# UNSUPERVISED DOMAIN ADAPTATION WITH JOINT SUPERVISED SPARSE CODING AND DISCRIMINATIVE REGULARIZATION TERM

*Lin Zhu*[1,2]*, Xiang Zhang*[1*]*,Wenju Zhang*[1]*,Xuhui Huang*[2]*, Naiyang Guan*[1]*, and Zhigang Luo*[1*]

[1]Institute of Software, College of Computer, National University of Defense Technology,
[2]Department of Computer Science and Technology, College of Computer
National University of Defense Technology,Changsha, Hunan, 410073, P.R. China
zhulin57626@163.com, zhangxiang_43@aliyun.com, zgluo@nudt.edu.cn

## ABSTRACT

Domain adaptation (DA) attempts to enhance the generalization capability of classifier through narrowing the gap of the distributions across domains. This paper focuses on unsupervised domain adaptation where labels are not available in target domain. Most existing approaches explore the domain-invariant features shared by domains but ignore the discriminative information of source domain. To address this issue, we propose a discriminative domain adaptation method (DDA) to reduce domain shift by seeking a common latent subspace jointly using supervised sparse coding (SSC) and discriminative regularization term. Particularly, DDA adapts SSC to yield discriminative coefficients of target data and further unites with discriminative regularization term to induce a common latent subspace across domains. We show that both strategies can boost the ability of transferring knowledge from source to target domain. Experiments on two real world datasets demonstrate the effectiveness of our proposed method over several existing state-of-the-art domain adaptation methods.

***Index Terms*—** Transfer learning, domain adaptation, sparse coding, subspace learning.

## 1. INTRODUCTION

Traditional machine learning methods favor the assumption that both the source and target distributions are identical. However, in practical applications, the source and target data might be from different distributions [1, 2]. Then re-collecting the source dataset is needed every time the distribution of the target varies [1]. The requirement for avoiding re-collecting source datasets encourages the development of domain adaptation [1, 3, 4].

Domain adaptation (DA) allows the difference between the source and target distributions. It attempts to eliminate the difference by transferring the knowledge of the easily found and labeled source domain to the unlabeled target domain. And a major assumption is respected that both domains share the latent space involving the latent factors [1, 2, 5, 6, 7, 8, 9]. Based on this assumption, various methods have been investigated, such as geodesic flow kernel(GFK) [10], regularized cross-domain metric transfer learning [11], and Bregman divergence-based regularization transfer learning [12].

Recently, advances in sparse coding have made itself a great significant candidate in domain adaptation. In domain adaptation, sparse representation aims to learn a subspace by selecting the proper atoms as few as possible in two manners. The first way is to assume that a shared dictionary is trained over both domains by transfer sparse coding (TSC [6]) and its supervised versions [2]. They seek the effective coefficients in an embedding fashion and thus confront the out-of-the-sample problem. For the second, both source and target domain data reconstruct each other in the common latent subspace. Zhang *et al.* [8] developed the latent sparse transfer domain learning method (LSDT) which learns the latent subspace by jointly sparse coding and subspace learning. However, they do not work well in classification tasks due to the limited discriminant power.

To address this issue, this paper proposes a discriminative domain adaptation method (DDA) which jointly combines supervised sparse coding(SSC) with discriminative subspace learning to boost the discriminant capacity of learned features. Firstly, DDA takes advantage of the source labels and is also tailored for the case where labels are still available in the target domain which cannot be guaranteed in several previous methods such as LSDT [8], TSC [6] and STSC [2]. Secondly, SSC instead of traditional sparse coding is adopted in DDA to make use of label information [13]. To the best of our knowledge, we are the first to introduce SSC [13] to learn more discriminative coding for domain adaptation. Thirdly, we boost the learned common subspace via a novel discriminative regularization term. In contrast with both LTSL [7] and LSDT [8], the proposed regularization term can simultaneously boost the discriminative power of both target repre-

*Corresponding author. Thanks to National High-tech R&D Program (No. 2015AA020108) and National Natural Science Foundation of China (No. U1435222) for funding.

sentation and the common latent subspace. This is based on the intuition that the representation elements of each datum vary from different classes. By this idea, in the learned common latent subspace, we reconstruct each target datum by the product of source data with the corresponding coefficients, and then encourage the distance among the reconstructions of different classes to be as far as possible. Lastly, our method can also be extended to a nonlinear model via the kernel trick. Experiments of image classification on two popular datasets including 4DA and DeCAF show that DDA outweighs the baseline methods in quantity.

