.e., S'_B), (2) the marginal distribution difference between the same class of source and target domains is minimized (i.e., S'_W) and (3) the amount of variance between the various classes is minimized (i.e., S_W). Therefore, the optimization problem of CIDA, i.e. $J'(A)$, is composed of S'_B, S'_W and S_W as follows:

$$J'(A) = \frac{A S'_B A^T}{A(cS_W + (1-c)S'_W)A^T} \tag{6}$$

where $c \in [0, 1]$ is a parameter to regulate between S_W and S'_W. Similar to FLDA optimization problem, $J'(A)$ also can be solved by an eigenvalue decomposition of $(cS_W + (1-c)S'_W)^{-1} S'_B$ where the k eigenvectors that corresponds to k largest eigenvalues are chosen as matrix A. In contrast to FLDA in which the number of extracted features is dependent to the number of available classes, i.e., $K - 1$, CIDA extracts more features according to the rank of S'_B. In fact, the number of extracted features of CIDA is $min\{m, N_s \times K - 1\}$ where almost increases with regard to the number of source domains.

3.3.3 Adaptive classifier

In the second phase, CIDA exploits an adaptive classifier to meet the following two complementary objectives: (1) the empirical risk minimization of prediction function on labeled

source data, which adapts across the source and target domains, (2) the rate of consistency maximization among the prediction function and the geometric data structure to preserve the input data structure. In the rest, the adaptive classifier and its objectives are expressed, in detail.

Learning based on the empirical risk minimization. The first objective of an adaptive classifier is to minimize the empirical risk of the prediction function on the labeled source data. The loss function is formulated as follows:

$$l(f(g(x_i)), y_i) = \sum_{i=1}^{n_s} max(0, 1 - y_i * f(g(x_i))) \tag{7}$$

where l computes the hinge loss, f denotes the prediction function of the classifier in order to predict the labels of labeled source data and $g(x)$ is the mapping function of a feature vector x, which transfers data into a new representation. Equation 7 computes the sum squared error of true and predicted label of f on source data.

Learning based on the data structure preservation. The second objective of an adaptive classifier is to maximize the consistency across the prediction function and the geometric data structure. We realize this objective by the manifold assumption. According to the manifold assumption, if two points x_s and x_t are close together in the underlying geometry of marginal distribution, it is induced that the conditional distribution of two points is similar as well, i.e., $Q_s(y_s \mid x_s) \approx Q_t(y_t \mid x_t)$ [53]. Therefore, the marginal distribution knowledge is utilized in order to learn a prediction function with good performance for target domain.

Generally, the structure of input data is modeled via a nearest neighbor graph that contains $n_s + n_t$ vertices on which each data point represents a node. For each data point, P nearest neighbors are determined and connected via edges. In order to determine the weight of each edge that connects the nodes x_i and x_j, the following weight function is employed:

$$W_{i,j} = e^{-\| \frac{(x_i - x_j)^2}{\delta} \|} \tag{8}$$

where δ is the normalization parameter to normalize matrix W and $W_{i,j}$ is the weight of nodes x_i and x_j. Then, the function M_f is defined in order to maximize the consistency between the prediction function and the manifold underlying the marginal distribution as follows:

$$M_f(P_s, P_t) = \sum_{i,j=1}^{n_s+n_t} (f(x_i) - f(x_j))^2 W_{ij} = \sum_{i,j=1}^{n_s+n_t} f(x_i) \overline{L}_{i,j} f(x_i) f(x_j) \tag{9}$$

where $\overline{L}$ is the normalized Laplacian matrix and P_s and P_t are the marginal distribution of source and target domains, respectively. Moreover, D is a diagonal matrix, which its elements are defined as follows:

$$D_{ii} = \sum_{j=1}^{n_s+n_t} W_{ij} \tag{10}$$

where D_{ii} illustrates the sum of i^{th} node weights with other nodes. Also, $L = D - W$ is considered as the un-normalized Laplacian matrix that L_{ii} shows the sum of node i weights with other nodes except itself. The normalized form of Laplacian matrix L is defined as follows [54]:

$$\overline{L} = I - D^{-\frac{1}{2}} W D^{\frac{1}{2}}. \tag{11}$$

where I is the identity matrix. Thus, the optimization problem of the adaptive classifier is defined as follows:

$$min_{f\in H}\sum_{i=1}^{n_s} l(f(g(x_i)), y_i) + \sigma \parallel f \parallel^2 + \gamma M_f(P_s, P_t) \tag{12}$$

where H is a set of classifiers and σ and γ are the regularization parameters and $\parallel f \parallel$ is the norm of f. Let the prediction function f is defined as $f(g(x_i)) = w^T\varphi(g(x_i))$ where w denotes the classifier parameters and φ shows the mapping function that transfers data from the original space to Hilbert space. Also, the kernel function k is defined as $k(g(x_i), g(x_j)) =< \varphi(g(x_i)), \varphi(g(x_j)) >$. According to the Representer theorem [55], the minimizer of the optimization problem in Eq. 12 can be formulated as:

$$f(g(x)) = \sum_{i=1}^{n_s+n_t} \alpha_i k(g(x_i), g(x_j)). \tag{13}$$

where α_i is the classifier parameters. If the Eqs. 7 and 9 are rewritten using Eq. 13 and incorporates the results into Eq. 12, the final optimization problem will be:

$$\alpha = argmin_{\alpha\in R^{n_s}}(Y - \alpha^T) + argmin_{\alpha\in R^{n_s+n_t}} tr(\gamma\alpha^T\mathbf{K}\overline{L}\mathbf{K}\alpha + \sigma\alpha^T\mathbf{K}\alpha) \tag{14}$$

where $\mathbf{K}$ denotes the kernel matrix. Therefore, the value of α is achieved from the following relation:

$$\alpha = (\sigma I + (R + \gamma\overline{L})\mathbf{K})^{-1}RY^T \tag{15}$$

where R is a diagonal matrix in which $R_{ii} = 1$ if $x_i \in X_s$ and $R_{ii} = 0$ otherwise. Moreover, Y is the label set. Now, we have a robust classifier that adapts the source and target domains. Algorithm 1 shows the complete procedure of CIDA. In each iteration, CIDA finds a projection matrix A and learns an adaptive classifier f based on the projected data to refine the pseudolabels of target data. In general, CIDA updates the projection matrix and classifier parameters in an iterative manner to predict the pseudotarget labels with superior accuracy. In the next section, the data description and the implementation details are explained.

3.4 Computational complexity

In this section, the computational complexity of CIDA is analyzed. According to Algorithm 1, the number of iterations of main loop is adjusted constant (e.g., 10), with $O(1)$. In more details, the computational cost is as follows: $O(K^2N_s^2)$ for computing S'_B, i.e., Line3; $O(N_sK)$ for computing S'_W, i.e., Line 5; $O((N_s + 1)K(n_s + n_t))$ for computing S_W, i.e., Line 6; $O(m^3)$ for solving the eigenvalue decomposition problem, i.e., Line 7; $O((n_s + n_t)^2)$ for adaptive classifier construction, i.e., Line 18. Since $N_s << K << m << (n_s + n_t)$, the total computational complexity of CIDA is $O((n_s + n_t)^2)$.

4 Experimental setup

In this section, the evaluation data are introduced and the implementation details are discussed.

Algorithm 1 Cross- and multiple-domains visual transfer learning via iterative Fischer linear discriminant analysis (CIDA)

Input: source and target data X; source domain labels y_s; regularization parameter c, γ, σ
Output: target domain labels y_t
$S'_B = \frac{1}{N_s^2} \sum_{i=1}^{K-1} \sum_{j=i+1}^{K} \sum_{u=1}^{N_s} \sum_{v=1}^{N_s} p_i^u p_j^v (\mu_i^u - \mu_j^v)(\mu_i^u - \mu_j^v)^T$
repeat until convergence
  $S'_W = \sum_{u=1}^{N_s} \sum_{i=1}^{K} (\mu_i^u - \mu_i^t)(\mu_i^u - \mu_i^t)^T$
  $S_W = \sum_{u=1}^{N_s+1} \sum_{i=1}^{K} \sum_{j=1}^{n_i} (x_i^j - \mu_i)(x_i^j - \mu_i)^T$, %%$(N_s+1)^{th}$ iteration denotes the target data process.
  solve the eigenvalue decomposition of $(cS_W + (1-c)S'_W)^{-1} S'_B$
  choose k eigenvectors corresponding to k largest eigenvalues as projection matrix A
  select a kernel function $k(x_i, x_j)$ and compute the kernel matrix $\mathbf{K}$ via $\mathbf{K}_{ij} = k(x_i, x_j)$ on projected data
  for $i = 1 : n_s + n_t$
    for $j = 1 : n_s + n_t$
      $W_{i,j} = e^{-\| \frac{(x_i - x_j)^2}{\delta} \|}$
      $D_{ii} = D_{ii} + W_{ij}$
  $\overline{L} = I - D^{-\frac{1}{2}} W D^{\frac{1}{2}}$
  construct the diagonal matrix $R_{ii} = \begin{cases} 1 & \text{if } x_i \in X_s \\ 0 & \text{otherwise} \end{cases}$
  $\alpha = (\sigma I + (R + \gamma \overline{L})\mathbf{K})^{-1} R Y^T$, %%$\alpha$ illustrates the classifier parameters
  learn the adaptive classifier f using $f(x) = \sum_{i=1}^{n_s+n_t} \alpha_i k(x_i, x), x \in D_t$
  predict the target data labels y_t via f
  update the pseudo target labels
end repeat
return the target domain labels y_t that are determined via classifier f

Table 1 Three benchmark domain adaptation datasets

Dataset	#instances	#features	#classes	Tag	Domain
Office	1410	800	10	A, W, D	Object
Caltech	1123	800	10	C	Object
PIE	11,554	1024	68	C05 (P1), C07 (P2), C09 (P3), C27 (P4), C29 (P5)	Face

4.1 Data description

CIDA is evaluated on three benchmark visual domain adaptation datasets that are summarized in Table 1. Office and Caltech-256 datasets are a collection of four different domains, which were investigated in [8,13,23,56] and contain the images of webcam domain (**W**) that were taken from a web camera with low resolutions, images in Amazon domain (**A**) that were downloaded from online merchants, images in DSLR domain (**D**) that were taken from a digital SLR camera with high resolutions, and images in Caltech-256 domain (**C**) that were downloaded and sieved from google images [57]. In our experiments, we use the public Office dataset published by Gong et al. [8] to compare the reported results with other state-of-the-arts.

