Class-specific Reconstruction Transfer Learning via Sparse Low-rank Constraint

Shanshan Wang[1], Lei Zhang[1]*, and Wangmeng Zuo[2]
[1]College of Communication Engineering, Chongqing University, Chongqing, China
[2]School of Computer Science and Technology, Harbin Institute of Technology, Harbin, China
wangshanshan@cqu.edu.cn, leizhang@cqu.edu.cn, wmzuo@hit.edu.cn

Abstract

*Subspace learning and reconstruction have been widely explored in recent transfer learning work and generally a specially designed projection and reconstruction transfer matrix are wanted. However, existing subspace reconstruction based algorithms neglect the class prior such that the learned transfer function is biased, especially when data scarcity of some class is encountered. Different from those previous methods, in this paper, we propose a novel reconstruction-based transfer learning method called **C**lass-specific **R**econstruction **T**ransfer **L**earning (CRTL), which optimizes a well-designed transfer loss function without class bias. Using a class-specific reconstruction matrix to align the source domain with the target domain which provides help for classification with class prior modeling. Furthermore, to keep the intrinsic relationship between data and labels after feature augmentation, a projected Hilbert-Schmidt Independence Criterion (pHSIC), that measures the dependency between two sets, is first proposed by mapping the data from original space to RKHS in transfer learning. In addition, combining low-rank and sparse constraints on the class-specific reconstruction coefficient matrix, the global and local data structures can be effectively preserved. Extensive experiments demonstrate that the proposed method outperforms conventional representation-based domain adaptation methods.*

1. Introduction

Image classification methods aim to build a classification model from training samples and then apply it to classify test samples. Generally, with the fundamental assumption of machine learning, the fixed model can work well if the test samples are with similar distribution of the training samples [23]. Yet, in real world, it is impossible to guarantee that those samples of similar semantics have the same data distribution. Various sampling associated factors can lead to different distributions such as resolutions, illuminations, background, etc. Therefore, conventional methods fail in handling such issues that the basic assumption of similar distribution the data should hold in machine learning is violated. Fig.1 shows some images whose distributions are different. In this case, if the images in the first set are used to train a classifier model, the model cannot work well when classifying other images.

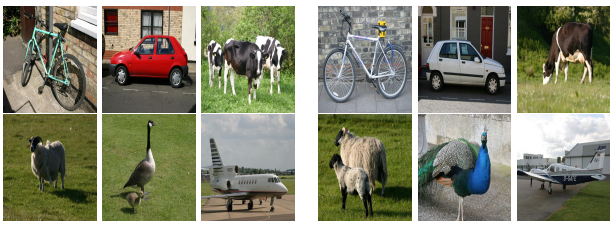

Figure 1: Different distributions from different domain subjects

To solve the problem, one straightforward method is to collect a large amount of labeled source data that have the same distribution with test data and use them to retrain the model, that is $data\text{-}driven$. However, collecting and labeling sufficient data is tedious and difficult, which consumes a lot of labor costs. The other effective method is transfer learning by leveraging a number of data from target domain, that is $model\text{-}driven$. Transfer learning tends to transfer the knowledge from source domain to target domain by exploiting their structural and similar high-level semantic relationship. In that case, one can use distribution different yet relevant data to enhance the classification performance. While combining deep learning and transfer learning is another quite effective method, but it is not the focus of this paper. Generally, when facing with classification tasks in a given source domain, transfer learning is often proposed to leverage the prior knowledge in other different but related $domain\ data$, which is also referred as $domain\ adaptation$ [13, 24, 26, 29, 30, 35]. The domain data generally share the same task yet different distributions [34]. In order to address the problem of different distributions, previous work on domain adaptation and transfer learning have made great contributions. The methods can

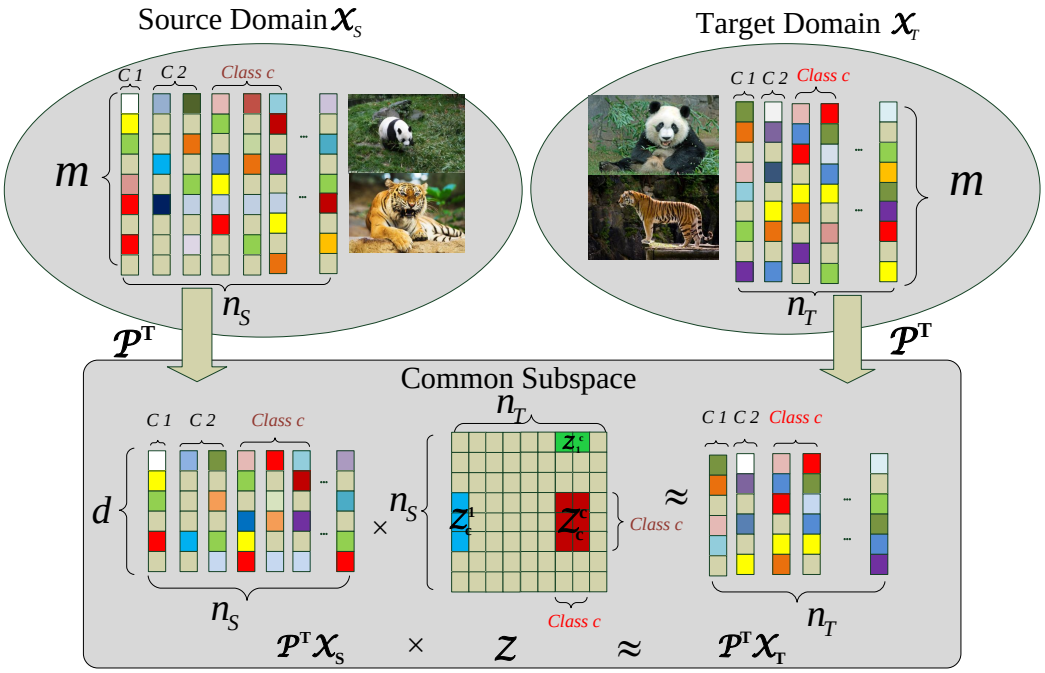

Figure 2: Illustration of our proposed Class-specific Reconstruction Transfer Learning (CRTL)

be divided as two categories: (1) methods of changing the data representation; (2) methods of modifying the trained classifier [31].

In this paper, we focus on the former, i.e. a class-specific reconstruction transfer learning model is proposed. The proposed method aims at constructing a class-specific and statistical dependence preserved model across domains in reproducing kernel Hilbert space (RKHS), such that the projected feature distribution can be aligned between domains. The idea of the proposed CRTL method is described in Fig.2, where the correspondence matrix $\mathcal{Z}$ is class-specific.

Maximum Mean Discrepancy (MMD) [11], acts as a discrepancy metric across domains, has been used in many unsupervised domain adaptation methods. The information of categories is neglected. Inspired by classifier adaptation, to enhance the correlation between the projected feature and labels, a statistical method that can describe such intrinsic relationship regardless of the modeling is wanted. Hilbert-Schmidt Independence Criterion (HSIC) [12] proposed based on Hilbert-Schmidt norm in RKHS was used to measure the dependency between two sets. Therefore, in this paper, the HSIC, instead of MMD is modeled for domain adaptation, by projecting the data from original space $\mathcal{R}^{\mathcal{D}}$ to RKHS $\mathcal{H}$, that is defined as $\varphi : \mathcal{R}^{\mathcal{D}} \to \mathcal{H}$.

