Homologous Component Analysis for Domain Adaptation

Youfa Liu, Weiping Tu, Bo Du, *Senior Member, IEEE*, Lefei Zhang, *Member, IEEE*, and Dacheng Tao, *Fellow, IEEE*

***Abstract*— Covariate shift assumption-based domain adaptation approaches usually utilize only one common transformation to align marginal distributions and make conditional distributions preserved. However, one common transformation may cause loss of useful information, such as variances and neighborhood relationship in both source and target domains. To address this problem, we propose a novel method called homologous component analysis (HCA) where we try to find two totally different but homologous transformations to align distributions with side information and make conditional distributions preserved. As it is hard to find a closed-form solution to the corresponding optimization problem, we solve them by means of the alternating direction minimizing method (ADMM) in the context of Stiefel manifolds. We also provide a generalization error bound for domain adaptation in the semi-supervised case, and two transformations can help to decrease this upper bound more than only one common transformation does. Extensive experiments on synthetic and real data show the effectiveness of the proposed method by comparing its classification accuracy with the state-of-the-art methods, and the numerical evidence on chordal distance and Frobenius distance shows that resulting optimal transformations are different.**

***Index Terms*— Visual domain adaptation, homologous component analysis, visual categorization.**

I. Introduction

MANY machine learning methods usually depend on the assumption that the training and test data have the same distribution. However, there would be problems with this assumption. In indoor WiFi localization problem which attempts to detect a user's current location based on the past collected WiFi data, the performance of a learnt model in one time period (the source domain) may be reduced for another location in a later period (the target domain) [1]. Domain adaptation aims to transfer the knowledge from the source domain to the target domain by leveraging the available data and proves to be able to tackle this kind of problems. Domain adaptation can be coarsely categorized into two families, i.e. unsupervised domain adaptation and semi-supervised domain adaptation. For unsupervised domain adaptation, there is no available labeled data form target domain. For semi-supervised domain adaptation, there is a large amount of available labeled data which comes from the source domain but few available labeled data is from the target domain. This paper concerns the later. Domain adaptation approaches have many successful applications in computer vision [2]–[6], [46]–[49], [53], [54], recommendation system [7]–[9], natural language processing [10]–[12], reinforcement learning [13]–[15] and many other areas.

For most domain adaptation methods, at least one of two assumptions are made:

(1) *class Imbalance [1], [38], [40]:* The distributions of labels in the source and target domain are different but the conditional distributions of data with respect to the labels are the same, i.e.

$$\Pr_S(\mathbf{y}) \neq \Pr_T(\mathbf{y}), \quad \Pr_S(\mathbf{x}|\mathbf{y}) = \Pr_T(\mathbf{x}|\mathbf{y}),$$

where $\Pr_S$ and $\Pr_T$ represent distributions on source domain and target domain respectively;

(2) *covariate Shift [1], [38], [41] or* Sample Selection Bias *[42], [43]:* the conditional distributions of the labels with respect to data are equal, but the data distribution in two domains are different, i.e.

$$\Pr_S(\mathbf{x}) \neq \Pr_T(\mathbf{x}), \quad \Pr_S(\mathbf{y}|\mathbf{x}) = \Pr_T(\mathbf{y}|\mathbf{x}).$$

The difference should be small to ensure the effectiveness of the domain adaptation techniques [16].

This paper studies domain adaptation with the assumption of covariant shift. Large amounts of works on covariate shift concentrate on feature representation based methods. There are some recent works, such as TCA [17], GFK [18], CORAL [3], LSDT [2], JGSA [19] and VDA [20]. Pan *et al.* [17] proposed the method called transfer component analysis (TCA) which attempts to learn some transfer components cross

Manuscript received August 14, 2018; revised March 14, 2019 and April 30, 2019; accepted July 1, 2019. Date of publication July 29, 2019; date of current version November 4, 2019. This work was supported in part by the National Natural Science Foundation of China under Grant 61822113, Grant 41871243, and Grant 61671335, in part by the Natural Science Foundation of Hubei Province under Grant 2018CFA050, in part by the National Key R&D Program of China under Grant 2018YFA0605500, in part by the Fundamental Research Funds for the Central Universities under Grant 2042018kf0206, and in part by the Australian Research Council Projects under Grant FL-170100117, Grant DP-180103424, and Grant IH-180100002. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Keigo Hirakawa. *(Corresponding authors: Weiping Tu; Bo Du.)*

Y. Liu, W. Tu, and B. Du are with School of Computer Science, Wuhan University, Wuhan 430072, China, and also with the National Engineering Research Center for Multimedia Software, Wuhan University, Wuhan 430072, China (e-mail: liuyfa1991@whu.edu.cn; tuweiping@whu.edu.cn; remoteking@whu.edu.cn).