We choose following ten common classes across Office and Caltech-256 datasets: *headphones*, *touring-bike*, *computer-monitor*, *computer-mouse*, *computer-keyboard*, *laptop-101*, *calculator*, *video projector*, *backpack*, and *coffee-mug*. Also, we utilize SURF features [58]

for all images and represent each image with 800-bin histograms from trained codebooks on Amazon images and standardize the histograms by z-score normalization.

We conduct three different scenarios to compare our proposed approach against other state-of-the-art domain adaptation approaches. (1) Single source domain in which one domain is considered as the training set and another domain is supposed as test set, i.e., $C \longrightarrow A, C \longrightarrow W, \ldots, D \longrightarrow W$. (2) Double source domains where two domains are selected as the training set and another domain is selected as test set, i.e., $A\&C \longrightarrow D, A\&C \longrightarrow W, \ldots, D\&W \longrightarrow C$. (3) Triple source domains in which three domains are considered as the training set and another domain is considered as test set, i.e. $A\&W\&D \longrightarrow C, \ldots, C\&A\&W \longrightarrow D$. Therefore, CIDA is evaluated on twenty-eight different tasks on Office dataset.

PIE is another benchmark domain adaptation dataset, which is the abbreviation of "Pose, Illumination, Expression." The dataset contains the face images of 68 individuals with 41,368 images of size 32×32 that were taken from 13 synchronized cameras and 21 flashes under different poses, illuminations and expressions. We select following five sets of PIE dataset, each pertaining to a different pose: PIE1 (C05, left pose), PIE2 (C07, upward pose), PIE3 (C09, downward pose), PIE4 (C27, frontal pose) and PIE5 (C29, right pose). In our experiments, we use the public PIE dataset published by Gong et al. [8] to have a fair comparison.

In order to evaluate the classification performance of CIDA versus other methods, four scenarios are designed as follows. (1) Single source domain in which one domain is considered as the training set and another domain is considered as test set, i.e., $P1 \longrightarrow P2, P1 \longrightarrow P3, \ldots, P5 \longrightarrow P4$. (2) Double source domains where two domains are selected as the training set and another domain is selected as test set, i.e., $P1\&P2 \longrightarrow P3, P1\&P2 \longrightarrow P4, \ldots, P4\&P5 \longrightarrow P3$. (3) Triple source domains in which three domains are chosen as the training set and another domain is chosen as test set, i.e., $P1\&P2\&P3 \longrightarrow P4, \ldots, P3\&P4\&P5 \longrightarrow P2$. (4) Quadruple source domains where four domains are selected as the training set and another domain is selected as test set, i.e., $P1\&P2\&P3\&P4 \longrightarrow P5, \ldots, P2\&P3\&P4\&P5 \longrightarrow P1$. Therefore, CIDA is tested on seventy-five different tasks.

4.2 Method evaluation

We systematically compare our CIDA results with two baseline machine learning methods, i.e. nearest neighbor (NN) and PCA, and other related state-of-the-art domain adaptation approaches including TCA [29], GFK [8], FIDOS [43], TSL [30], LTSL [59] and TSL-LRSR [60]. Since these methods are considered as dimensionality reduction approaches, we train a classifier on the labeled training data (i.e., NN), and then apply it on test data to predict the primary labels of the unlabeled test data. To validate the theoretical results of this research, the proposed method are compared with the best reported results of standard machine learning and other state-of-the-art domain adaptation methods.

4.3 Implementation details

In order to evaluate the performance of CIDA against other methods, the classification accuracy is utilized as the evaluation criterion. We set the number of iteration for convergence of CIDA to 10 and regulate $c = 0.71$ for Office+Caltech datasets and $c = 0.01$ for PIE datasets. Also, we consider $\sigma = 0.0001$ and $\gamma = 0.1$ for all datasets. In the next section, the parameter setting will be presented, in detail.

Table 2 Classification accuracy (%) on Office+Caltech-256 datasets

Target	Source	NN	PCA	TCA	GFK	FIDOS	TSL-LRSR	TSL	LTSL	CIDA
A	C	23.70	36.95	37.86	37.07	44.78	51.25	32.78	38.07	**52.30**
	W	22.96	31.00	31.00	30.42	31.32	34.13	27.24	37.60	**37.68**
	D	28.50	32.05	30.99	32.42	28.60	33.19	19.83	**45.80**	36.33
	C,D	24.01	36.53	43.11	38.01	45.30	51.78	29.75	31.40	**52.19**
	C,W	24.84	37.68	44.57	38.50	46.03	50.63	31.84	26.30	**52.92**
	D,W	29.02	36.43	34.34	35.14	35.39	36.43	21.71	37.60	**39.25**
	C,D,W	24.95	38.10	44.05	40.16	46.03	51.97	27.97	24.50	**51.98**
W	C	25.76	32.54	26.78	34.61	38.64	38.64	44.07	47.00	**47.80**
	A	29.83	35.59	28.47	36.68	37.29	36.61	34.24	39.10	**39.32**
	D	63.39	75.93	73.22	74.29	58.98	77.29	60.68	68.80	**80.68**
	C,D	32.20	59.32	**68.14**	55.86	48.47	59.32	48.14	37.5	63.73
	A,C	27.46	32.54	38.31	38.37	41.69	37.97	40.00	34.70	**48.14**
	A,D	41.02	64.07	**69.15**	54.75	53.56	62.71	36.95	48.00	60.34
	A,C,D	45.22	68.79	**71.34**	39.36	63.06	63.06	51.59	21.14	66.88

Table 2 continued

Target	Source	NN	PCA	TCA	GFK	FIDOS	TSL-LRSR	TSL	LTSL	CIDA
D	C	25.48	38.22	39.49	36.69	42.68	47.13	**49.04**	39.43	46.50
	A	25.48	27.39	34.39	36.34	32.48	38.85	**49.04**	45.71	43.95
	W	59.24	77.07	83.44	**77.58**	70.70	82.80	61.78	56.29	72.61
	C,W	45.22	72.61	**75.16**	66.62	65.61	67.52	49.68	30.29	70.06
	A,C	27.39	37.58	**51.59**	39.49	43.31	49.05	28.03	29.43	50.32
	A,W	49.68	66.88	71.97	66.56	73.25	74.52	38.22	34.57	**75.16**
	A,C,W	33.22	57.29	**60.00**	38.98	46.44	53.56	43.05	27.70	59.32
C	A	26.00	34.73	34.73	35.70	39.63	**43.37**	21.10	19.90	41.76
	D	26.27	29.65	30.28	29.23	27.69	31.61	22.71	21.90	**33.04**
	W	19.86	26.36	26.36	29.72	25.29	29.83	22.53	19.70	**33.48**
	A,D	26.36	35.80	40.69	37.77	41.14	**45.24**	20.66	19.60	45.06
	A,W	26.63	36.15	39.27	38.09	39.89	45.06	22.71	15.90	**45.06**
	D,W	24.49	30.90	33.48	31.56	30.10	31.61	18.97	20.80	**34.46**
	A,D,W	26.71	37.67	39.27	37.36	41.59	44.79	20.30	15.60	**44.97**
Avg. single source		31.37	39.79	39.75	40.90	39.84	45.39	37.09	39.99	**47.12**
Avg. double source		31.53	45.54	50.82	45.06	46.98	50.99	32.22	30.51	**53.06**
Avg. triple source		32.53	50.46	53.66	38.97	49.28	53.35	35.73	22.24	**55.79**
Overall Avg.		31.60	43.77	46.48	42.40	44.24	48.92	34.80	33.39	**50.90**

5 Experimental results and discussion

In this section, we compare the performance of our proposed method with eight related state-of-the-art and baseline methods on benchmark visual domain adaptation datasets.