Due to the domain difference between the source and target domain, a latent projection [8] was wanted for projecting the source and target data into a common subspace. However, learning a common subspace without data correspondence across domains constrained the domain transfer performance. To this end, we propose a joint learning method for pursuit of latent subspace $\mathbf{P}$ and also a reconstruction (correspondence) matrix $\mathcal{Z}$ [8], simultaneously.

In order to obtain a better subspace, many methods [27, 31] attempt to make $\mathbf{P}$ discriminative by constructing some regularizer and discriminative constraints, rather than considering the class-specific characteristic of the reconstruction matrix $\mathcal{Z}$ [33]. These methods generally ignore class prior distributions [14, 27, 31, 37] that is beneficial to construct a well-designed reconstruction transfer loss function without class bias. Different from those methods, we have an idea to make $\mathcal{Z}$ class-specific, such that the low-rank and sparsity constraints can be relaxed. For example, the sparsity constraint on $\mathcal{Z}$ expects that the source data of class c can reconstruct the target data of the same class (e.g. collaborative representation). For reducing the burden of the regularizer during learning, a class-specific reconstruction transfer loss function is constructed, such that the learnt transfer matrix is more structural and analytic. Essentially, when labeled data is arrayed by categories, the correspondence matrix $\mathcal{Z}$ shows a intrinsic block-diagonal structure [6, 20].

Low-rank representation (LRR) [19] was suggested to get the block diagonal solution for subspace segmentation. Different from LRR, sparse subspace clustering (SSC) [5] was suggested for data points that lie in a union of low-dimensional subspaces, which not only handles the data points near the intersections of subspaces, but also avoids the trivial solution. Due to the benefits of both regulariza-

tion constraints, LRR and SSC based constraints have been exploited in the model for capturing the global and local structures during domain correspondence.

In summary, the main contribution and novelty of this work are threefold:

- To keep the intrinsic relationship between domain data and labels, Hilbert-Schmidt Independence Criterion (HSIC) instead of the unsupervised MMD criterion is modeled to measure the dependence in reproducing kernel Hilbert space (RKHS). Also, a projected HSIC (pHSIC) is proposed for feature augmentation.

- In order to model the class prior distributions rather than domain distribution, a class-specific reconstruction transfer loss function and a pHSIC-based common subspace learning are proposed, which can be jointly learned for class associated and structural subspace transfer. In this case, the pressure of regularizer for structural $\boldsymbol{\mathcal{Z}}$ is significantly relaxed.

- Using both LRR and SSC regularization constraints, the global and local structures are effectively preserved with better block diagonal characteristic and robustness to outliers.

The rest of paper is organized as follows. In Section 2 we review the related work in domain adaptation. In Section 3 we present the proposed CRTL method and optimization. In Section 4 the experimental results by using benchmark datasets are presented. In Section 5 we discuss the proposed model, and finally Section 6 concludes this paper.

2. Preliminaries and Related Work

In recent years, a number of transfer learning methods have been proposed and it can be summarized as two categories: methods of classifier adaptation and methods of feature adaptation. For the former, one representative method called ASVM proposed by Yang et al. [16] tends to learn the perturbation term for adapting the source classifier to the target classifier. Xue et al. [32] proposed a method exploiting the common knowledge to share model parameters across domains based on dirichlet process prior. Zhang et al. [21] proposed a domain adaptation ELM method for classifier adaptation, and also a robust extreme domain adaptation method [36] by using Laplacian graph regularization for local structure preservation and shared domain classifier. Duan et al. [4] proposed an adaptive multiple kernel learning (AMKL) to recognize consumer from annotated web videos. Since it is impossible to eliminate the domain disparity between the source and target domain by using classifier adaptation, for the latter, the feature adaptation methods were proposed for domain disparity elimination. Subspace projection and coding is a appropriate way to achieve the goal, and the classifier trained by the projected source data is also adaptive to the projected target data. Shekhar et al. [28] proposed a shared domain dictionary learning (SDDL) method, which assumes that one joint dictionary can be learned for both domains, and representation based classifier was considered. Xu et al. [31] proposed a discriminative transfer subspace learning via low-rank and sparse representation by jointly learning the reconstruction and classifier. This paper is closely related with our work. However, the method is modeled on the whole source data distribution and the class prior distributions are not considered. Also, the subspace $\mathbf{P}$ is regularized by using a least square based classifier and the statistical dependence across domains that HSIC shows is not considered. Zhang et al. [37] proposed a latent sparse domain transfer (LSDT) method for visual adaptation by using both the source and target data for latent subspace and domain correspondence. This paper is also closely related with our work, but still does not include the merits of the proposed CRTL method. Shao et al. [27]proposed a LTSL method for reconstruction transfer based on low-rank constraint, in which the subspace and reconstruction matrix are learnt separately, and the near-optimal transfer is achieved. Gong et al. [9] proposed a GFK method by using geodesic flow kernel to modeling domain shift. But it is quite different from our method that it is an unsupervised domain adaptation method.

2.1. HSIC Criterion

Hilbert-Schmidt Independence Criterion [12] is an independence criterion based on the eigenspectrum of cross-covariance operators in reproducing kernel Hilbert space. HSIC is used to measure the dependency between two sets $\boldsymbol{\mathcal{X}}$ and $\boldsymbol{\mathcal{Y}}$. Let k_x and k_y denote the kernel function with respect to the RKHS $\boldsymbol{\mathcal{F}}$ and $\boldsymbol{\mathcal{G}}$. According to [12], HSIC independence Criterion is shown in Equation (1).

$$\begin{aligned} &\mathbf{HSIC}(\boldsymbol{\mathcal{X}}, \boldsymbol{\mathcal{Y}}, \boldsymbol{\mathcal{F}}, \boldsymbol{\mathcal{G}}) \\ &= \| C_{\mathcal{XY}} \|^2_{HS} = (N-1)^{-2} Tr(\boldsymbol{\mathcal{K}}_{\boldsymbol{\mathcal{X}}} \mathbf{H} \boldsymbol{\mathcal{K}}_{\boldsymbol{\mathcal{Y}}} \mathbf{H}) \\ &s.t. \mathbf{H} = \mathbf{I} - N^{-1} \mathbf{1}_{N\times 1} \mathbf{1}^T_{N\times 1} \end{aligned} \quad (1)$$

where N denotes the size of set $\mathcal{X}$ and $\mathcal{Y}$, $\| C_{\mathcal{XY}} \|^2_{HS}$ is Hilbert-Schmidt norm of the cross-covariance operator. $\boldsymbol{\mathcal{K}}_{\boldsymbol{\mathcal{X}}}$ and $\boldsymbol{\mathcal{K}}_{\boldsymbol{\mathcal{Y}}}$ are two kernel Gram matrix w.r.t $\mathcal{F}$ and $\mathcal{G}$ in RKHS, respectively. $\mathbf{H}$ is the centering matrix. With characteristic kernels k_x and k_y, it can be proved that the value of HSIC is zero if and only if $\boldsymbol{\mathcal{X}}$ and $\boldsymbol{\mathcal{Y}}$ are independent [12].

HSIC consists of an empirical estimation of the Hilbert-Schmidt norm of the cross-covariance operator and it has remarkable simplicity advantage compared with previous kernel-based independence criteria. Also, HSIC do not suffer from slow learning rate. In this paper, a projected HSIC criterion is proposed for improving the dependency between enhanced features and labels during transfer.