## 2. RELATED WORK

In this paper, we focus on the situations where target data are unlabeled while source data are labeled, and tasks of both source and target are the same. Recently, plenty of domain adaptation methods have been proposed for various applications,such as multimedia applications [14, 15, 6], NLP problems [16] and so on. Gopalan *et al.* [17] proposed a method to learn intermediate representations across domains from Grassmann manifold. Pan *et al.* [18] proposed transfer component analysis (TCA) to minimize the maximum mean discrepancy (MMD [19]) across domains in an infinite-dimensional reproducing kernel Hilbert space (RKHS). Xu *et al.* [5] proposed discriminative transfer subspace learning to capture data structure by imposing joint low-rank and sparse constraints on the coding coefficients.

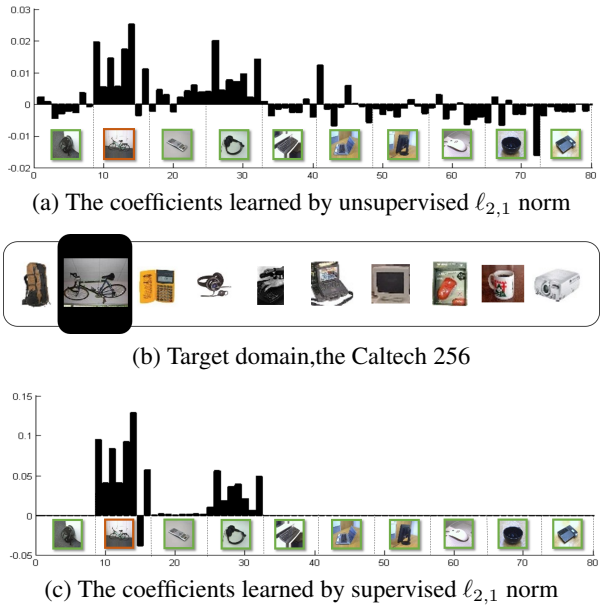
Feature representation, which tries to learn a good representation between source and target domains that minimizes domain divergence [1, 20], has been broadly utilized in transfer learning [5, 2, 6, 21, 7, 8]. Sparse coding as an effective feature representation method has been adopted in transfer learning. Raina *et al.* [21] applied sparse coding algorithm to learn a succinct higher-level representation. Al-Shedivat *et al.* [17] proposed STSC by combining multi-class SVM, sparse coding, MMD and Graph-Laplacian regularization term. Recently, Zhang *et al.* [8] proposed LSDT by sparsely reconstructing target data with all the data based on $\ell_1$ norm. However, these proposed methods ignore the label information of source data when learning the coefficients.

## 3. DISCRIMINATIVE DOMAIN ADAPTATION

This section jointly employs supervised sparse coding (SSC) and a discriminative term to yield discriminative coefficients that represent target domain data in the common latent subspace.

### 3.1. Problem Formulation

Most of existing domain adaptation methods based on sparse coding [5, 8, 9] assume that target data can be reconstructed by source data in a common latent subspace. Let



(a) The coefficients learned by unsupervised $\ell_{2,1}$ norm

(b) Target domain,the Caltech 256

(c) The coefficients learned by supervised $\ell_{2,1}$ norm

**Fig. 1**. Coefficients learned by unsupervised $\ell_{2,1}$-norm (a) and supervised $\ell_{2,1}$-norm (c), respectively. Their corresponding target is the "bike" image of (b). It implies that for each target datum the coding coefficients over source data from different classes can characterize its class.

source and target data denote by $X_S \in R^{d\times N_S}$ and $X_T \in R^{d\times N_T}$, respectively, where $N_S$ and $N_T$ respectively indicate the number of samples in source and target domains with the dimensionality $d$. The above assumption can be formulated as:

$$\begin{aligned} \min_{P,Z} \quad & \left\|P^T X_T - P^T X_S Z\right\|_F^2 + \tau\|Z\|_{2,1}, \\ s.t. \quad & P^T P = I \end{aligned} \tag{1}$$