5.1 Results evaluation

Object recognition: The classification accuracy of CIDA and other methods on Office+Caltech datasets is reported in Table 2 that is considered for single, double and triple source domains, respectively. In order to interpret better, the results are visualized in Figs. 1, 2 and 3. We comprehend the following observations from the reported experimental results. (1) CIDA gains

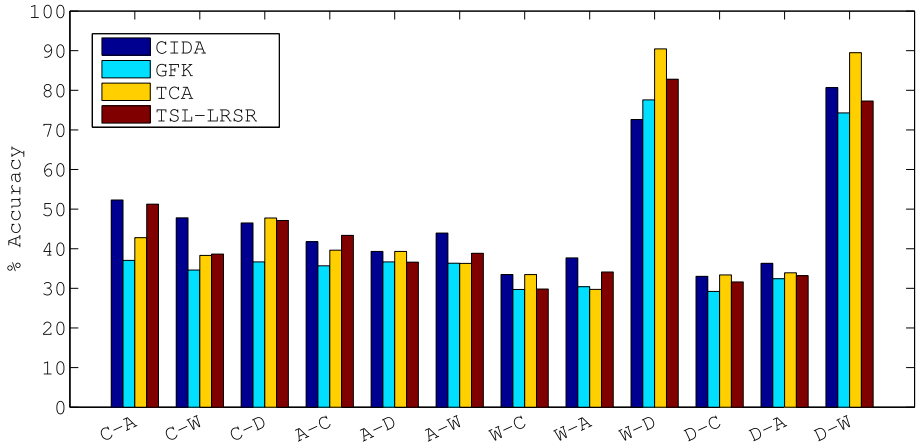


Fig. 1 Classification accuracy (%) of single source domain scenario on Office and Caltech-256 datasets. CIDA outperforms other dimensionality reduction and DA approaches in 7 out of 12 tasks using NN classifier

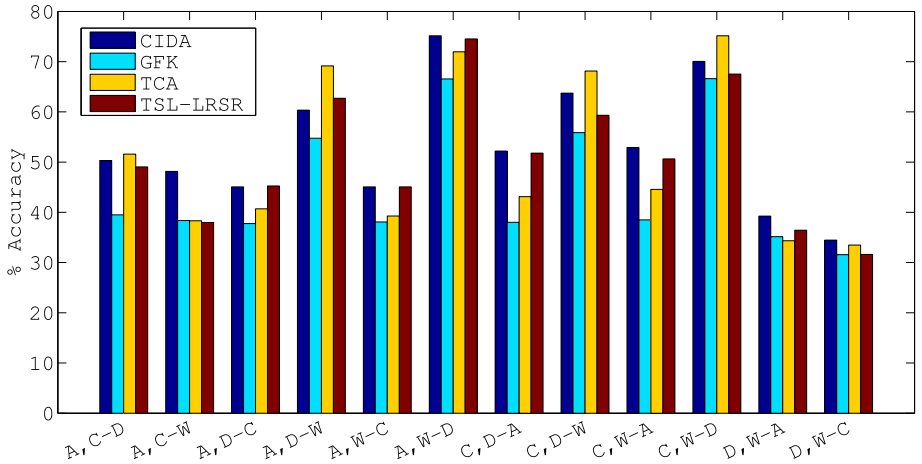


Fig. 2 Classification accuracy (%) of double source domains scenario on Office and Caltech-256 datasets. CIDA outperforms other dimensionality reduction and DA approaches in 7 out of 12 tasks using NN classifier

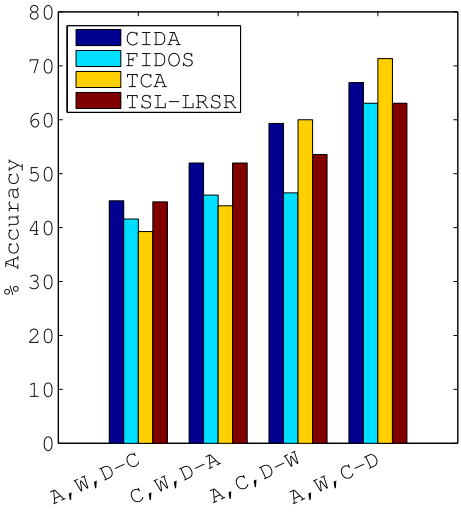


Fig. 3 Classification accuracy (%) of triple source domains scenario on Office and Caltech-256 datasets. CIDA outperforms other dimensionality reduction and DA approaches in 2 out of 4 tasks using NN classifier

best performance in terms of the average classification accuracy (**47.12%**) in single source domain settings where it performs better than the state-of-the-art domain adaptation methods in 7 out of 12 DA tasks. Moreover, due to the mismatched distribution among the training and test datasets, the performance improvement of CIDA over NN is (**15.75%**). This substantiates that CIDA performs robustly and effectively for domain image classification tasks. (2) CIDA achieves a significant improvement (**2.07%**) compared to the best baseline method TSL-LRSR in double source domain settings where the performance of CIDA is higher than the novel domain adaptation methods in 7 out of 12 DA tasks. Also, CIDA obtains (**21.53%**) performance improvement compared to NN. (3) The performance improvement of CIDA in comparison with the best baseline method TSL-LRSR in the triple source domain settings is (**2.44%**) where CIDA outperforms the modern domain adaptation methods in 2 out of 4 DA tasks. In addition, CIDA has (**23.26%**) improvement over NN classifier.

Face recognition: We summarize the classification accuracy of CIDA and other methods on PIE datasets in Table 3 that is considered for single, double, triple and quadruple source domains, respectively. In order to interpret better, the results are visualized in Figs. 4, 5, 6 and 7. We get the following observations from the reported experimental results. (1) CIDA obtains remarkable improvement in terms of the average classification accuracy (**7.31%**) compared to the best method TSL-LRSR in single source domain settings, which outperforms all other domain adaptation methods in 15 out of 20 DA tasks. Also, CIDA obtains (**17.05%**) improvement compared to NN. (2) CIDA achieves the significant improvement in terms of the average classification accuracy (**4.42%**) in comparison with the best method TSL-LRSR in double source domain settings where CIDA performs classification task with more accuracy in 14 out of 30 DA tasks. In addition, CIDA achieves (**33.51%**) performance improvement over NN. (3) The improvement accuracy of CIDA is (**3.33%**) in terms of the classification accuracy in comparison with the best method TSL-LRSR in triple source domain settings where CIDA outperforms other methods in 12 out of 20 DA tasks. CIDA also gains (**34.86%**) performance improvement compared to NN. (4) CIDA achieves (**2.26%**) performance improvement in average classification accuracy compared to the best baseline method TSL-LRSR in quadruple source domain settings where it outperforms other methods in 3 out of 5 DA tasks. Moreover, CIDA gains (**33.86%**) in comparison with NN. In the rest,

Table 3 Classification accuracy (%) on Multi-PIE datasets

Target	source	NN	PCA	TCA	GFK	FIDOS	TSL-LRSR	TSL	LTSL	CIDA
P1	P2	24.49	25.78	21.52	29.89	27.79	**44.96**	45.47	44.84	38.60
	P3	21.37	28.54	19.00	31.64	33.19	28.87	42.44	8.28	**44.48**
	P4	32.95	41.33	30.79	43.13	51.35	66.48	36.49	8.71	**72.09**
	P5	18.49	19.06	14.29	21.71	21.13	38.63	37.39	39.67	**39.83**
	P2,P3	10.98	31.42	23.41	36.24	32.23	**60.26**	45.05	36.91	58.19
	P2,P4	49.37	41.03	32.50	45.42	46.82	66.12	43.13	72.88	**74.04**
	P2,P5	10.71	25.33	22.45	31.07	28.39	57.95	41.84	**67.68**	51.32
	P3,P4	49.64	41.15	30.82	45.48	44.72	43.22	41.78	57.99	**71.85**
	P3,P5	11.34	28.93	21.58	32.76	31.3	64.44	42.92	**56.39**	54.83
	P4,P5	50.18	40.43	31.03	45.21	46.55	**79.62**	41.45	74.44	71.37
	P2,P3,P4	48.80	41.03	32.05	46.93	42.26	61.94	42.65	63.12	**74.67**
	P2,P3,P5	11.04	30.31	25.12	36.67	29.65	**72.24**	43.22	53.59	59.57
	P2,P4,P5	49.04	40.88	32.71	46.3	43.37	**73.35**	42.44	48.97	73.05
	P3,P4,P5	49.70	40.1	31.63	46.05	42.26	60.53	41.36	49.84	**71.22**
	P2,P3,P4,P5	48.83	40.19	32.56	47.27	39.77	71.37	42.47	38.04	**74.28**

Table 3 continued

Target	source	NN	PCA	TCA	GFK	FIDOS	TSL-LRSR	TSL	LTSL	CIDA
P2	P1	26.09	20.81	15.59	24.44	20.87	30.08	33.46	8.22	**37.63**
	P3	41.01	39.17	34.25	43.56	36.28	29.04	37.02	**58.19**	40.7
	P4	62.68	51.5	52.85	48.83	51.87	61.76	39.29	17.97	**65.56**
	P5	24.19	20.87	14.00	23.91	22.9	33.46	34.62	7.09	**34.62**
	P1,P3	47.08	38.98	34.44	39.02	34.68	49.05	38.49	**55.16**	54.27
	P1,P4	29.83	46.35	54.76	45.91	40.88	66.97	39.78	65.75	**72.74**
	P1,P5	31.61	21.42	14.24	26.63	23.33	38.74	40.52	**57.57**	54.39
	P3,P4	45.37	48.93	51.44	49.64	40.7	74.59	40.82	**79.56**	74.59
	P3,P5	52.06	41.62	34.38	43.59	37.14	**49.05**	40.58	41.81	47.27
	P4,P5	34.07	45.18	52.36	47.81	44.44	**72.38**	39.1	43.28	69.55
	P1,P3,P4	45.37	46.16	52.61	48.15	38.61	69.37	41.68	61.23	**76.37**