2.2. Sparse plus Low-rank based Transfer

The traditional methods have difficulty in capturing the intrinsic structures of data owning to the different distributions such as the local and global structure [31]. As mentioned before, low-rank representation is advantageous in getting the block diagonal solution for subspace segmentation, so that the global structure can be preserved. Different from LRR, sparse constraint can make relevant samples from different domains more interlace than irrelevant samples, and the local structure of data is thus preserved. When the training data is used for building the model, the noisy data included in the training data are treated equally which is harmful to transfer model. With the sparse coding, the outliers from the source data can be selectively treated in reconstructing the target data and avoid the trivial solution.

2.3. Deep Transfer Learning

Deep learning, as a data-driven transfer learning method, has witnessed a great achievements in many fields. However, when solving domain data problems by using deep learning technology, massive labeled training data are required. The data amount is increased with the increase of convolutional neural network (CNN) parameters [2]. For the tasks of small data, deep learning may not work well. Constructing joint data-driven and model-driven deep transfer learning is an effective way to face with domain data challenge. The number of required data is not as much as deep learning by exploiting transfer learning method [22]. On one hand, a network with fewer parameters and smaller structure can be re-trained. On the other hand, a large number of data always cause overfitting, while transfer learning allows the model to *see* different domain data.

Following experiments in 4DA dataset with fine-tuned CNN features prove that the proposed CRTL method is also useful for deep feature adaptation.

3. Class-specific Reconstruction Transfer

3.1. Notations

In this paper, the source and target domain are defined by subscript S and T. The training set of source and target domain is defined as $\boldsymbol{\mathcal{X}_S} \in \mathcal{R}^{m\times n_S}$ and $\boldsymbol{\mathcal{X}_T} \in \mathcal{R}^{m\times n_T}$, where m denotes dimension of data, n_S and n_T denote the number of samples in source and target domain, respectively. Let $\mathbf{P} \in \mathcal{R}^{m\times d}(m \geq d)$ be the discriminative basis transformation that maps the original data space dimension m of source and target domain into subspace dimension d respectively. $\boldsymbol{\mathcal{Z}} \in \mathcal{R}^{n_S\times n_T}$ represents reconstruction coefficient matrix, and $\mathbf{I}$ denotes the identity matrix. $\| \bullet \|_p$ and $\| \bullet \|_F$ denote l_p -norm and Frobenius norm respectively, $\| \bullet \|_*$ denotes nuclear norm. The superscript $\mathbf{T}$ denotes the transpose operator, and $\mathbf{Tr}(\bullet)$ denotes the trace operator of matrix.

In RKHS, the transformation $\boldsymbol{\mathcal{P}}$ is used instead of $\mathbf{P}$ in raw space. The kernel Gram matrix $\boldsymbol{\mathcal{K}}$ is defined as $\left[\boldsymbol{\mathcal{K}}\right]_{i,j} =< \varphi(\mathbf{x}_i), \varphi(\mathbf{x}_j) >_{\mathcal{H}} = \varphi(\mathbf{x}_i)^H\varphi(\mathbf{x}_j) = k(\mathbf{x}_i, \mathbf{x}_j)$, where k is a kernel function.

3.2. Projected HSIC

What we expect is that after projection, the intrinsic contact between the augmented features and labels can be well preserved for better dependency. Thus, the observations $Z_H := \{(x_1, l_1) \dots (x_n, l_n)\}$ can be used to construct Hilbert-Schmidt Independency Criterion after feature augmentation. Specifically, we call the HSIC with feature augmentation as projected HSIC (pHSIC), which is constructed with the same principle but different from HSIC, a projection $\boldsymbol{\mathcal{P}}$ is integrated for knowledge transfer. As described in [12], the proposed pHSIC can be formulated as

$$\begin{aligned}&\mathbf{pHSIC}(Z_H, \boldsymbol{\mathcal{F}}, \boldsymbol{\mathcal{G}}) = (N-1)^{-2}Tr(\boldsymbol{\mathcal{K}}\mathbf{H}\boldsymbol{\mathcal{L}}\mathbf{H})\\ &= (N-1)^{-2}\mathbf{Tr}(k(\boldsymbol{\mathcal{P}}^T\varphi[\mathbf{X}_S, \mathbf{X}_T], \boldsymbol{\mathcal{P}}^T\varphi[\mathbf{X}_S, \mathbf{X}_T])\mathbf{H}\boldsymbol{\mathcal{L}}\mathbf{H})\\ &s.t. \mathbf{H} = \mathbf{I} - N^{-1}\mathbf{1}_{N\times 1}\mathbf{1}_{N\times 1}^T\end{aligned} \tag{2}$$

where $\boldsymbol{\mathcal{K}}, \boldsymbol{\mathcal{L}} \in \mathcal{R}^{N\times N}$, $\boldsymbol{\mathcal{K}}_{i,j} = k(x'_i, x'_j)$, $\boldsymbol{\mathcal{L}}_{i,j} = l(y_i, y_j)$, $\mathbf{H}_{i,j} = \delta_{i,j} - N^{-1}$. $k(\bullet)$ and $l(\bullet)$ denote kernel functions. $\boldsymbol{\mathcal{K}} = k(\boldsymbol{\mathcal{X}'}.\boldsymbol{\mathcal{X}'})$, $\boldsymbol{\mathcal{X}'} = [\boldsymbol{\mathcal{X}'_S}, \boldsymbol{\mathcal{X}'_T}]$ denote the projected data, $\boldsymbol{\mathcal{L}} = l(\boldsymbol{\mathcal{Y}}, \boldsymbol{\mathcal{Y}})$. $\boldsymbol{\mathcal{Y}}$ denotes the data labels. $\mathbf{H}$ is a centering matrix. φ is a nonlinear function for feature augmentation, which maps the data from original space $\mathcal{R}^{\mathcal{D}}$ to RKHS $\mathcal{H}$ as $\varphi : \mathcal{R}^{\mathcal{D}} \to \mathcal{H}$. By using kernel trick, the nonlinear function φ does not need to be explicit, which will be presented in section 3.4. Thus, by maximizing $\mathbf{pHSIC}$, the dependency between features and labels can be well improved for domain classification.

3.3. Transfer Loss Function

As described in Fig.2, a better reconstruction matrix $\boldsymbol{\mathcal{Z}}$ under the discriminative subspace $\boldsymbol{\mathcal{P}}$ is expected. By leveraging class prior, we wish to learn a structural and class-specific reconstruction matrix $\boldsymbol{\mathcal{Z}}$, instead of learning a shared reconstruction matrix $\boldsymbol{\mathcal{Z}}$ based on the whole data distribution of all classes. Specifically, we wish that the data of class i in target domain can only be represented by the data of the same class in source domain using $\boldsymbol{\mathcal{Z}}_i^i$, so that a more structured reconstruction matrix $\boldsymbol{\mathcal{Z}}$ can be obtained. Thus, the transfer loss between source domain and target domain with respect to each class can be constructed.