L. Zhang is with the School of Computer Science, Wuhan University, Wuhan 430072, China (e-mail: zhanglefei@whu.edu.cn).

D. Tao is with the UBTECH Sydney Artificial Intelligence Centre, Faculty of Engineering, The University of Sydney, Darlington, NSW 2008, Australia, and also with the School of Computer Science, Faculty of Engineering, The University of Sydney, Darlington, NSW 2008, Australia (e-mail: dacheng.tao@sydney.edu.au).

Digital Object Identifier 10.1109/TIP.2019.2929421

1057-7149 © 2019 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission.
See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

domains in a reproducing kernel Hilbert space (RKHS) via maximum mean discrepancy (MMD). Data distributions in different domains are aligned in the subspace generated by these transfer components. Gong *et al.* [18] proposed the kernel method called GFK which models domain shift by integrating an infinite number of subspaces that characterize changes in geometric and statistical properties from the source to the target domain. Sun *et al.* [3] proposed the CORrelation ALignment (CORAL) which minimizes domain shift by aligning the second-order statistics of source and target distributions. Zhang *et al.* [2] proposed the reconstruction-based method called laten sparse domain transfer (LSDT) which is a joint learning model combining the sparse coding and subspace representation. LSDT method is extended to a kernel-based framework for tackling non-linear subspace shifts in reproducing kernel Hilbert space (RKHS). Zhang *et al.* [19] proposed a unified framework called JGSA which reduces the shift between the source and target domain in both statistical and geometric sense. Zhang *et al.* proposed JGSA [19] which learns two coupled projections that project the source domain and target domain data into low-dimensional subspaces where the geometrical shift and distribution shift are reduced simultaneously. Tahmoresnezhad and Hashemi [20] proposed the visual domain adaptation (VDA) approach which tries to reduce the joint distribution and conditional distribution across domains. These methods aim at discovering a good feature representation across domains such that domain divergence can be reduced [1], [39] and hence the well learnt feature representation help to transfer knowledge form source domain to target domain.

These works except JGSA [19] aim to find a commonly used transformation to project the source and target domain data into a latent space so as to align the distributions of transformed data. It is worth mentioning that JGSA [19] seemingly uses two transformations but actually one transformation because Frobenius distance of two transformation in the proposed model should be very small. Although this kind of idea is feasible and indeed changes the covariate shift problem into the traditional machine learning setting, the important information in source and target domain, for instances, variances and neighborhood relationship in source and target domain, may be lost.

The lost information may help to enhance the performance of classifiers. One transformation may not make full use of these information in source and target domain at the same time because of the essential difference of two domains. To utilize these useful information in both source and target domain, we try to find two *essentially different transformations* $\wp_S$ and $\wp_T$ under the homologous constraint to align the distributions between projected source and target data with side information (i.e. the projected data information with respect to source domain and target domain) and preserve the conditional distributions, i.e. $\Pr_S(\wp_S(\mathbf{x})) \approx \Pr_T(\wp_T(\mathbf{x}))$, $\Pr_S(\mathbf{y}|\wp_S(\mathbf{x})) = \Pr_T(\mathbf{y}|\wp_T(\mathbf{x}))$, and maximize the variance of projected source and target data. To realize this point, we propose the homologous component analysis method. We display it intuitively in Figure 1. Two independent transformations may lead to negative domain adaptation(or negative transfer learning [1], [32]). Homologous analysis component is based on the homologous relationship between two using transformations. Given two transformations $\wp_S$ and $\wp_T$ which come from a Stiefel manifold $\mathcal{S}_D^d$, we impose the homologous constraint (See Definition 2 for homology) on them.

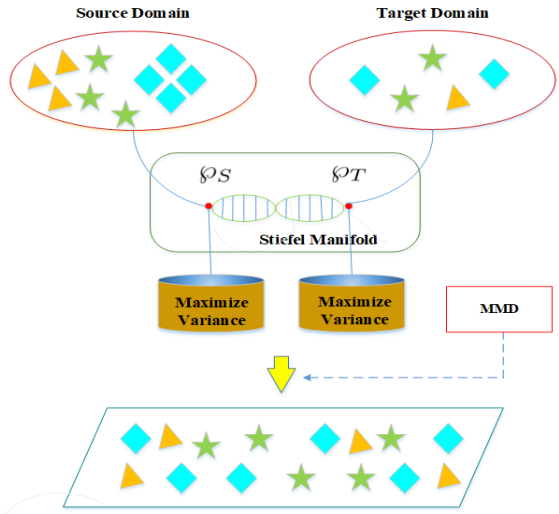


Fig. 1. Illustration of homologous component analysis. It consists of three parts: variance maximization, distribution alignment realized by MMD and homologous constraint. The DNA shape which links two transformations represents the homologous relationship between transformations $\wp_S$ and $\wp_T$. MMD represents maximum mean discrepancy (See part A in section IV for details.) which is used for distributions alignment.