Table 3 continued

Target	source	NN	PCA	TCA	GFK	FIDOS	TSL-LRSR	TSL	LTSL	CIDA
	P1,P3,P5	51.44	39.23	34.5	41.03	35.11	**68.63**	41.80	50.46	61.94
	P1,P4,P5	34.56	41.68	53.16	44.48	36.89	71.09	41.31	73.19	**75.26**
	P3,P4,P5	49.66	47.94	51.44	50.48	40.15	**75.75**	39.72	44.13	75.26
	P1,P3,P4,P5	49.66	46.9	52.12	47.87	38.31	73.79	42.05	**83.14**	79.37
P3	P1	26.59	26.1	16.24	29.99	27.51	**55.58**	33.58	50.62	47.55
	P2	46.63	42.65	32.9	44.94	34.8	34.87	42.59	**54.67**	44.91
	P4	73.22	61.4	64.52	54.45	66.05	70.83	43.01	33.94	**72.79**
	P5	28.31	22.79	14.89	26.77	26.41	35.78	35.11	9.41	**50.67**
	P1,P2	48.59	43.14	32.97	42.51	38.66	54.35	38.24	**61.08**	61.03
	P1,P4	28.74	57.54	65.5	55.06	54.96	**76.16**	40.44	65.78	75
	P1,P5	28.74	24.51	14.4	29.28	30.39	56.99	39.64	49.85	**65.81**
	P2,P4	44.42	59.01	63.05	58.28	56.13	**78.86**	42.89	66.54	77.08
	P2,P5	52.27	40.87	34.93	45.39	40.75	51.23	43.44	57.47	**63.24**
	P4,P5	31.43	58.21	64.58	56.72	56.99	64.34	41.48	67.24	**78.00**
	P1,P2,P4	44.06	56.74	63.3	57.54	51.96	**78.49**	41.79	71.86	77.82
	P1,P2,P5	50.67	41.91	33.33	43.5	39.46	61.21	41.91	64.93	**70.16**
	P1,P4,P5	31.37	53.55	64.71	53.59	50.31	**81.07**	41.67	51.13	77.39
	P2,P4,P5	46.63	57.84	63.17	58.92	53.25	67.34	42.40	54.33	**80.94**
	P1,P2,P4,P5	46.32	55.39	63.24	57.56	49.75	**80.51**	42.34	61.44	79.23

Table 3 continued

Target	source	NN	PCA	TCA	GFK	FIDOS	TSL-LRSR	TSL	LTSL	CIDA
P4	P1	30.67	38.21	25.44	41.56	39.62	68.37	35.27	10.82	**69.18**
	P2	54.07	51.16	45.27	52.77	46.59	57.16	45.63	24.36	**69.84**
	P3	46.53	52.36	49.2	53.57	53.08	64.19	41.12	42.45	**69.42**
	P5	31.24	28.00	21.06	31.07	31.12	43.23	46.08	6.05	**62.06**
	P1,P2	44.55	52.33	48.24	54.12	49.8	87.5	40.13	50.56	**87.92**
	P1,P3	43.89	56.35	53.47	57.48	54.01	74.89	40.1	51.06	**84.11**
	P1,P5	42.14	35.87	25.98	43.02	41.72	80.71	41.78	49.82	**84.83**
	P2,P3	17.48	59.06	57.86	61.34	52.93	82.64	54.49	55.28	**87.11**
	P2,P5	16.49	50.89	45.48	52.75	46.8	**87.98**	51.61	60.07	83.96
	P3,P5	14.93	54.07	49.86	56.07	51.1	74.41	50.32	62.22	**78.61**
	P1,P2,P3	43.62	59.57	60.62	63.8	54.25	**93.12**	43.32	83.92	91.53
	P1,P2,P5	41.93	50.2	48.24	54.3	47.79	82.19	43.71	58.91	**90.42**
	P1,P3,P5	41.69	56.26	52.96	58.47	52.15	80.41	43.26	56.44	**89.22**
	P2,P3,P5	17.21	59.9	57.64	61.87	51.19	85.91	54.76	73.71	**88.86**
	P1,P2,P3,P5	41.60	59.51	60.41	63.12	52.45	91.38	45.42	73.56	**92.34**

Table 3 continued

Target	source	NN	PCA	TCA	GFK	FIDOS	TSL-LRSR	TSL	LTSL	CIDA
P5	P1	16.67	15.99	11.83	21.26	22.79	28.12	26.78	11.45	**36.09**
	P2	26.53	22.49	14.64	26.97	27.63	19.42	31.43	**51.32**	34.01
	P3	26.23	27.21	16.18	29.3	35.29	30.76	30.27	33.40	**46.81**
	P4	37.19	30.64	20.04	32.13	42.65	48.35	32.11	9.02	**59.25**
	P1,P2	32.90	22.43	17.83	26.21	27.94	51.9	28.00	**62.63**	52.08
	P1,P3	29.96	26.47	18.38	31.22	33.27	42.95	28.86	**59.41**	52.82
	P1,P4	25.55	25.8	22.3	32.47	34.87	**71.38**	30.15	53.07	59.01
	P2,P3	36.58	28.92	19.12	31.12	35.78	45.1	37.25	33.33	**57.05**
	P2,P4	33.03	29.9	20.47	34.9	37.5	39.4	32.9	51.9	**63.24**
	P3,P4	30.33	30.58	20.96	34.44	39.77	53.86	32.84	24.4	**64.22**
	P1,P2,P3	36.95	27.45	20.1	32.51	30.7	60.54	30.64	46.44	**62.13**
	P1,P2,P4	33.76	28.25	22.79	34.19	31.74	62.07	30.58	61.49	**64.52**
	P1,P3,P4	31.43	28.00	22.79	34.61	34.25	64.03	30.70	59.64	**63.66**
	P2,P3,P4	35.17	31.25	22.3	36.83	36.21	55.51	32.84	56.45	**67.46**
	P1,P2,P3,P4	35.91	28.43	23.41	36.32	31.31	63.3	31.25	27.03	**66.42**
Avg. single source		34.76	33.30	26.73	35.60	35.95	44.50	37.46	26.52	**51.81**
Avg. double source		34.14	40.22	35.96	43.04	41.15	63.23	40.33	56.37	**67.65**
Avg. triple source		39.71	43.91	42.26	47.51	42.08	71.24	40.59	59.19	**74.57**
Avg. quadruple source		44.47	46.08	46.35	50.43	42.32	76.07	40.71	56.64	**78.33**
Overall Avg.		36.37	39.66	35.72	42.63	40.05	61.02	39.64	49.07	**65.81**

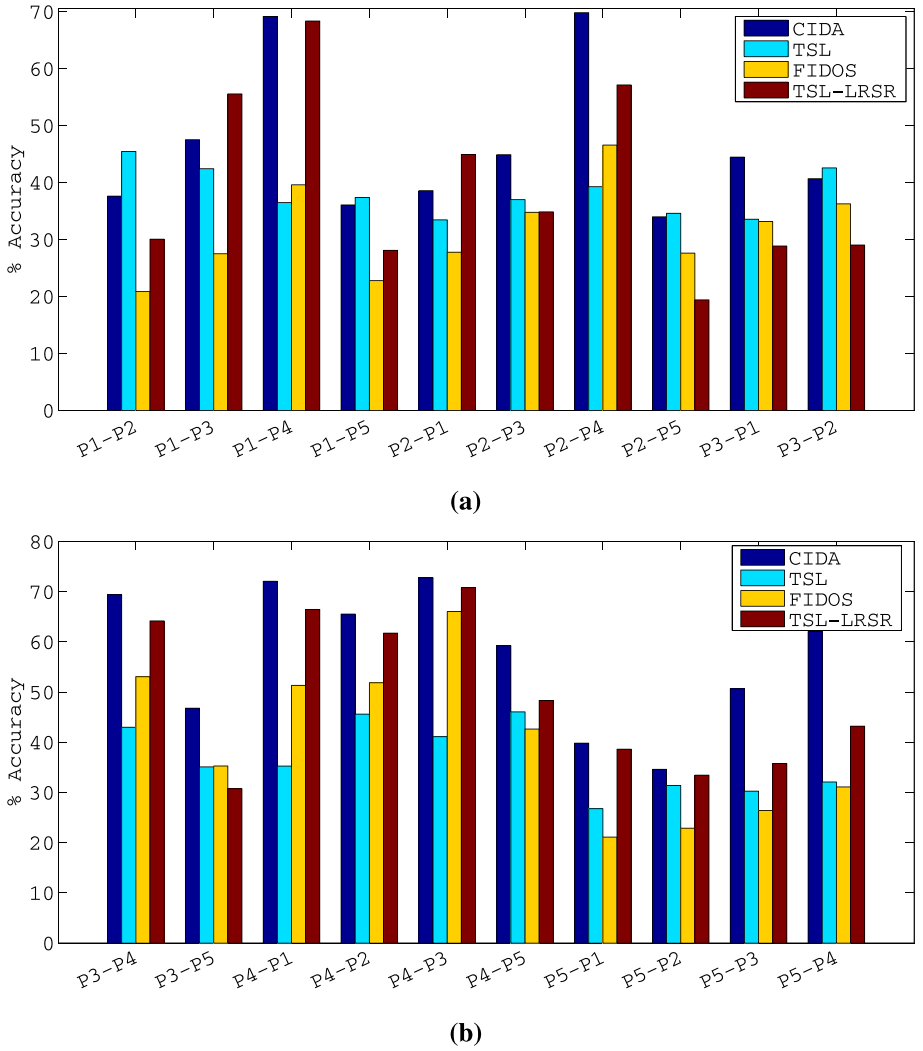


Fig. 4 Classification accuracy (%) of single source domain scenario on PIE datasets. CIDA outperforms other dimensionality reduction and DA approaches in 15 out of 20 tasks using NN classifier. **a** the first ten tasks, **b** the second ten tasks

we compare CIDA with other methods, in detail. In the rest, the performance of compared methods is investigated with detail.

PCA is probably the most popular dimensionality reduction approach, which attempts to discover a shared representation across domains besides the maximum variance preservation on the new representation. Since PCA does not consider the distribution difference between domains, it does not perform well versus domain adaptation baseline methods. Nevertheless, PCA obtains better performance against NN.