For all the labeled $\boldsymbol{\mathcal{X}}_T$ and $\boldsymbol{\mathcal{X}}_S$, we prospect that after projection $\boldsymbol{\mathcal{P}}$ and nonlinear φ mapping, the source and target data of the same classes can be closely linked between $\boldsymbol{\mathcal{X}'}_T$ and $\boldsymbol{\mathcal{X}'}_S$, while the the data of different classes have no connections at all. Therefore, the data of class c in target domain is expressed by the data of the same class

in source domain via minimizing the reconstruction error $\beta_1 \parallel \boldsymbol{\mathcal{P}}^T\varphi(\boldsymbol{\mathcal{X}}_T^c) - \boldsymbol{\mathcal{P}}^T\varphi(\boldsymbol{\mathcal{X}}_S^c)\boldsymbol{\mathcal{Z}}_c^c \parallel_F^2$, where $\boldsymbol{\mathcal{X}}_T^c$ represents the target data of class c, $\boldsymbol{\mathcal{X}}_S^c$ represents the source data of class c, and $\boldsymbol{\mathcal{Z}}_c^c$ represents the class-specific representation coefficient with respect to class c. Furthermore, for avoiding the impact during transfer from the data of other classes, we also consider to minimize the representation between classes. That is, we wish the target data of class c cannot be expressed by the source data of class k (excluding class c). Therefore, this item can be constructed as $\sum_{k=1,k\neq c} \beta_2 \parallel \boldsymbol{\mathcal{P}}^T\varphi(\boldsymbol{\mathcal{X}}_S^c)\boldsymbol{\mathcal{Z}}_k^c \parallel_F^2$, where $\boldsymbol{\mathcal{Z}}_k^c$ represents the reconstruction coefficient from the source data of class c to the target data of class k. With the above analysis, the transfer loss function can be formulated as follows.

$$\begin{aligned} &E(\boldsymbol{\mathcal{X}}_S, \boldsymbol{\mathcal{X}}_T, \boldsymbol{\mathcal{P}}, \boldsymbol{\mathcal{Z}}) \\ &= \sum_{c=1}^{C}(\beta_1 \parallel \boldsymbol{\mathcal{P}}^T\varphi(\boldsymbol{\mathcal{X}}_T^c) - \boldsymbol{\mathcal{P}}^T\varphi(\boldsymbol{\mathcal{X}}_S^c)\boldsymbol{\mathcal{Z}}_c^c \parallel_F^2) \\ &\quad + \sum_{c=1}^{C}\sum_{k=1,k\neq c}^{C} \beta_2 \parallel \boldsymbol{\mathcal{P}}^T\varphi(\boldsymbol{\mathcal{X}}_S^c)\boldsymbol{\mathcal{Z}}_k^c \parallel_F^2 \end{aligned} \tag{3}$$

3.4. Model Formulation

As mentioned in [31, 37], the sparsity constraint is helpful to preserve the local structure of data such that each target sample can be well reconstructed by a few very associated samples from the source domain. Furthermore, the SSC method can account for the noise in data corruption and remove outliers with their intrinsic relatedness preserved. In addition, SSC ensures that the data from different domains can be well interlaced and significantly reduce the disparity of the domain distributions. Different from sparsity constraint, low-rank constraint is helpful to preserve the global structure of data, and it is advantageous to reveal a block-diagonal structure. In constructing the reconstruction matrix $\boldsymbol{\mathcal{Z}}$ in this paper, a joint sparse and low-rank regularizer is used to better account for the local and global characteristics simultaneously. Eventually, we have chosen sparse+low-rank constraints on the reconstruction matrix $\boldsymbol{\mathcal{Z}}$. By considering the HSIC and loss function, the objective function of the proposed CRTL method can be formulated as follows.

$$\begin{aligned} &\min_{\boldsymbol{\mathcal{P}},\boldsymbol{\mathcal{Z}}} E(\boldsymbol{\mathcal{X}}_S, \boldsymbol{\mathcal{X}}_T, \boldsymbol{\mathcal{P}}, \boldsymbol{\mathcal{Z}}) - \mathbf{pHSIC}(Z_H, \boldsymbol{\mathcal{X}}, \boldsymbol{\mathcal{L}}) \\ &+ \lambda_1 \parallel \boldsymbol{\mathcal{Z}} \parallel_* + \lambda_2 \parallel \boldsymbol{\mathcal{Z}} \parallel_1 \\ &s.t. \boldsymbol{\mathcal{P}}^{\mathbf{T}}\boldsymbol{\mathcal{P}} = \mathbf{I}, \lambda_i > 0, i = 1, 2 \end{aligned} \tag{4}$$

Further, by combining the pHSIC criterion in Equation (2) and the transfer loss function in Equation (3), the model (4) can be rewritten as:

$$\begin{aligned} &\min_{\boldsymbol{\mathcal{P}},\boldsymbol{\mathcal{Z}}} \sum_{c=1}^{C}(\beta_1 \parallel \boldsymbol{\mathcal{P}}^T\varphi(\mathbf{X}_T^c) - \boldsymbol{\mathcal{P}}^T\varphi(\boldsymbol{\mathcal{X}}_S^c)\boldsymbol{\mathcal{Z}}_c^c \parallel_F^2) \\ &\quad + \sum_{c=1}^{C}\sum_{k=1,k\neq c}^{C} \beta_2 \parallel \boldsymbol{\mathcal{P}}^T\varphi(\boldsymbol{\mathcal{X}}_S^c)\boldsymbol{\mathcal{Z}}_k^c \parallel_F^2 \\ &\quad - \frac{1}{(N-1)^2}\mathbf{Tr}(k(\boldsymbol{\mathcal{P}}^T\varphi(\boldsymbol{\mathcal{X}}), \boldsymbol{\mathcal{P}}^T\varphi(\boldsymbol{\mathcal{X}}))\mathbf{H}\boldsymbol{\mathcal{L}}\mathbf{H}) \\ &\quad + \lambda_1 \parallel \boldsymbol{\mathcal{Z}} \parallel_* + \lambda_2 \parallel \boldsymbol{\mathcal{Z}} \parallel_1 \\ &\quad s.t. \boldsymbol{\mathcal{P}}^{\mathbf{T}}\boldsymbol{\mathcal{P}} = \mathbf{I}, \boldsymbol{\mathcal{X}} = [\boldsymbol{\mathcal{X}}_S, \boldsymbol{\mathcal{X}}_T], \mathbf{1}^{1\times n_S}\boldsymbol{\mathcal{Z}} = \mathbf{1}^{1\times n_T} \end{aligned} \tag{5}$$

Generally, the optimal mapping $\boldsymbol{\mathcal{P}}^*$ can be represented as $(\boldsymbol{\mathcal{P}}^*)^T = \boldsymbol{\Phi}^T\varphi(\boldsymbol{\mathcal{X}})^T$, that is, the projection $\boldsymbol{\mathcal{P}}$ is formulated by linearly representing the data $\varphi(\boldsymbol{\mathcal{X}})$ by using a matrix $\boldsymbol{\Phi}$. Therefore, based on kernel trick, the objective function (5) can reformulated as

$$\begin{aligned} &\min_{\boldsymbol{\Phi},\boldsymbol{\mathcal{Z}}} \sum_{c=1}^{C}(\beta_1 \parallel \boldsymbol{\Phi}^T\boldsymbol{\mathcal{K}}_T^c - \boldsymbol{\Phi}^T\boldsymbol{\mathcal{K}}_S^c\boldsymbol{\mathcal{Z}}_c^c \parallel_F^2) \\ &\quad + \sum_{c=1}^{C}\sum_{k=1,k\neq c}^{C} \beta_2 \parallel \boldsymbol{\Phi}^T\boldsymbol{\mathcal{K}}_S^c\boldsymbol{\mathcal{Z}}_k^c \parallel_F^2 \\ &\quad - \frac{1}{(N-1)^2}\mathbf{Tr}(\boldsymbol{\Phi}^T\boldsymbol{\mathcal{K}}\mathbf{H}\boldsymbol{\mathcal{L}}\mathbf{H}\boldsymbol{\mathcal{K}}\boldsymbol{\Phi}) \\ &\quad + \lambda_1 \parallel \boldsymbol{\mathcal{Z}} \parallel_* + \lambda_2 \parallel \boldsymbol{\mathcal{Z}} \parallel_1 \\ &\quad s.t. \boldsymbol{\Phi}^T\boldsymbol{\mathcal{K}}\boldsymbol{\Phi} = \mathbf{I}, \mathbf{1}^{1\times n_S}\boldsymbol{\mathcal{Z}} = \mathbf{1}^{1\times n_T} \end{aligned} \tag{6}$$

3.5. Optimization

Although it seems that three variables are involved in model (6), both $\boldsymbol{\mathcal{Z}}_c^c$ and $\boldsymbol{\mathcal{Z}}_k^c$ can be expressed by $\boldsymbol{\mathcal{Z}}$ using some easily designed constant matrix. Therefore, two variables $\boldsymbol{\Phi}$ and $\boldsymbol{\mathcal{Z}}$ are involved in (6). To solve the problem, we use alternating optimization strategy, i.e. solving one variable while fixing the other one is considered. With two updating steps for $\boldsymbol{\Phi}$ and $\boldsymbol{\mathcal{Z}}$, the complete optimization of the proposed method is illustrated in Algorithm 1.