where $P \in R^{d\times m}$ denotes the common latent subspace, and $Z$ indicates the coefficient matrix. Besides, $\|\cdot\|_F$ signifies the Frobenius norm of matrix, and $\|\cdot\|_{2,1}$ is used for better sparsity and robustness [22]. The objective (1) takes no account of the labels of source domain data. Inspired by [13], we restrict the coefficients from the same class to be similar while forcing the ones from different classes to be sparse through supervised $\ell_{2,1}$-norm. And each column of the coefficient matrix $Z = [Z_1, \ldots, Z_{N_T}]$ can be divided into $C$ parts, which means $Z_j = [Z_j^1, \ldots, Z_j^C]^T$, where $Z^i = [Z_1^i, \ldots, Z_{N_T}^i]$ means the representation of target data for the $i$-th class and $Z_j^i \in R^{n_i\times 1}$ means the representation of the $j$-th target data for the $i$-th class. Then, we can rewrite (1) as

$$\begin{aligned} \min_{P,Z} \quad & \left\|P^T X_T - P^T X_S Z\right\|_F^2 + \tau \sum_{j=1}^{N_T}\sum_{i=1}^{C} \left\|Z_j^i\right\|_2, \\ s.t. \quad & P^T P = I \end{aligned} \tag{2}$$

Figure 1 displays the differences of the learned coefficients by (1) and (2). The objective (2) can better capture the group-wise relationships among target domain data and indirectly polish up the learned subspace. To further yield the discriminative subspace, we develop a discriminative term to enforce the target domain reconstruction differences from different source classes to be maximized. In this case, source data can be grouped as $X_S = [X_S^1, \cdots, X_S^C]$, where $X_S^i \in R^{D \times n_i}$ means the samples of source data in class $i$. Thereafter, we can formulate the discriminative term as:

$$\max_{P,Z} \quad \sum_{i,j}^{C} \left\| P^T {X_S}^i Z^i - P^T {X_S}^j Z^j \right\|_F^2 \tag{3}$$

This term involves three novel aspects: 1) it can exert the effectiveness of the labels; 2) it still inherently encourages the representative coefficients of identical class to be relevant; 3) by maximizing (3), target domain will be prone to the reconstruction part of specific source class, meanwhile far away from that of the other source classes, which guarantees class-consistency from source domain to target domain. By combining (2) and (3), the problem formulation of discriminative domain adaptation (DDA) is written as follows:

$$\begin{aligned} \min_{P,Z} \quad & \left\| P^T X_T - P^T X_S Z \right\|_F^2 + \tau \sum_{j=1}^{N_T} \sum_{i=1}^{C} \left\| Z_j^i \right\|_2 \\ & - \gamma \sum_{i,j}^{C} \left\| P^T {X_S}^i Z^i - P^T {X_S}^j Z^j \right\|_F^2, \\ s.t., & P^T P = I \end{aligned} \tag{4}$$

We adapt the kernel trick to encode nonlinear structure of data. Assume that $P = \varphi(X)\Phi$, where $X = [X_S, X_T]$, and $X \in R^{d \times N}$. Besides, $\varphi$ and $\Phi$ are denoted by nonlinear function and transformation matrix in the kernel space, respectively. Substituting the form above into (4), the final objective function of DDA can be recast as follows:

$$\begin{aligned} \min_{\Phi,Z} \quad & \left\| \Phi^T K_T - \Phi^T K_S Z \right\|_F^2 + \tau \sum_{j=1}^{N_T} \sum_{i=1}^{C} \left\| Z_j^i \right\|_2 \\ & - \gamma \sum_{i,j}^{C} \left\| \Phi^T {K_S}^i Z^i - \Phi^T {K_S}^j Z^j \right\|_F^2 \\ s.t. \quad & \Phi^T K \Phi = I_M \end{aligned} \tag{5}$$

where $K = \varphi(X)^T \varphi(X)$, $K_S = \varphi(X)^T \varphi(X_S)$, and $K_T = \varphi(X)^T \varphi(X_T)$. Empirically, we can select the linear kernel to improve the efficiency of DDA especially when data dimensionality is dominant over the number of instances.