TCA is a novel domain adaptation method that learns common transfer components among domains and maps the original data into the new subspace according to the transferred components. TCA is affected by te following two major restrictions: (1) TCA projects domains

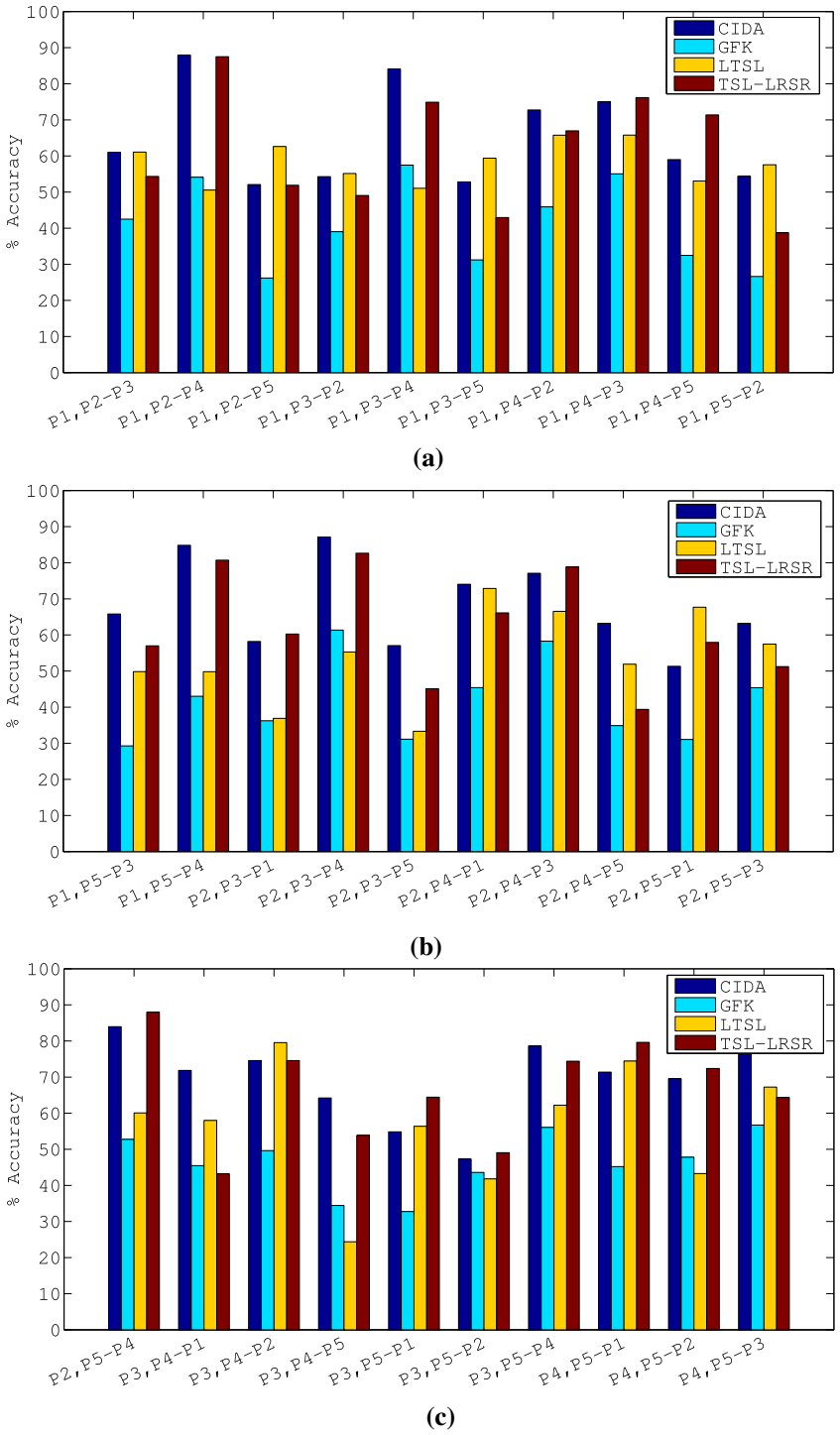


Fig. 5 Classification accuracy (%) of double source domains scenario on PIE datasets. CIDA outperforms other dimensionality reduction and DA approaches in 14 out of 30 tasks using NN classifier. **a** the first ten tasks, **b** the second ten tasks, **c** the third ten tasks

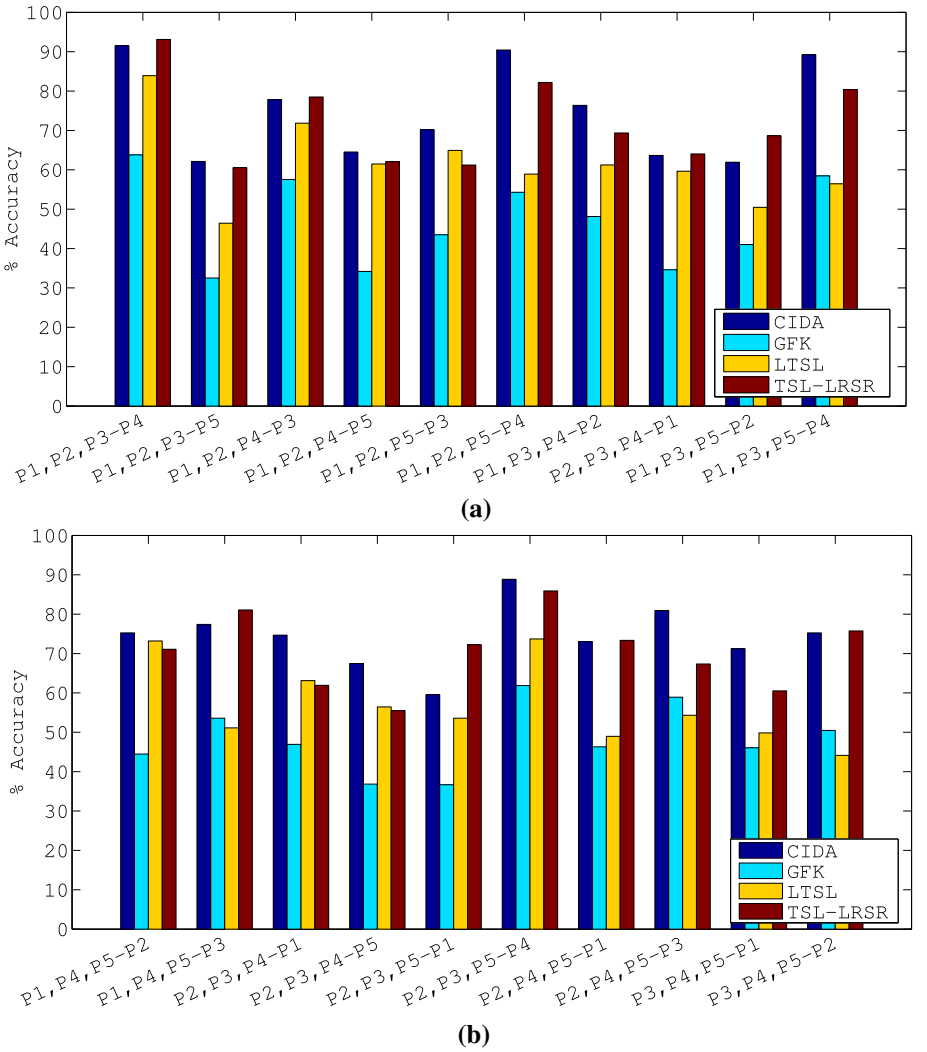


Fig. 6 Classification accuracy (%) of triple source domains scenario on PIE datasets. CIDA outperforms other dimensionality reduction and DA approaches in 12 out of 20 tasks using NN classifier. **a** the first ten tasks, **b** the second ten tasks

into an unsupervised manner and does not consider the label information of source data, and (2) TCA only reduces the marginal distribution difference across domains and does not consider the conditional distribution difference. However, CIDA benefits from the source domain labels in constructing the shared low-dimensional subspace and also discriminates across various classes.

GFK is another well-known DA approach that transfers domains into a shared low-dimensional subspace besides reducing the marginal distribution difference. The main limitation of GFK is the low-sized dimension of the embedded subspace that causes the original data represented inaccurately on the embedded subspace. However, CIDA learns an

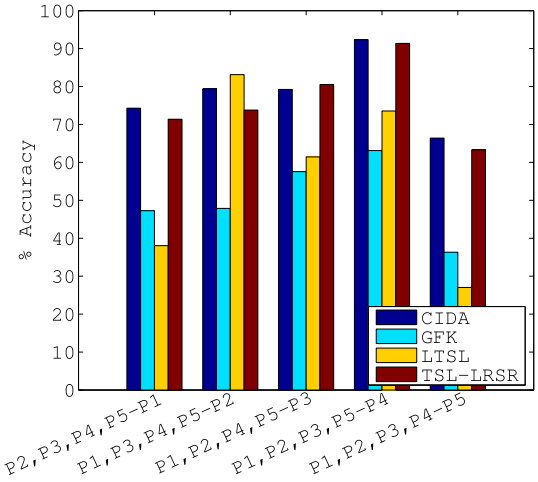
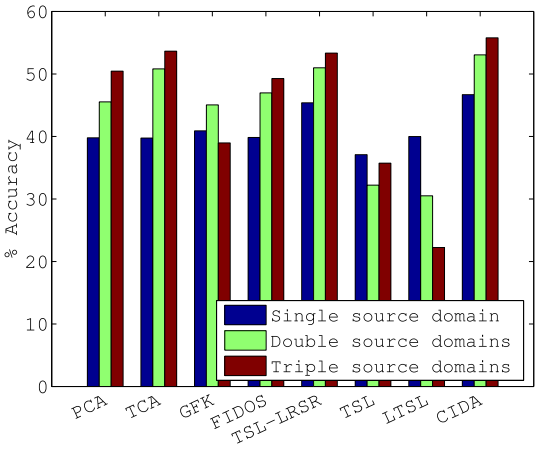
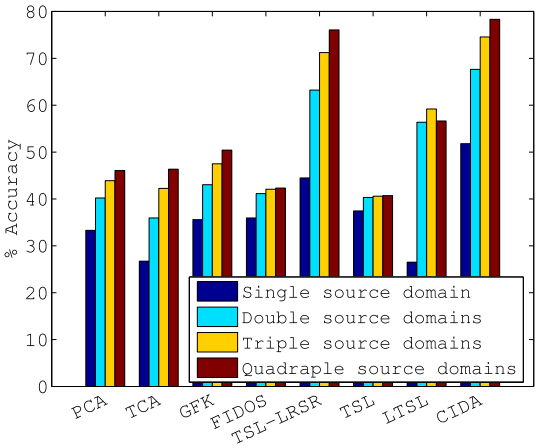