3.6. Classification

For classification, the projected source data and target data can be represented as $\boldsymbol{\mathcal{X}}_S{}' = \boldsymbol{\Phi}^T\varphi(\boldsymbol{\mathcal{X}})^T\varphi(\boldsymbol{\mathcal{X}}_S)$ and $\boldsymbol{\mathcal{X}}_T{}' = \boldsymbol{\Phi}^T\varphi(\boldsymbol{\mathcal{X}})^T\varphi(\boldsymbol{\mathcal{X}}_T)$. Then general classifiers (e.g. SVM, least square method, SRC) can be used based on the augmented training data $[\boldsymbol{\mathcal{X}}_S{}', \boldsymbol{\mathcal{X}}_T{}']$ with label $\boldsymbol{\mathcal{Y}} = [\boldsymbol{\mathcal{Y}}_S, \boldsymbol{\mathcal{Y}}_T]$. Notably, for the COIL-20 experiment, in order to keep the same experiment setting with DTSL [31], the classifier is trained only on $\boldsymbol{\mathcal{X}}_S{}'$ with label $\boldsymbol{\mathcal{Y}}_S$. Finally, the predicted labels of unlabeled target test data $\boldsymbol{\mathcal{X}}_{Tu}{}' = \boldsymbol{\Phi}^T\varphi(\boldsymbol{\mathcal{X}})^T\varphi(\boldsymbol{\mathcal{X}}_{Tu})$ are obtained accordingly.

Algorithm 1 The Proposed CRTL

Input: $\boldsymbol{\mathcal{X}}_S \in \mathcal{R}^{m\times n_S}$, $\boldsymbol{\mathcal{X}}_T \in \mathcal{R}^{m\times n_T}$, $\boldsymbol{\mathcal{Y}}_S \in \mathcal{R}^{n_S\times 1}$, $\boldsymbol{\mathcal{Y}}_T \in \mathcal{R}^{n_T\times 1}$, $\beta_1, \beta_2, \lambda_1, \lambda_2$

Procedure:

1. Compute $\boldsymbol{\mathcal{K}}_T = \varphi(\boldsymbol{\mathcal{X}})^T\varphi(\boldsymbol{\mathcal{X}}_T)$, $\boldsymbol{\mathcal{K}}_S = \varphi(\boldsymbol{\mathcal{X}})^T\varphi(\boldsymbol{\mathcal{X}}_S)$, $\boldsymbol{\mathcal{X}} = [\boldsymbol{\mathcal{X}}_S, \boldsymbol{\mathcal{X}}_T]$, $\boldsymbol{\mathcal{K}} = \varphi(\boldsymbol{\mathcal{X}})^T\varphi(\boldsymbol{\mathcal{X}})$
2. Construct constant matrixs $\boldsymbol{\mathcal{A}}_c$, $\boldsymbol{\mathcal{A}}_k$, $\boldsymbol{\mathcal{B}}_c$, there is
   $\boldsymbol{\mathcal{Z}}_c = \boldsymbol{\mathcal{Z}}\boldsymbol{\mathcal{A}}_c$
   $\boldsymbol{\mathcal{Z}}_k^c = \boldsymbol{\mathcal{B}}_c\boldsymbol{\mathcal{Z}}_k = \boldsymbol{\mathcal{B}}_c\boldsymbol{\mathcal{Z}}\boldsymbol{\mathcal{A}}_k$
   $\boldsymbol{\mathcal{Z}}_c^c = \boldsymbol{\mathcal{B}}_c\boldsymbol{\mathcal{Z}}_c = \boldsymbol{\mathcal{B}}_c\boldsymbol{\mathcal{Z}}\boldsymbol{\mathcal{A}}_c$
3. Initialize: add auxiliary variable $\boldsymbol{\mathcal{J}}$, $\boldsymbol{\mathcal{G}}$, where $\boldsymbol{\mathcal{Z}} = \boldsymbol{\mathcal{J}} = \boldsymbol{\mathcal{G}}$
   add Lag-multipliers $\boldsymbol{\mathcal{R}}_1$, $\boldsymbol{\mathcal{R}}_2$, $\boldsymbol{\mathcal{R}}_3$ and penalty parameter μ.
4. **While** not converge **do**
   4.1 **Step1**: Fix $\boldsymbol{\mathcal{J}}$, $\boldsymbol{\mathcal{G}}$ and $\boldsymbol{\mathcal{Z}}$, and update $\boldsymbol{\Phi}$ by solving eigenvalue decomposition.
   4.2 **Step2**: Fix $\boldsymbol{\Phi}$, and update $\boldsymbol{\mathcal{Z}}$ using ADMM;
   Fix $\boldsymbol{\mathcal{Z}}$ and $\boldsymbol{\mathcal{G}}$, and update $\boldsymbol{\mathcal{J}}$ by using the singular value thresholding (SVT) [1]operator.
   $\boldsymbol{\mathcal{J}}_{K+1} = \min_{\boldsymbol{\mathcal{J}}_K} \lambda_1 \parallel \boldsymbol{\mathcal{J}}_K \parallel_* + \frac{\mu_K}{2} \parallel \boldsymbol{\mathcal{J}}_K - (\boldsymbol{\mathcal{Z}}_K + \frac{\boldsymbol{\mathcal{R}}_{1K}}{\mu_K}) \parallel_F^2$
   Fix $\boldsymbol{\mathcal{Z}}$ and $\boldsymbol{\mathcal{J}}$, and update $\boldsymbol{\mathcal{G}}$ by shrinkage operator
   $\boldsymbol{\mathcal{G}}_{K+1} = shrink(\boldsymbol{\mathcal{Z}}_K + \frac{\boldsymbol{\mathcal{R}}_{2K}}{\mu_K}, \frac{\lambda_2}{\mu_K})$
   Fix $\boldsymbol{\mathcal{J}}$ and $\boldsymbol{\mathcal{G}}$, and update $\boldsymbol{\mathcal{Z}}$ according to Gradient descent operator
   $\boldsymbol{\mathcal{Z}}_{K+1} = \boldsymbol{\mathcal{Z}}_K - \alpha \bullet \frac{\bigtriangledown(\boldsymbol{\mathcal{Z}})}{\|\bigtriangledown(\boldsymbol{\mathcal{Z}})\|}$
   4.3 Update the multipliers $\boldsymbol{\mathcal{R}}_1$, $\boldsymbol{\mathcal{R}}_2$ and $\boldsymbol{\mathcal{R}}_3$
   $\boldsymbol{\mathcal{R}}_1 = \boldsymbol{\mathcal{R}}_1 + \mu(\boldsymbol{\mathcal{Z}} - \boldsymbol{\mathcal{J}})$
   $\boldsymbol{\mathcal{R}}_2 = \boldsymbol{\mathcal{R}}_2 + \mu(\boldsymbol{\mathcal{Z}} - \boldsymbol{\mathcal{G}})$
   $\boldsymbol{\mathcal{R}}_3 = \boldsymbol{\mathcal{R}}_3 + \mu(\mathbf{1}^{1\times N_S}\boldsymbol{\mathcal{Z}} - \mathbf{1}^{1\times N_T})$
   4.4 Update the parameter μ
   $\mu = min(\mu \times 1.01, max_\mu)$
   4.5 Check convergence
end while

Output: $\boldsymbol{\Phi}$ and $\boldsymbol{\mathcal{Z}}$.