### 3.2. Optimization

The objective (5) is non-convex jointly with two variables. Local solutions are obtained by solving one variable with the rest variables fixed. Lagrange multipliers $\Lambda$ are introduced for the orthogonal constraint of (5). The optimization problem can be solved by alternately performing the following two steps:

**Algorithm 1** Optimization for DDA

**Input:** $X_S, X_T, Y, \gamma, \tau$

1: Compute $K_T = X^T X_T, K = X^T X, K_S = X^T X_S$
2: Initialize $\Phi$ with an randomly orthogonal matrix.
3: **while** not converge **do**
4: fix $\Phi$ , update $Z$ via (7).
5: fix $Z$ , update $\Phi$ by solving (9).
6: **end while**

**Output:** $\Phi$

• Update $Z$

With $\Phi$ fixed, updating $Z$ is equivalent to solving the following objective:

$$\begin{aligned} \min_{\Phi,Z} \quad & \left\| \Phi^T K_T - \Phi^T K_S Z \right\|_F^2 + \tau \sum_{j=1}^{N_T} \sum_{i=1}^{C} \left\| Z_j^i \right\|_2 \\ & - \gamma \sum_{i,j}^{C} \left\| \Phi^T {K_S}^i Z^i - \Phi^T {K_S}^j Z^j \right\|_F^2 \end{aligned} \tag{6}$$

Let $X_T^q$ be the $q$-th target datum and $Z_q$ the corresponding coefficient vector, respectively, and ${K_T}^q = X^T X_T^q$. By setting the partial derivatives of three terms of (6) with respect to $Z_q$ to zero, we obtain

$$Z_q = ({K_S}^T \Phi \Phi^T K_S - 2\gamma\Omega + \tau D)^{-1} ({K_S}^T \Phi \Phi^T {K_T}^q) \tag{7}$$

where $\Omega = CM - {K_S}^T \Phi \Phi^T K_S$,

$$M = \begin{bmatrix} K_1^T \Phi \Phi^T K_1 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & K_C^T \Phi \Phi^T K_C \end{bmatrix},$$

and

$$D = \begin{bmatrix} \frac{1}{2\left\| Z_q^1 \right\|_2} I_{n_1} & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \frac{1}{2\left\| Z_q^C \right\|_2} I_{n_c} \end{bmatrix}$$

Then, we can combine $Z_q$ to yield $Z$, where $q = 1, \ldots, N_T$.

• Update $\Phi$

Let $G_1 = K_T - K_S Z$ and $G_{i,j} = {K_S}^i Z^i - {K_S}^j Z^j$, where $\mathrm{i} = 1, ..., N_S; j = 1, ..., N_S$ . Then, $\Phi$ can be updated by solving the following optimization problem:

$$\begin{aligned} \min_{\Phi} \quad & \left\| \Phi^T G_1 \right\|_F^2 - \gamma \sum_{i,j}^{C} \left\| \Phi^T G_{i,j} \right\|_F^2 \\ s.t. \quad & \Phi^T K \Phi = I. \end{aligned} \tag{8}$$

Setting the derivative of (8) with respect to $\Phi$ to zero, solving $\Phi$ becomes a generalized eigenvalue problem,

$$K^{-1}(G_1 {G_1}^T - \gamma \sum_{i,j}^{C} (G_{i,j} {G_{i,j}}^T))\Phi = \Phi\Lambda. \tag{9}$$

Therefore, $\Phi$ consists of the eigenvectors corresponding to the smallest $m$ eigenvalues of $K^{-1}(G_1 {G_1}^T - \gamma \sum_{i,j}^{C} (G_{i,j} {G_{i,j}}^T))$.

**Table 1**. Average classification accuracy of SURF feature

| SURF | A→ C | A→ W | A→ D | C→ A | C→ W | C→ D | W→ A | W→ C | W→ D | D→ A | D→ C | D→ W |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| LTSL(LDA) +RLS | 22.4± 0.6 | 25.7± 0.9 | 25± 1.1 | 31.3± 0.8 | 28.8± 1.1 | 34.6± 1.2 | 32.7± 0.7 | 27.6± 0.8 | 55.5± 0.8 | 31.1± 0.5 | 27± 0.4 | 58.4± 0.8 |
| LTSL(LDA) +KNN | 17.6± 0.8 | 31.8± 1.1 | 29.5± 0.9 | 31.8± 0.8 | 40.5± 1.1 | 38.1± 1.4 | 34.8± 1 | 23.6± 0.8 | 49.4± 1.4 | 32.7± 0.9 | 23.6± 0.5 | 49.1± 1.2 |
| TSL_LRSR+RLS | 37.2± 0.4 | 33.2± 0.6 | 34.5± 0.8 | 34.9± 0.5 | 29.3± 1 | 33.4± 0.9 | 32.3± 0.5 | 29.4± 0.3 | 63.9± 0.9 | 31.9± 0.3 | 30.2± 0.2 | 72.5± 0.5 |
| TSL_LRSR+KNN | 36.9± 0.3 | 32.8± 0.6 | 33.6± 0.8 | 34.6± 0.4 | 28.9± 1 | 33.5± 0.9 | 32± 0.5 | 28.9± 0.3 | 63.4± 0.9 | 31.5± 0.4 | 29.9± 0.3 | 72.2± 0.5 |
| LSDT | 38.6± 0.3 | 37.6± 0.5 | 39.2± 0.5 | 38.4± 0.5 | 33.6± 1.2 | 39.3± 0.7 | 36.7± 0.3 | 32.8± 0.3 | 68± 0.7 | 35.3± 0.3 | 32.8± 0.2 | 76.7± 0.4 |
| DDA | **39.7± 0.3** | **38.3± 0.5** | **40.5± 0.6** | **39.5± 0.5** | **35.1± 1.3** | **40.6± 0.8** | **37.5± 0.3** | **33.4± 0.3** | **69.3± 0.8** | **36.1± 0.2** | **33.2± 0.2** | **78.2± 0.4** |