Fig. 7 Classification accuracy (%) of quadruple source domains scenario on PIE datasets. CIDA outperforms other dimensionality reduction and DA approaches in 3 out of 5 tasks using NN classifier



(a)



(b)

Fig. 8 Average classification accuracy (%) of different methods under various scenarios. GFK, TSL and LTSL perform poorly on multiple source scenario tasks. However, CIDA systematically benefits from the available knowledge in different domains to adapt the input data. **a** Office+Caltech datasets, **b** PIE datasets

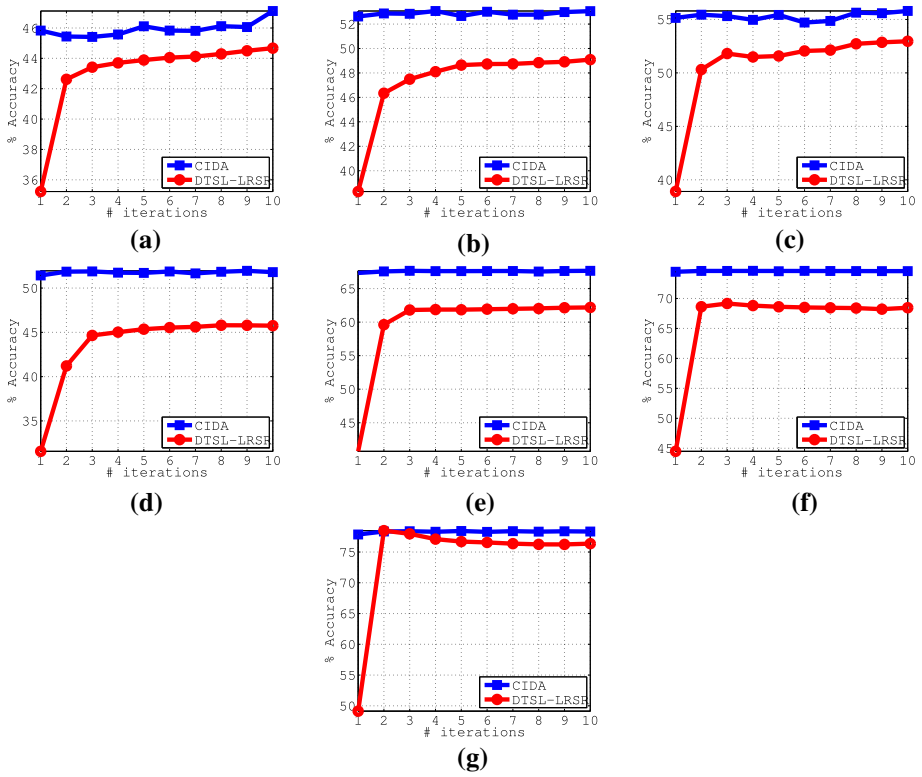


Fig. 9 Average classification accuracy (%) with respect to the number of iterations for Office+Caltech and PIE datasets under different scenarios. CIDA predicts the accurate labels to target samples in an iterative manner. Almost, the predicted labels of each stage are better than the previous one. **a**, **b** and **c** are single, double and triple source domain, respectively, on Office+Caltech datasets. **d**–**g** are single, double, triple and quadruple source domain, respectively, on PIE datasets

accurate shared subspace that exactly represents the original data according to the high rank of between class scatter matrix.

TSL is another noticeable method that adapts the marginal distribution of source and target domains based on the kernel density estimation. TSL suffers from following three important weaknesses. (1) TSL does not reduce the conditional distribution difference among the source and target domains due to its dependence to the distribution density. (2) Since TSL is sensitive to data size, it does not describe the distribution of data using the kernel density estimation when the target domain contains a few data. (3) Even with enough data, TSL has convergence problem when data have a large scale such as PIE dataset. But, CIDA performs well on both small and big datasets and has considerable improvement against TSL.

LTSL is a novel framework that transfers data into a shared subspace such that some combination of the source samples represent the target samples. Also, LTSL utilizes a low-rank constraint to preserve the structure of the source and target domains. However, there are two reasons that LTSL is insufficient in domain adaptation and subspace alignment. (1) In LTSL, since the subspace learning and reconstruction process are independent, domain adaptation performance is limited. (2) In TSL, the target data are only reconstructed with the

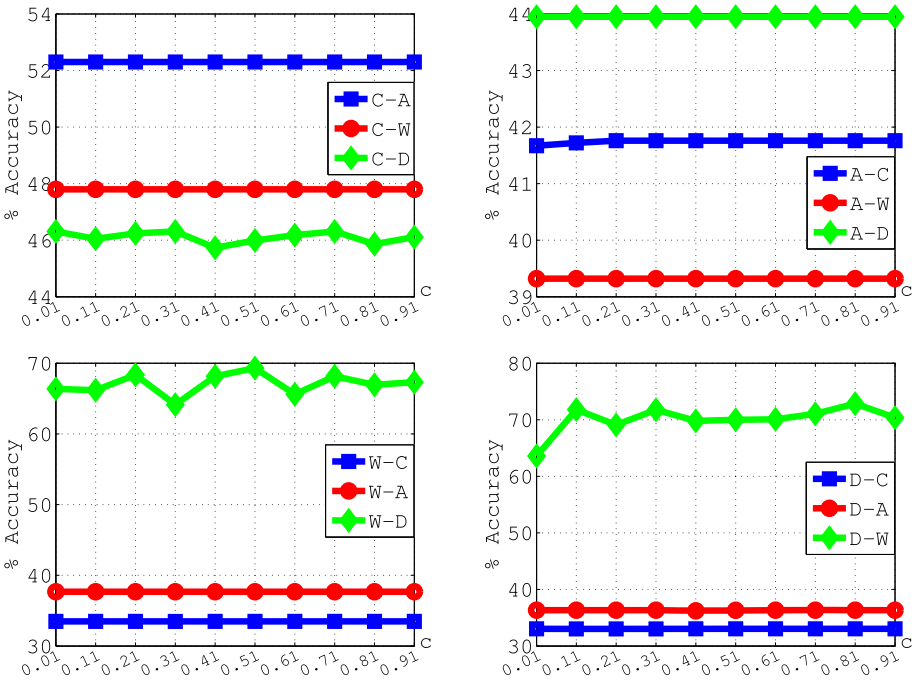


Fig. 10 Parameter evaluation with respect to the classification accuracy (%) and parameter c, for Office+Caltech datasets under single source domain scenario. CIDA is not sensitive to the value of c in most cases

source data. Thus, LTSL performs poorly on small dataset. However, CIDA jointly benefits from the representation and classification learning to adapt the source and target domains.

FIDOS is a modern framework that constructs the shared low-dimensional subspace besides the reduction of distribution difference and preserving the discrimination across classes. FIDOS similar to CIDA is an FLDA-based approach, but it is only sufficient for the strong related datasets.

TSL-LRSR is another approach that transfers the source and target data into a shared subspace in which each target data are reconstructed using the composition of the source samples. TSL-LRSR employs the low-rank and sparse constraints on the reconstruction matrix to preserve the local and global structure of data. Moreover, TSL-LRSR learns a flexible linear classifier and a non-negative label relaxation matrix to maximize the margins across various classes. In spite of the complicated structure of TSL-LRSR, CIDA benefits from simple and robust optimization problem that adapts the distribution mismatch.

5.2 Multi-source domain adaptation problems

We remark that some of methods such as GFK, TSL and LTSL perform poorly on the experiments on multiple source scenarios (according to Fig. 8). In fact, the multi-source scenario causes the severe multi-modality problem across various classes and much distribution mismatches across domains. In this way, the learned classifier on the source domains performs poorly to predict the labels of target domain. However, CIDA systematically benefits from the available knowledge in different domains to adapt the input data. Following three major

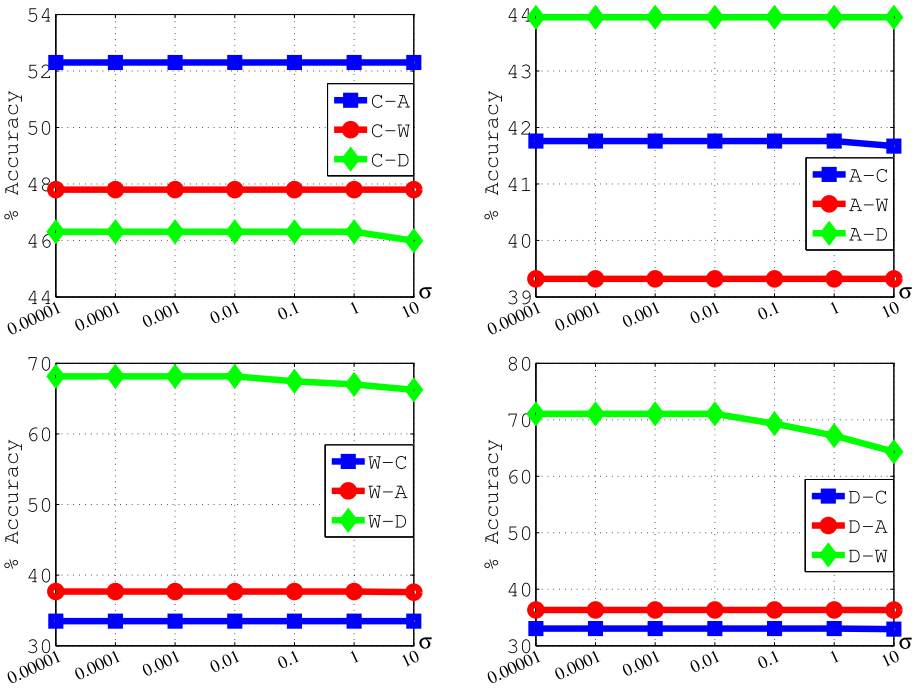


Fig. 11 Parameter evaluation of CIDA with respect to the classification accuracy (%). The parameter σ on Office+Caltech datasets under single source domain scenario. CIDA is not sensitive to the value of σ in most cases. Also, CIDA achieves acceptable results with small values of σ. Indeed, we consider $\sigma \in [0.00001\ 0.01]$ for all datasets

factors contribute to the supremacy of our approach against other DA and machine learning approaches: 1) CIDA maximizes the marginal distribution difference of the various classes of source and target domains, 2) CIDA minimizes the distribution difference between the same classes of the source and target domains, 3) CIDA minimizes the amount of variance between the samples of each class.