4. Experiments

In this section, the experiments on benchmark DA tasks, including cross-domain object recognition: 4DA-CNN object dataset and COIL-20 object dataset, cross-pose face recognition: Multi-PIE face dataset, and cross-domain handwritten digit recognition: USPS dataset, SEMEION dataset and MNIST dataset, have been conducted for evaluation. Several closely related methods, such as SGF [10], GFK [9], SA [7], LTSL [27], DTSL [31], and LSDT [37] have been compared.

4.1. Cross-domain Object Recognition

The benchmark 4DA-CNN office datasets and COIL-20 object dataset have been considered in this section.

Results on 4DA-CNN dataset (Amazon, DSLR, Webcam[1] and Caltech 256[2]) [3, 9]: In 4DA-CNN dataset, four domains simplified as A, D, W, and C are included, with each 10 object classes are contained. The features are extracted by feeding the raw 4DA data into the well trained convolutional neural network (AlexNet) on ImageNet [17], with 8 layers consisting of 5 convolutional layers and 3 fully connected layers. The features with dimension of 4096 from the 6th and 7th layers (e.g. DeCAF [3]) are explored. In the experiment, a standard configuration and protocol is used by following [9]. Specifically, 20 samples per class are selected from Amazon and 8 samples per class from D-SLR, Webcam and Caltech are randomly chosen when they are treated as source domain. 3 samples per class are chosen when they are used as target domain, while the rest data in target domain is used for performance test. The experimental results are shown in Table 1, from which we can observe that the average recognition accuracy of the proposed method shows the best performance, and the superiority is demonstrated in representation based DA methods.

Results on COIL-20 data: Columbia Object Image Library [25]: The COIL-20 dataset[3] contains 20 objects with 1440 gray scale images (72 multi-pose images per object). Each image has 128×128 pixels with 256 gray levels per pixel. In the experiment, by following the experimental protocol in [31], the size of each image is cropped as 32×32. The dataset is divided into two subsets C1 and C2, with each 2 quadrants are contained. Specifically, the C1 set contains quadrants 1 and 3 and the C2 contains quadrants 2 and 4. The two subsets are distribution different but relevant in semantic, and form a DA problem. The experimental results of cross-domain recognition are shown in Table 2, from which our proposed method achieves a significantly better performance over other related methods.

4.2. Cross-poses Face Recognition

The cross-pose face recognition, as a standard DA problem, is conducted. The CMU Multi-PIE face dataset[4] is a popular dataset with 337 subjects, which contains 4 different sessions with 15 poses, 20 illuminations, and 6 expressions. In our experiment, we select the first 60 subjects from Session 1 and Session 2. As a result, a smaller session 1 (**S1**) with 7 images with different poses per class under neutral expression and a smaller session 2 (**S2**) that is similar to S1 but under smile expression are constructed. The experimental configurations are as follows.

S1: One frontal face per subject is used as source training data, one 60° posed face is used as the target training data, and the rest 5 face images are used as the target test data.

S2: The experimental configuration is the same as **S1**.

S1+S2: The two frontal faces and the two 60° posed faces under neutral and smile expression are used as source training data and target training data, respectively. The rest

[1] http://www.eecs.berkeley.edu/~mfritz/domainadaptation/

[2] http://www.vision.caltech.edu/Image_Datasets/Caltech256/

[3] http://www.cs.columbia.edu/CAVE/software/softlib/coil-20.php

[4] http://www.cs.cmu.edu/afs/cs/project/PIE/MultiPie/Multi-Pie/Home.html

Tasks	SourceOnly		Naive Comb		SGF [10]		GFK [9]		SA [7]		LTSL [27]		LSDT [37]		CRTL	
	f_6	f_7	f_6	f_7	f_6	f_7	f_6	f_7	f_6	f_7	f_6	f_7	f_6	f_7	f_6	f_7
$A \rightarrow D$	80.8	81.3	94.5	94.1	90.5	92.0	92.6	94.3	94.2	92.8	95.5	94.5	**96.4**	**96.0**	**96.4**	95.8
$C \rightarrow D$	76.6	77.6	92.9	92.8	93.1	92.4	92.0	91.9	93.0	92.1	93.6	93.5	**95.4**	94.6	95.2	**94.8**
$W \rightarrow D$	96.1	96.2	99.1	98.9	97.7	97.6	97.8	98.5	98.6	98.5	99.1	98.8	**99.4**	**99.3**	**99.4**	**99.3**
$A \rightarrow C$	79.3	79.3	84.0	83.4	77.1	77.4	78.9	79.1	83.1	83.3	85.3	85.4	85.9	**87.0**	**86.2**	**87.0**
$W \rightarrow C$	59.5	68.1	81.7	81.2	74.1	76.8	77.5	76.1	81.1	81.0	82.3	82.6	83.1	84.2	**83.6**	**84.9**
$D \rightarrow C$	67.3	74.3	83.0	82.7	75.9	78.2	78.8	77.5	82.4	82.9	84.4	84.8	85.2	86.2	**85.5**	**86.4**
$D \rightarrow A$	77.0	81.8	90.5	90.9	88.0	88.0	88.9	90.1	90.4	90.7	91.1	91.9	92.2	92.5	**92.5**	**92.7**
$W \rightarrow A$	66.8	73.4	90.1	90.6	87.2	86.8	86.2	85.6	89.8	90.9	90.6	91.0	91.0	91.7	**91.3**	**92.2**
$C \rightarrow A$	85.8	86.5	89.9	90.3	88.5	89.3	87.5	88.4	89.5	89.9	90.4	90.9	**92.1**	**92.5**	92.0	**92.5**
$C \rightarrow W$	67.5	67.8	91.6	90.6	89.4	87.8	87.7	86.4	91.2	89.0	91.8	90.8	**93.3**	**93.5**	92.7	93.1
$D \rightarrow W$	95.4	95.1	97.9	98.0	96.8	95.7	97.0	96.5	97.5	97.5	98.2	97.8	**98.7**	98.3	**98.7**	**98.5**
$A \rightarrow W$	70.5	71.6	90.4	91.1	87.2	88.1	89.5	88.6	90.3	87.8	92.2	91.5	92.1	92.9	**92.3**	**93.0**
Average	76.9	79.4	90.5	90.4	87.1	87.5	87.9	87.8	90.1	89.7	91.2	91.1	92.1	92.4	**92.2**	**92.5**

Table 1: Recognition accuracy (%) of different domain adaptation over 10 object categories on 4DA-CNN with deep feature representation

Tasks	SVM	TSL	RDALR [15]	LTSL [27]	DTSL [31]	LSDT [37]	CRTL
$C1 \rightarrow C2$	82.7	80.0	80.7	75.4	84.6	81.7	**87.0**
$C2 \rightarrow C1$	84.0	75.6	78.8	72.2	84.2	81.5	**86.5**
Average	83.3	77.8	79.7	73.8	84.4	81.6	**86.8**

Table 2: Recognition accuracy (%) of different domain adaptation on COIL-20

10 face images are used as target test data.