**Table 2**. Average classification accuracy of DeCAF feature of layer 6

| $F_6$ | A→ C | A→ W | A→ D | C→ A | C→ W | C→ D | W→ A | W→ C | W→ D | D→ A | D→ C | D→ W |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| LTSL(LDA) +RLS | 66.8± 1.2 | 68.6± 1.4 | 76.6± 1.7 | 86.9± 0.4 | 74.8± 1.5 | **85.1± 1.3** | **80.8± 0.9** | 70.4± 1.1 | 94.7± 0.6 | 81.5± 1.4 | 72.7± 1 | 93± 0.7 |
| LTSL(LDA) +KNN | 33.6± 2 | 72.5± 1.2 | 78.7± 1.6 | 78.3± 1.1 | **76.3± 1.1** | 84.6± 1.3 | 79.2± 1.1 | 52.9± 2 | 89.3± 1.2 | 76.7± 1.2 | 54.4± 1.8 | 81.8± 1.7 |
| TSL_LRSR+RLS | 82.7± 0.2 | 70.3± 0.5 | 80.3± 0.6 | 81.9± 0.7 | 62± 0.7 | 69.8± 1.1 | 62.9± 0.9 | 63.5± 0.6 | 93.7± 0.5 | 72.2± 0.8 | 70.5± 0.5 | 93± 0.4 |
| TSL_LRSR+KNN | 82.3± 0.3 | 69.4± 0.6 | 79.3± 0.5 | 81.5± 0.7 | 61.1± 0.7 | 68.5± 1.1 | 62± 0.8 | 63.1± 0.6 | 93.8± 0.5 | 71.9± 0.7 | 70.3± 0.5 | 93.7± 0.4 |
| LSDT | 84.1± 0.2 | 75.2± 0.6 | 84.9± 0.4 | 89.1± 0.4 | 73.9± 0.8 | 83.6± 0.9 | 76.1± 0.9 | 71.8± 0.4 | 96.9± 0.3 | 86.2± 0.3 | 79.4± 0.2 | **97.6± 0.3** |
| DDA | **84.5± 0.2** | **76.1± 0.7** | **85.5± 0.5** | **89.6± 0.3** | 75.1± 0.9 | 84.4± 1.1 | 76.5± 1.1 | **72.9± 0.4** | **97.4± 0.3** | **87.5± 0.3** | **80.3± 0.3** | **97.6± 0.3** |