5.3 Effectiveness evaluation

We conduct experiments in 10 iterations to evaluate the performance of CIDA and the best baseline method TSL-LRSR via comparing their average classification accuracy. We run TSL-LRSR and CIDA on all datasets under different scenarios. Since CIDA has almost similar behavior against different methods, we only report analysis of CIDA and TSL-LRSR. Our results are reported in Fig. 9. In the next section, the convergence of CIDA will be investigated. As it is understood from the figures, in all scenarios, CIDA outperforms the best baseline method TSL-LRSR. Our proposed approach significantly reduces the distribution difference among the source and target domains. Also, CIDA employs an adaptive classifier to adapt the source and target domains. CIDA predicts the accurate labels of target samples in an iterative manner. Almost, the predicted labels of each stage are better than the previous one.

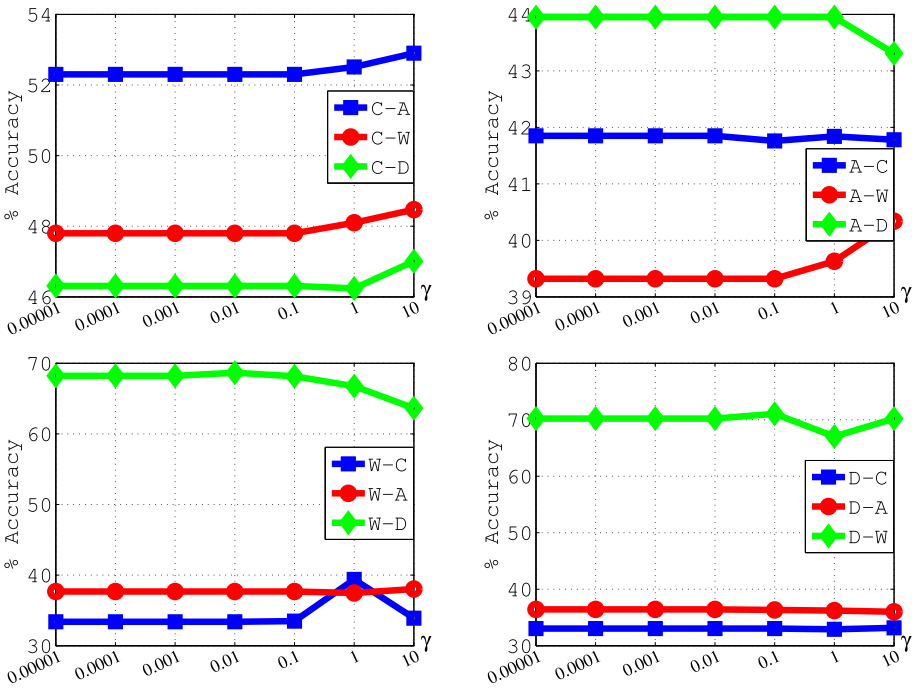


Fig. 12 Parameter evaluation of CIDA with respect to the classification accuracy (%). The parameter γ, on Office+Caltech datasets under single source domain scenario. CIDA is not sensitive to the value of γ in period [0.00001 0.1]

5.4 Impact of parameter settings

The performance of CIDA is evaluated regarding to the different values of parameters in various situations. In general, we adjust three regularization parameters c, σ and γ for CIDA on various datasets. Since CIDA has similar behavior on all datasets, we just report the results of CIDA on Office+Caltech datasets due to space limitation.

In Fig. 10, the experimental results of Office+Caltech datasets are reported for evaluating the parameter c. We run CIDA with respect to the various values of c. We report the classification accuracy of CIDA with $c \in [0.01\ \ 0.91]$ on 12 Office+Caltech datasets. As is clear from the figures, CIDA is not sensitive to the value of c in most cases.

Figure 11 illustrates the experimental results for parameter σ on Office+Caltech datasets. We plot classification accuracy of CIDA with $\sigma \in [0.00001\ \ 10]$ on 12 Office+Caltech datasets. As is clear from the plots, CIDA is not sensitive to the value of σ in most cases. Also, CIDA achieves the acceptable results with small values of σ. Indeed, we consider $\sigma \in [0.00001\ \ 0.01]$ for all datasets.

Figure 12 shows the experimental results of CIDA with respect to $\gamma \in [0.0000110]$ on Office+Caltech datasets. The results demonstrate that CIDA is not sensitive to the value of γ in period [0.00001 0.1].

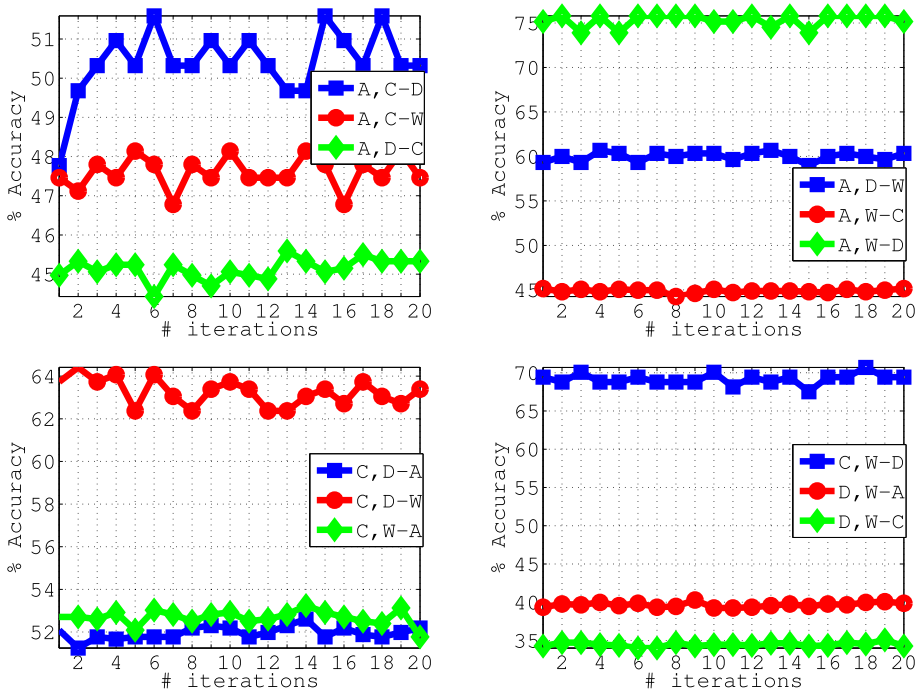


Fig. 13 Convergence evaluation of CIDA with respect to the classification accuracy (%) in 20 iterations on Office+Caltech datasets under double source domains scenario. CIDA is converged in 10 iteration in most cases

5.5 Convergence evaluation

The convergence property of CIDA is validated by conducting the general experiments on Office+Caltech datasets under double source domains scenario. Figure 13 indicates the classification accuracy of CIDA in 20 iterations. As is clear from the figures, CIDA is converged in 10 iteration in most cases.

6 Conclusion and future work

In this paper, we proposed a novel cross- and multiple-domains visual transfer learning via iterative Fischer linear discriminant analysis (CIDA) approach for visual domain adaptation. Compared to the existing works, CIDA is the first attempt to handle the challenges of both defective transformation and unevaluated discriminant analysis. CIDA trains a domain-invariant classifier with minimization of structural risk and customized FLDA-based adaptation. We also provide a hybrid solution to exploit the adaptive classifier.

The effectiveness of CIDA is validated from a variety of perspectives such as results, effectiveness, parameters and convergence, where its performance are compared with eight state-of-the-art baseline methods on various benchmark visual domain adaptation datasets under different scenarios. The experimental results indicate that CIDA significantly outperforms other DA methods specifically when the number of source domain increases. In the

future, we plan to generalize our approach to cope with non-linear feature extraction, utilizing online transfer learning and employing inductive transfer learning.