S1 $\rightarrow$ **S2**: The faces in **S1** are used as source training data, the frontal and 60° posed faces in **S2** are used as the target training data, and the rest data are used as test data.

With above settings, the recognition accuracies of different methods have been shown in Table 3. It is obvious that the proposed method performs significantly better over other DA methods in handling such pose change based nonlinear transfer problem.

4.3. Cross-domain Handwritten Digits Recognition

Three handwritten digits datasets: MNIST (M)[5], USPS (U)[6] and SEMEION (S)[7] with 10 classes from digit $0 \sim 9$ are used for evaluating the proposed CRTL method. The MNIST dataset consists of 70,000 instances with image size of 28×28, the USPS dataset consists of 9298 examples with image size of 16×16, and the SEMEION dataset consists of 2593 images with size of 16×16. In experiments, we crop the MNIST dataset into 16×16. For DA experiment, each dataset is used as the source and target domain alternatively, and 6 cross-domain tasks are obtained. Also, 100 samples per class from source domain and 10 samples per class from target domain randomly are selected for training. 5 random splits are used, and the average classification accuracies are reported in Table 4. From the results, we observe that our

[5] http://yann.lecun.com/exdb/mnist/

[6] http://www-i6.informatik.rwth-aachen.de/~keysers/usps.html

[7] http://archive.ics.uci.edu/ml/datasets/Semeion+Handwritten+Digit

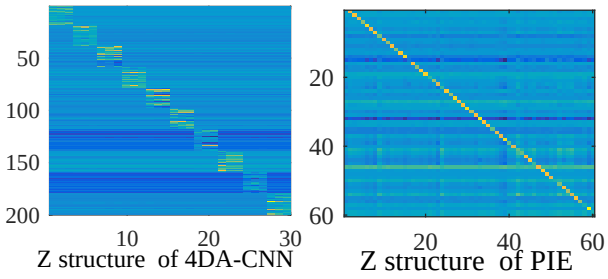

Figure 3: Visualization of reconstruction matrix $\mathcal{Z}$

CRTL outperforms other representation based DA methods. The superiority is therefore proved.

5. Discussion

5.1. Parameter Setting

In our method, the trade-off coefficients β_1, β_2, λ_1 and λ_2 are fixed as 1 in experiments. The Gaussian kernel function $k(\mathbf{x}_i, \mathbf{x}_j) = \exp(- \parallel \mathbf{x}_i - \mathbf{x}_j \parallel^2 / 2\sigma^2)$ is used, where σ can be tuned for different tasks. e.g. $\sigma = 1.2$ for 4DA-CNN, $\sigma = 0.5$ for COIL-20, $\sigma = 0.2$ for CMU Multi-PIE and $\sigma = 1.0$ for handwritten digits. The SVM classifier is used in COIL-20 experiment, and least square classifier is used in other domain adaptation experiments.

Tasks	Naive Comb	A-SVM	SGF [10]	GFK [9]	SA [7]	LTSL [27]	LSDT [37]	CRTL
S1 ($0° \rightarrow 60°$)	61.0	57.0	53.7	56.0	51.3	56.0	59.7	**65.7**
S2 ($0° \rightarrow 60°$)	62.7	62.7	55.0	58.7	62.7	60.7	63.3	**69.0**
S1+S2 ($0° \rightarrow 60°$)	60.2	60.1	53.8	56.3	61.7	60.7	61.7	**68.5**
S1→S2	93.6	94.3	92.5	96.7	98.3	96.7	95.8	**98.7**
Average	69.4	68.5	63.8	67.0	68.5	68.5	70.1	**75.5**

Table 3: Recognition accuracy (%) of different domain adaptation on face recognition across poses

Tasks	Naive Comb	A-SVM	SGF [10]	GFK [9]	SA [7]	LTSL [27]	LSDT [37]	CRTL
$M \rightarrow U$	78.8	78.3	79.2	82.6	78.8	83.2	79.3	**85.4**
$S \rightarrow U$	83.6	76.8	77.5	82.7	82.5	83.6	84.7	**86.2**
$M \rightarrow S$	51.9	70.5	51.6	70.5	74.4	72.8	69.1	**76.2**
$U \rightarrow S$	65.3	74.5	70.9	76.7	74.6	65.3	67.4	**82.6**
$U \rightarrow M$	71.7	73.2	71.1	74.9	72.9	71.7	70.5	**82.0**
$S \rightarrow M$	67.6	69.3	66.9	74.5	72.9	67.6	70.0	**78.4**
Average	69.8	73.8	69.5	77.0	76.0	74.0	73.5	**81.8**

Table 4: Recognition accuracy (%) of different domain adaptation on handwritten digits recognition

5.2. Visualization

Fig.3 shows the visualization of the reconstruction matrix $\mathcal{Z}$ in 4DA-CNN and CMU Multi-PIE datasets, from which the block-diagonal structure of matrix $\mathcal{Z}$ can be clearly observed. The proposed method is effective in preserving the class-specific characteristic of $\mathcal{Z}$. It becomes robust even when data is badly corrupted [18].

5.3. Convergence

The convergence of CRTL with iteration number t on PIE (S1+S2) and COIL-20 ($C1 \rightarrow C2$) recognition tasks are shown in Fig.4, from which we see that the algorithm can converge, but small perturbation still exists. This is not strange in non-convex optimization, because there is no closed-form solution of $\mathcal{Z}$ which also can be seen in convergence curves of the l_1 -norm and nuclear norm of $\mathcal{Z}$.

5.4. Computational Complexity Analysis

The computational complexity of **Algorithm 1** is presented. The algorithm includes two steps: update $\mathcal{Z}$ and update $\mathbf{\Phi}$. The computation of $\mathbf{\Phi}$ involves eigen-decomposition and matrix multiplication, and the complexity is $O(N^3)$. The computation of updating $\mathcal{Z}$ involves updating of $\mathcal{J}$, $\mathcal{G}$ and $\mathcal{Z}$. Thus the complexity of computing $\mathcal{Z}$ is $O(N^2) + O(N^2) + O(N^2)$. Suppose that the number of iterations in **Algorithm 1** is T, then the total computational complexity of CRTL can be expressed as $O(TN^3) + O(TN^2) + O(TN^2) + O(TN^2)$. Note that the complexity of kernel Gram matrix computation is not included here, which is outside the optimization loop.

6. Conclusion

In previous work [27, 31, 37], the sparse and low-rank constraints are considered for learning a structured $\mathcal{Z}$ for domain adaptation. All of them neglect the class prior distribution in modeling, and the statistical dependency between features and labels are also forgotten. To address them, we propose a class-specific reconstruction transfer learning (CRTL) method, which aims at constructing a feature augmented model without class bias. $First$, we cast the transfer learning problem by a class-specific reconstruction matrix $\mathcal{Z}$ modeling and optimization problem. $Second$, in order to keep the intrinsic statistical dependency between the domain data and labels after feature projection, a Projected Hilbert-Schmidt Independency Criterion (**pHSIC**) in RKHS is explored in CRTL. $Third$, for better insight of the global and local structure in $\mathcal{Z}$, the joint low-rank and sparse constraints are imposed. Extensive experiments on benchmark DA datasets demonstrate the superiority the proposed method over other related methods.