**Table 3**. Average classification accuracy of DeCAF feature of layer 7

| $F_7$ | A→ C | A→ W | A→ D | C→ A | C→ W | C→ D | W→ A | W→ C | W→ D | D→ A | D→ C | D→ W |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| LTSL(LDA) +RLS | 69.6± 0.9 | 65.2± 1.1 | 73.7± 1.8 | 86.5± 0.7 | 75.8± 1.3 | 84.2± 1.3 | 78.8± 1.4 | 71.2± 1.6 | 94.6± 0.6 | 80.8± 1.3 | 71.1± 0.9 | 93.6± 0.5 |
| LTSL(LDA) +KNN | 43.7± 1.9 | 76± 1.3 | 80.7± 1.4 | 79.3± 1.1 | **79.7± 0.8** | 85.4± 1.5 | 82.4± 0.6 | 62.5± 1.4 | 91.9± 1 | 77.3± 1.2 | 60.6± 1.4 | 88.3± 1.2 |
| TSL_LRSR+RLS | 84.1± 0.2 | 73.8± 0.5 | 84.4± 0.4 | 82.5± 0.7 | 63.3± 0.9 | 73.4± 1 | 65.2± 0.9 | 67.1± 0.7 | 93.5± 0.7 | 75± 0.6 | 73.2± 0.4 | 93.1± 0.6 |
| TSL_LRSR+KNN | 83.1± 0.3 | 72.2± 0.6 | 82.6± 0.4 | 82.5± 0.7 | 62.5± 0.8 | 71.3± 1 | 64.2± 0.9 | 66.8± 0.7 | 93.6± 0.6 | 74.7± 0.6 | 72.7± 0.4 | 94.4± 0.4 |
| LSDT | **85.7± 0.1** | **77.9± 0.6** | 87.2± 0.3 | 90.2± 0.3 | 76.9± 0.8 | 85.3± 0.8 | 80.9± 0.6 | 76.9± 0.4 | 97.1± 0.4 | 88.4± 0.2 | 82± 0.2 | 97.1± 0.3 |
| DDA | **85.7± 0.2** | 77.7± 0.7 | **87.3± 0.4** | **90.8± 0.3** | 76.8±0.9 | **85.6± 0.8** | **81.6± 0.7** | **78.5± 0.4** | **97.4± 0.3** | **89.2± 0.2** | **83.2± 0.2** | **97.3± 0.3** |

The details of two steps are summarized into **Algorithm 1**. For its convergence, both steps yield local optimal solution to each subproblem and guarantee a monotonous decrease of the objective value. Empirically, DDA converges quickly within no more than 10 round iterations. Due to page limits, we omit the details of convergence analysis.

## 4. EXPERIMENTS

In this section, we validate the effectiveness of the proposed DDA method on two office object datasets. In our experiments, we compare DDA with LSDT [8], TSL_LRSR [5],and LTSL with LDA [7]. Following [5, 7], we also utilize both RLS and KNN classifier for TSL_LRSR and LTSL. For our DDA method, we set the trade-off parameters $\gamma$ and $\tau$ to 1, and adopt the linear kernel $K = X^T X$ to improve efficiency.

### A.4DA-SURF

The 4DA database consists of four datasets including A (Amazon), W (Webcam), D (DSLR) and C (Caltech-256), and ten common classes with 2533 images are selected. We strictly follow the experimental setting by [10], for source domain, if it is Amazon, 20 samples per class are selected; otherwise, eight samples per class are randomly chosen, while for target domain, three samples per class are selected for training and the rest are left for testing. There are 12 combinations of each two domains, and 20 random splits of data are used.

In this experiment, the SURF features are extracted and quantized into 800-bin histogram based on the coding technique are used.

**Table 1** shows the results of average object category accuracies and the corresponding standard deviations. Best results are in bold font. The results indicate that DDA is superior to existing three state-of-the-art methods. The performance gain can be attributed to supervised sparse coding and discriminative regularized term.

### B.4DA-DeCAF

The 4DA-DeCAF features are obtained from the outputs of a pre-trained CNN network which contains five convolutional and three fully-connected layers [8, 23, 24]. Here, we respectively employ the 6th and 7th layer outputs as the input features whose dimension size is $4096$. **Table 2** and **3** display the results of average object category accuracies and the corresponding standard deviations. The results show that DDA is also superior to the state-of-the-art baseline methods.

## 5. CONCLUSION

In this paper, we proposed a discriminative domain adaptation method (DDA) to boost the ability of transferring knowledge from source domain to target domain. DDA reduces the distribution mismatch between domains by joint supervised sparse coding and discriminative regularization term. Both strategies improve the generalization capability of classifier on a learned common latent subspace. More importantly, it is unsupervised and merely depends on the labels of source domain. The results of DDA have been shown on two real-world datasets.

## 6. REFERENCES


[1] SJ Pan and Q Yang, "A survey on transfer learning," *IEEE Transactions on Knowledge and Data Engineering*, vol. 22, no. 10, pp. 1345–1359, 2010.

[2] M Al-Shedivat, JJ Wang, M Alzahrani, JZ Huang, and X Gao, "Supervised transfer sparse coding," in *AAAI Conference on Artificial Intelligence*, 2014.