The first theoretical analysis of the domain adaptation problem occurred in [16]. The VC-dimension based generalization bounds for domain adaptation in classification tasks are given and a kind of distance $d_{H\triangle H}$ induced by symmetric difference hypothesis space H is introduced. To make the domain adaptation more efficient, labeled data in target domain is leveraged. In other words, we consider semi-supervised domain adaptation. However, semi-supervised way is empirically effective. Based on [16], we derive a generalization bound for domain adaptation in semi-supervised case. Two transformations can help to decrease upper bound over source samples and labeled target samples more than only one common transformation does.

We summarize contributions of this paper are as follows:

(1) Define the homology of two transformations. Homologous relationship helps to transfer useful knowledge from source domain to target domain.

(2) We for the first time incorporate side information into the computation of MMD where the side information means the projected source data information. We also use the labels information in source domain to help distribution alignment.

(3) To the best of our knowledge, we are the first to provide a generalized error bound for semi-supervised domain adaptation which is related to the ratio of target labeled samples and source labeled samples, and includes the term which has the same structure over source and target labeled samples as empirical risk minimization.

The rest of this paper is organized as follows. In section II, we give a brief overview of the related work in domain adaptation. In section III, we present the notations. In section IV,

we introduce the necessary knowledge for the proposed model. In section V, we establish homologous component analysis model and we provide optimization procedure for our proposed methods in Appendix C. In section VI, we theoretically analyze the proposed models. In section VII, we conduct experiments on real-world datasets. Finally, we conclude the paper in section VIII.

II. Related Works

1) TCA [17]: Pan *et al.* [17] proposed the method called transfer component analysis (TCA) which attempts to learn some transfer components cross domains in a reproducing kernel Hilbert space (RKHS) via maximum mean discrepancy (MMD). Data distributions in different domains are aligned in the subspace generated by these transfer components. The source and target distribution can be empirically measured by

$$\mathrm{Dist}(X'_S, X'_T) = \| \frac{1}{n_s} \sum_{i=1}^{n_s} \phi(x_{S_i}) - \frac{1}{n_t} \sum_{i=1}^{n_t} \phi(x_{T_i}) \|^2_{\mathcal{H}} = tr(KL),$$

where $X'_S = \{\phi(x_{S_i})\}$, $X'_T = \{\phi(x_{T_i})\}$, ϕ is nonlinear mapping which embeds samples into a RKHS $\mathcal{H}$ and K is a kernel matrix over all samples from source and target domain; and $L = [L_{ij}] \succeq 0$ with $L_{ij} = \frac{1}{n_s^2}$ if $x_i, x_j \in X_S$; $L_{ij} = \frac{1}{n_t^2}$ if $x_i, x_j \in X_T$; otherwise $-\frac{1}{n_s n_t}$. To avoid the use of semi-definite programming (SDP) which produces high computational burden, Pan *et al.* [17] propose to use empirical kernel mapping $KK^{-\frac{1}{2}}$ [45] and then get

$$\mathrm{Dist}(X'_S, X'_T) = tr(W^T KLKW).$$

In our proposed model, we refer to this method but take side information (i.e. the projected data information with respect to source domain and target domain) into account.

2) JGSA [19]: Zhang *et al.* [19] proposed a unified framework called JGSA which reduces the shift between the source and target domain in both statistical and geometric sense. This model seemingly applies two transformations A and B to align distributions but actually can be understood as only using one transformation because subspace shift forces A to be close to B. Our proposed model use two different but homologous transformations to distribution shift and we provide the numerical evidence to show the difference of two resulted optimal transformations.

3) VDA [20]: Tahmoresnezhad and Hashemi [20] proposed the visual domain adaptation (VDA) approach which tries to reduce the joint distribution and conditional distribution across domains. There is one commonly used transformation. In our proposed models, we use two transformations under homologous constraint in MMD.

4) Other Works: Bousmalis *et al.* [50] extended domain adaptation methods for training a grasping system to grasp new objects from raw monocular RGB images. Mahmood *et al.* [51] proposed a reverse flow method which utilizes adversarial training to make real medical images more like synthetic images and assume that clinically-relevant features can be preserved via self-regularization. Rozantsev *et al.* [52] proposed a method beyond sharing weights for domain adaptation. [55], [61], [64] and [65] are MMD-based deep domain adaptation references.