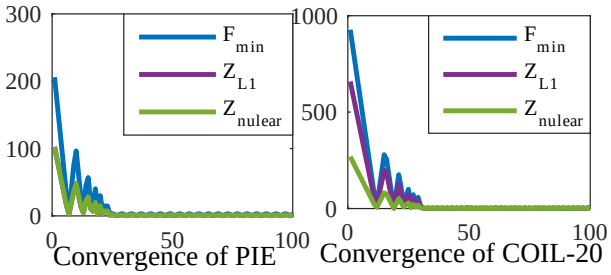

Figure 4: Convergence curve of the objective function

Acknowledgements

This work was supported by the National Science Fund of China under Grants (61771079, 61401048) and the Fundamental Research Funds for the Central Universities (No. 106112017CDJQJ168819).

References

[1] J. F. Cai, E. J. Candes, and Z. Shen. A singular value thresholding algorithm for matrix completion. *Siam Journal on Optimization*, 20(4):1956–1982, 2008.

[2] Z. Ding, M. Shao, and Y. Fu. Deep low-rank coding for transfer learning. In *IJCAI*, 2015.

[3] 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. *Computer Science*, 50(1):815–830, 2013.

[4] L. Duan, D. Xu, I. W. Tsang, and J. Luo. Visual event recognition in videos by learning from web data. In *CVPR*, pages 1959–1966, 2010.

[5] E. Elhamifar and R. Vidal. Sparse subspace clustering: algorithm, theory, and applications. *PAMI*, 35(11):2765–2781, 2012.

[6] J. Feng, Z. Lin, H. Xu, and S. Yan. Robust subspace segmentation with block-diagonal prior. In *CVPR*, pages 3818–3825, 2014.

[7] B. Fernando, A. Habrard, M. Sebban, and T. Tuytelaars. Unsupervised visual domain adaptation using subspace alignment. In *ICCV*, pages 2960–2967, 2013.

[8] J. Ghosn and Y. Bengio. Bias learning, knowledge sharing. *IEEE Transactions on Neural Networks*, 14(4):748–765, 2003.

[9] B. Gong, Y. Shi, F. Sha, and K. Grauman. Geodesic flow kernel for unsupervised domain adaptation. In *CVPR*, pages 2066–2073, 2012.

[10] R. Gopalan, R. Li, and R. Chellappa. Domain adaptation for object recognition: An unsupervised approach. In *ICCV*, pages 999–1006, 2011.

[11] A. Gretton, K. Borgwardt, M. J. Rasch, B. Scholkopf, and A. J. Smola. A kernel method for the two-sample problem. *NIPS*, abs/0805.2368(2007):513 – 520, 2008.

[12] A. Gretton, O. Bousquet, A. Smola, and B. Scholkopf. Measuring statistical dependence with hilbert-schmidt norms. In *ALT*, volume 16, pages 63–78. Springer, 2005.

[13] J. Hoffman, E. Rodner, J. Donahue, B. Kulis, and K. Saenko. Asymmetric and category invariant feature transformations for domain adaptation. *IJCV*, 109(1):28–41, 2014.

[14] H. C. A. Hsu T M H, Chen W Y. Unsupervised domain adaptation with imbalanced cross-domain data. In *ICCV*, pages 4121–4129, 2015.

[15] I. H. Jhuo, D. Liu, D. T. Lee, and S. F. Chang. Robust visual domain adaptation with low-rank reconstruction. In *CVPR*, pages 2168–2175, 2012.

[16] R. J.Yang and A. Hauptmann. Cross-domain video concept detection using adaptive svms. *ACM MM*, 2007.

[17] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. *NIPS*, 25(2):1097–1105, 2012.

[18] Y. Li, J. Liu, H. Lu, and S. Ma. Learning robust face representation with classwise block-diagonal structure. *IEEE TIFS*, 9(12):2051–2062, 2014.

[19] G. Liu, Z. Lin, and Y. Yu. Robust subspace segmentation by low-rank representation. In *ICML*, pages 663–670, 2010.

[20] C. Y. Lu, H. Min, Z. Q. Zhao, L. Zhu, D. S. Huang, and S. Yan. Robust and efficient subspace segmentation via least squares regression. In *ECCV*, pages 347–360, 2012.

[21] L.Zhang and D.Zhang. Domain adaptation extreme learning machines for drift compensation in e-nose systems. *IEEE Trans.Instru. Meas*, 64(7):1790–1801, 2015.

[22] G. Mesnil, Y. Dauphin, X. Glorot, S. Rifai, Y. Bengio, I. Goodfellow, E. Lavoie, X. Muller, G. Desjardins, and D. Warde-Farley. Unsupervised and transfer learning challenge: a deep learning approach. *Workshop on Unsupervised and Transfer Learning*, 7:1–15, 2012.

[23] S. M.Kan, J.Wu and X.Chen. Domain adaptation for face recognition: Targetize source domain bridged by common subspace, 2014. IJCV.

[24] H. V. Nguyen, H. T. Ho, V. M. Patel, and R. Chellappa. Dash-n: Joint hierarchical domain adaptation and feature learning. *IEEE Transactions on Image Processing*, 24(12):5479–5491, 2015.

[25] C. Rate and C. Retrieval. Columbia object image library (coil-20). *Computer*, 2011.

[26] K. Saenko, B. Kulis, M. Fritz, and T. Darrell. Adapting visual category models to new domains. 2010.

[27] M. Shao, D. Kit, and Y. Fu. Generalized transfer subspace learning through low-rank constraint. *IJCV*, 109(1):74–93, 2014.

[28] S. Shekhar, V. M. Patel, H. V. Nguyen, and R. Chellappa. Generalized domain-adaptive dictionaries. In *CVPR*, pages 361–368, 2013.

[29] B. Sun, J. Feng, and K. Saenko. Return of frustratingly easy domain adaptation. In *AAAI*, volume 6, page 8, 2016.

[30] B. Sun and K. Saenko. Deep coral: Correlation alignment for deep domain adaptation. *arXiv preprint arXiv:1607.01719*, 2016.

[31] Y. Xu, X. Fang, J. Wu, X. Li, and D. Zhang. Discriminative transfer subspace learning via low-rank and sparse representation. *IEEE Trans Image Process*, 25(2):850–863, 2015.

[32] Y. Xue, X. Liao, L. Carin, and B. Krishnapuram. Multi-task learning for classification with dirichlet process priors. *Journal of Machine Learning Research*, 8(1):35–63, 2007.

[33] M. Yang, L. Zhang, X. Feng, and D. Zhang. Fisher discrimination dictionary learning for sparse representation. In *ICCV*, pages 543–550, 2011.

[34] Y. Yao and G. Doretto. Boosting for transfer learning with multiple sources. In *CVPR*, pages 1855–1862, 2010.

[35] H. Zhang, V. M. Patel, S. Shekhar, and R. Chellappa. Domain adaptive sparse representation-based classification. In *IEEE International Conference and Workshops on Automatic Face and Gesture Recognition*, pages 1–8, 2015.

[36] L. Zhang and D. Zhang. Robust visual knowledge transfer via extreme learning machine based domain adaptation. *IEEE Trans Image Process*, 25(10):1–1, 2016.

[37] L. Zhang, W. Zuo, and D. Zhang. Lsdt: Latent sparse domain transfer learning for visual adaptation. *IEEE Trans Image Process*, 25(3):1177–1191, 2016.