References

1. Mahya A, Jafar T (2021) Metric transfer learning via geometric knowledge embedding. Appl Intell 51(2):921–934
2. Karimpour M, Saray SN, Tahmoresnezhad J, Pourmahmood AM (2020) Multi-source domain adaptation for image classification. Mach Vis Appl 31(6):1–19
3. John B, Mark D, Fernando P et al (2007) Biographies, bollywood, boom-boxes and blenders: domain adaptation for sentiment classification. ACL 7:440–447
4. Sanodiya RK, Paul D, Yao L, Mathew J, Juhi A (2020) A feature selection approach to visual domain adaptation in classification. In: International conference on neural information processing, Springer, pp 77–89
5. Wang F, Ding Y, Liang H, Wen J (2021) Discriminative and selective pseudo-labeling for domain adaptation. In: International conference on multimedia modeling, Springer, pp 365–377
6. Blitzer J, McDonald R, Pereira F (2006) Domain adaptation with structural correspondence learning. In: Proceedings of the 2006 conference on empirical methods in natural language processing, Association for Computational Linguistics, pp 120–128
7. Chen M, Weinberger KQ, Blitzer J (2011) Co-training for domain adaptation. In: Advances in neural information processing systems, pp 2456–2464
8. 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 (CVPR), IEEE, pp 2066–2073
9. Gopalan R, Li R, Chellappa R (2011) Domain adaptation for object recognition: an unsupervised approach. In: 2011 IEEE international conference on computer vision (ICCV), IEEE, pp 999–1006
10. Tahmoresnezhad J, Hashemi S (2016) Transductive transfer learning via maximum margin criterion. Sci Iran 23(3):1239–1250
11. Bergamo A, Torresani L (2010) Exploiting weakly-labeled web images to improve object classification: a domain adaptation approach. In: Advances in neural information processing systems, pp 181–189
12. Kumar A, Saha A, Daume H (2010) Co-regularization based semi-supervised domain adaptation. In: Advances in neural information processing systems, pp 478–486
13. Kate S, Brian K, Mario F, Trevor D (2010) Adapting visual category models to new domains. Comput Vis ECCV 2010:213–226
14. Ben-David S, Blitzer J, Crammer K, Pereira F (2007) Analysis of representations for domain adaptation. In: Advances in neural information processing systems, pp 137–144
15. Ciprian C, Alex A (2006) Adaptation of maximum entropy capitalizer: little data can help a lot. Comput Speech Lang 20(4):382–399
16. Hal Daume III and Daniel Marcu (2006) Domain adaptation for statistical classifiers. J Artif Intell Res 26:101–126
17. Jafar T, Sattar H (2017) Exploiting kernel-based feature weighting and instance clustering to transfer knowledge across domains. Turk J Electr Eng Comput Sci 25(1):292–307
18. Mansour Y, Mohri M, Rostamizadeh A (2009) Domain adaptation with multiple sources. In: Advances in neural information processing systems, pp 1041–1048
19. 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
20. Crammer K, Earns M, Wortman J (2008) Learning from multiple sources. J Mach Learn Res, 9(Aug):1757–1774
21. Tahmoresnezhad J, Hashemi S (2015) Common feature extraction in multi-source domains for transfer learning. In: 2015 7th conference on information and knowledge technology (IKT), IEEE, pp 1–5
22. Fan W, Davidson I, Zadrozny B, Yu PS (2005) An improved categorization of classifier's sensitivity on sample selection bias. In: Fifth IEEE international conference on data mining, IEEE
23. Gong B, Grauman K, Sha F (2013) Connecting the dots with landmarks: discriminatively learning domain-invariant features for unsupervised domain adaptation. In: International conference on machine learning, pp 222–230
24. Jing J, ChengXiang Z (2007) Instance weighting for domain adaptation in nlp. ACL 7:264–271
25. Sugiyama M, Nakajima S, Kashima H, Buenau PV, Kawanabe M (2008) Direct importance estimation with model selection and its application to covariate shift adaptation. In: Advances in neural information processing systems, pp 1433–1440

26. Samaneh R, Jafar T, Vahid S (2021) A transductive transfer learning approach for image classification. Int J Mach Learn Cybernet 12(3):747–762
27. Long M, Wang J, Ding G, Sun J, Yu PS (2014) Transfer joint matching for unsupervised domain adaptation. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 1410–1417
28. Satpal S, Sarawagi S (2007) Domain adaptation of conditional probability models via feature subsetting. In: PKDD, vol 4702, pp 224–235. Springer
29. Pan SJ, Tsang IW, Kwok JT, Yang Q (2011) Domain adaptation via transfer component analysis. IEEE Trans Neural Netw 22(2):199–210
30. Si S, Dacheng T, Bo G (2010) Bregman divergence-based regularization for transfer subspace learning. IEEE Trans Knowl Data Eng 22(7):929–942
31. Tahmoresnezhad J, Hashemi S (2015) A generalized kernel-based random k-samplesets method for transfer learning. Iran J Sci Technol Trans Electr Eng 39:193–207
32. Duan L, Tsang IW, Xu D, Maybank SJ (2009) Domain transfer svm for video concept detection. In: CVPR 2009. IEEE conference on computer vision and pattern recognition
33. Lorenzo B, Mattia M (2010) Domain adaptation problems: a dasvm classification technique and a circular validation strategy. IEEE Trans Pattern Anal Mach Intell 32(5):770–787
34. Long M, Wang J, Ding G, Pan SJ, Philip SY (2014) Adaptation regularization: A general framework for transfer learning. IEEE Trans Knowl Data Eng 26(5):1076–1089
35. Shiva Noori Saray and Jafar Tahmoresnezhad (2021) Joint distinct subspace learning and unsupervised transfer classification for visual domain adaptation. SIViP 15(2):279–287
36. Marzieh Gheisari and Mahdieh Soleymani Baghshah (2015) Unsupervised domain adaptation via representation learning and adaptive classifier learning. Neurocomputing 165:300–311
37. Elahe G, Jafar T (2020) Joint discriminative subspace and distribution adaptation for unsupervised domain adaptation. Appl Intell 50(7):2050–2066
38. Vural Elif (2018) Generalization bounds for domain adaptation via domain transformations. In: 2018 IEEE 28th international workshop on machine learning for signal processing (MLSP), IEEE, pp 1–6
39. Ghifary M, Balduzzi D, Kleijn WB, Zhang M (2017) Scatter component analysis: a unified framework for domain adaptation and domain generalization. IEEE Trans Pattern Anal Mach Intell 1:1–1
40. Kouw WM, Van Der Maaten LJP, Krijthe JH, Loog M (2016) Feature-level domain adaptation. J Mach Learn Res 17(1):5943–5974
41. 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
42. Jafar T, Sattar H (2017) Visual domain adaptation via transfer feature learning. Knowl Inf Syst 50(2):585–605
43. Dinh CV, Duin RPW, Piqueras-Salazar I, Loog M (2013) Fidos: a generalized fisher based feature extraction method for domain shift. Pattern Recogn 46(9):2510–2518
44. Wenting Tu, Shiliang Sun (2012) Cross-domain representation-learning framework with combination of class-separate and domain-merge objectives. In *Proceedings of the 1st International Workshop on Cross Domain Knowledge Discovery in Web and Social Network Mining*, pages 18–25. ACM
45. Wang Z, Song Y, Zhang C (2008) Transferred dimensionality reduction. Mach Learn Knowl Discov Databases, pp 550–565
46. Borgwardt KM, Gretton A, Rasch MJ, Kriegel H-P, Schölkopf B, Smola AJ (2006) Integrating structured biological data by kernel maximum mean discrepancy. Bioinformatics 22(14):e49–e57
47. Yao Y, Doretto G (2010) Boosting for transfer learning with multiple sources. In: 2010 IEEE conference on computer vision and pattern recognition (CVPR), pp 1855–1862. IEEE
48. Yoshida Y, Hirao T, Iwata T, Nagata M, Matsumoto Y (2011) Transfer learning for multiple-domain sentiment analysis-identifying domain dependent/independent word polarity. In: *AAAI*
49. Moreno O, Shapira B, Rokach L, Shani G (2012) Talmud: transfer learning for multiple domains. In: Proceedings of the 21st ACM international conference on Information and knowledge management, ACM, pp 425–434
50. Li S, Zong C (2008) Multi-domain adaptation for sentiment classification: Using multiple classifier combining methods. In: International conference on natural language processing and knowledge engineering, 2008. NLP-KE'08, IEEE, pp 1–8
51. Rita C, Qian S, Wei F, Ian D, Sethuraman P, Jieping Y (2012) Multisource domain adaptation and its application to early detection of fatigue. ACM Trans Knowl Discov Data (TKDD) 6(4):18
52. Zhang Y, Cao B, Yeung D-Y (2010) Multi-domain collaborative filtering. In: Proceedings of the twenty-sixth conference on uncertainty in artificial intelligence, UAI'10, Arlington, Virginia, United States, AUAI Press, pp 725–732
53. Belkin M, Niyogi P, Sindhwani V (2006) Manifold regularization: a geometric framework for learning from labeled and unlabeled examples. J Mach Learn Res, 7(Nov):2399–2434

54. Ulrike VL (2007) A tutorial on spectral clustering. Stat Comput 17(4):395–416
55. Schölkopf B, Herbrich R, Smola A (2001) A generalized representer theorem. In: Computational learning theory, Springer, pp 416–426
56. Long M, Wang J, Sun J, Yu Philip S (2015) Domain invariant transfer kernel learning. IEEE Trans Knowl Data Eng 27(6):1519–1532
57. Griffin G, Holub A, Perona P 2007) Caltech-256 object category dataset
58. Herbert B, Tinne T, Luc VG (2006) Surf: speeded up robust features. Comput Vis ECCV 2006:404–417
59. Ming S, Dmitry K, Yun F (2014) Generalized transfer subspace learning through low-rank constraint. Int J Comput Vision 109(1–2):74–93
60. Yong X, Xiaozhao F, Jian W, Xuelong L, David Z (2016) Discriminative transfer subspace learning via low-rank and sparse representation. IEEE Trans Image Process 25(2):850–863

Mehri Mardani received her BS degree in information technology (IT) engineering from Foulad Institute of Technology, Isfahan, Iran, in 2015. She got her master of science degree in IT engineering from Urmia University of Technology, Urmia, Iran, in 2017.

Jafar Tahmoresnezhad received his Ph.D. degree in computer science from Shiraz University, Shiraz, Iran, in 2015. Following academic appointments at Urmia University of Technology, he is currently an associate professor at Faculty of IT and Computer Engineering, Urmia University of Technology, Urmia, Iran. His research interests include pattern recognition, transfer learning, deep learning, data mining and computer security.