[3] B Kulis, K Saenko, and T Darrell, "What you saw is not what you get: Domain adaptation using asymmetric kernel transforms," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2011, pp. 1785–1792.

[4] K Weiss, TM Khoshgoftaar, and D Wang, "A survey of transfer learning," *Journal of Big Data*, vol. 3, no. 1, pp. 1–40, 2016.

[5] Y Xu, X Fang, J Wu, X Li, and D Zhang, "Discriminative transfer subspace learning via low-rank and sparse representation," *IEEE Transactions on Image Processing*, vol. 25, no. 2, pp. 850–863, 2016.

[6] M Long, G Ding, J Wang, J Sun, Y Guo, and PS Yu, "Transfer sparse coding for robust image representation," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2013, pp. 407–414.

[7] M Shao, D Kit, and Y Fu, "Generalized transfer subspace learning through low-rank constraint," *International Journal of Computer Vision*, vol. 109, no. 1-2, pp. 74–93, 2014.

[8] L Zhang, W Zuo, and D Zhang, "Lsdt: Latent sparse domain transfer learning for visual adaptation," *IEEE Transactions on Image Processing*, vol. 25, no. 3, pp. 1177–1191, 2016.

[9] L Zhang, SK Jha, T Liu, and G Pei, "Discriminative kernel transfer learning via $l_{2,1}$-norm minimization," in *International Joint Conference on Neural Networks*, 2016, pp. 2220–2227.

[10] B Gong, Y Shi, F Sha, and K Grauman, "Geodesic flow kernel for unsupervised domain adaptation," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2012, pp. 2066–2073.

[11] K Saenko, B Kulis, M Fritz, and T Darrell, "Adapting visual category models to new domains," in *European Conference on Computer Vision*, 2010, pp. 213–226.

[12] S Si, D Tao, and B Geng, "Bregman divergence-based regularization for transfer subspace learning," *IEEE Transactions on Knowledge and Data Engineering*, vol. 22, no. 7, pp. 929–942, 2010.

[13] J Huang, F Nie, H Huang, and CHQ Ding, "Supervised and projected sparse coding for image classification.," in *AAAI Conference on Artificial Intelligence*, 2013.

[14] J Yang, R Yan, and AG Hauptmann, "Cross-domain video concept detection using adaptive svms," in *ACM International Conference on Multimedia*, 2007, pp. 188–197.

[15] L Duan, IW Tsang, D Xu, and SJ Maybank, "Domain transfer svm for video concept detection," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2009, pp. 1375–1381.

[16] J Jiang and C Zhai, "Instance weighting for domain adaptation in nlp," in *Association for Computational Linguistics*, 2007, vol. 7, pp. 264–271.

[17] R Gopalan, R Li, and R Chellappa, "Domain adaptation for object recognition: An unsupervised approach," in *IEEE International Conference on Computer Vision*, 2011, pp. 999–1006.

[18] SJ Pan, IW Tsang, JT Kwok, and Q Yang, "Domain adaptation via transfer component analysis," *IEEE Transactions on Neural Networks*, vol. 22, no. 2, pp. 199–210, 2011.

[19] A Gretton, KM Borgwardt, M Rasch, B Schölkopf, AJ Smola, et al., "A kernel method for the two-sample-problem," *Advances in Neural Information Processing Systems*, vol. 19, pp. 513, 2007.

[20] S Ben-David, J Blitzer, K Crammer, F Pereira, et al., "Analysis of representations for domain adaptation," *Advances in Neural Information Processing Systems*, vol. 19, pp. 137, 2007.

[21] R Raina, A Battle, H Lee, B Packer, and AY Ng, "Self-taught learning: transfer learning from unlabeled data," in *International Conference on Machine Learning*, 2007, pp. 759–766.

[22] F Nie, H Huang, X Cai, and CH Ding, "Efficient and robust feature selection via joint $l_{2,1}$-norms minimization," in *Advances in Neural Information Processing Systems*, 2010, pp. 1813–1821.

[23] A Krizhevsky, I Sutskever, and GE Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in Neural Information Processing Systems*, 2012, pp. 1097–1105.

[24] J Donahue, Y Jia, O Vinyals, J Hoffman, N Zhang, E Tzeng, and T Darrell, "DeCAF: A Deep Convolutional Activation Feature for Generic Visual Recognition," in *International Conference on Machine Learning*, 2014, vol. 32, pp. 647–655.