III. Notations

In this paper, we denote source and target domain data by $\mathbf{X}_S$ and $\mathbf{X}_T = \mathbf{X}_{Tl} \cup \mathbf{X}_{Tu}$ respectively, where $\mathbf{X}_S = \{x_S^1, x_S^2, \ldots, x_S^{m_S}\} \subseteq \mathbb{R}^{D \times m_S}$ with available labels $\mathbf{y}_S = \{y_S^1, y_S^2, \ldots, y_S^{m_S}\}$, $\mathbf{X}_{Tl} = \{x_T^1, x_T^2, \ldots, x_T^{m_{Tl}}\} \subseteq \mathbb{R}^{D \times m_{Tl}}$ with available labels $\{y_T^1, y_T^2, \ldots, y_T^{m_{Tl}}\}$ and $\mathbf{X}_{Tu} = \{x_T^{m_{Tl}+1}, x_T^{m_{Tl}+2}, \ldots, x_T^{m_{Tl}+m_{Tu}}\} \subseteq \mathbb{R}^{D \times m_{Tu}}$ whose labels are unavailable. $m_{Tl} \ll m_S$. For convenience, let $m_T = m_{Tl} + m_{Tu}$. Denote $X_S = [x_S^1, x_S^2, \ldots, x_S^{m_S}]$, $Y_S = [y_S^1, y_S^2, \ldots, y_S^{m_S}]^T$, $X_T = [x_T^1, x_T^2, \ldots, x_T^{m_T}]$ and $Y_{Tl} = [y_T^1, y_T^2, \ldots, y_T^{m_{Tl}}]^T$ respectively. Let $\mathcal{S}_D^d$ be a Stiefel manifold and $\mathcal{S}_D^d = \{P \in \mathbb{R}^{D \times d} : P^T P = I_d\}$ [21], where I_d is the identity matrix of size $d \times d$. Let $\mathcal{G}(d, D)$ be a Grassmann manifold [26]. It consists of all d-dimensional subspaces embedded in D-dimensional Euclidean spaces $\mathbb{R}^D$, where $1 \leq d \leq D$. One convenient way is to represent Grassmann manifold by the equivalent classes of all the thin-tall orthogonal matrices under the orthogonal group $\mathcal{O}(d)$ of order d. Hence $\mathcal{G}(d, D) = \mathcal{S}_D^d / \mathcal{O}(d)$ [22]. $\langle \cdot, \cdot \rangle$ denotes the standard inner product in Euclidean space $\mathbb{R}^D$. If u and v belong to $\mathbb{R}^D$, their tensor product is $u \otimes v := uv^T \in \mathbb{R}^{D \times D}$. $\| \cdot \|_2$ denotes the l_2-norm in Euclidean space $\mathbb{R}^D$. $\| \cdot \|_F$ denotes the Frobenius norm on $\mathbb{R}^{D \times D}$. If $A \in \mathbb{R}^{p \times q}$, we denote by A^T its transposition.

IV. Preliminaries

A. Maximum Mean Discrepancy (MMD)

We follow the notations in [23]. Let $\mathcal{X}$ be a compact metric space, x and y be random variables on it with two Borel probability measures P and Q. Let $X = \{x_1, x_2, \ldots, x_m\}$ and $Y = \{y_1, y_2, \ldots, y_n\}$ be two independently and identically distributed (i.i.d) samples form P and Q respectively. Gretton *et al.* [23] research on determining whether two distributions P and Q coincide.

Definition 1: [23] Let $\mathcal{F}$ be a class of functions $f : \mathcal{X} \to \mathbb{R}$, we define the maximum discrepancy (MMD) as

$$\mathrm{MMD}[\mathcal{F}, P, Q] := \sup_{f \in \mathcal{F}} \left(\mathbf{E}_{x \sim P}[f(x)] - \mathbf{E}_{y \sim Q}[f(y)] \right).$$

A biased empirical estimation of the MMD is obtained by replacing the population expectations with empirical expectations computed on the samples X and Y,

$$\mathrm{MMD}_b[\mathcal{F}, X, Y] := \sup_{f \in \mathcal{F}} \left(\frac{1}{m} \sum_{i=1}^{m} f(x_i) - \frac{1}{n} \sum_{i=1}^{n} f(y_i) \right).$$

Gretton *et al.* [23] consider the function class of $\mathcal{F}$ as the unit ball in a reproducing kernel Hilbert space (RKHS) $\mathcal{H}$ with Mercer kernel $k : \mathcal{X} \times \mathcal{X} \longrightarrow \mathbb{R}$. The mean embedding of probability distribution P into $\mathcal{H}$ is an element $\mu_P \in \mathcal{H}$ such that $\mathbf{E}_{x \sim P}[f(x)] = \langle f, \mu_P \rangle_{\mathcal{H}}, \forall f \in \mathcal{H}$. If $\mathcal{H}$ is universal, then MMD is a metric on $\mathcal{X}$. In this case, $\mathrm{MMD}[\mathcal{F}, P, Q] = 0$ if and only if $P = Q$. Steinwart [24] proves that Gaussian and Laplace RKHSs are universal. In this paper, we simply use

the Gaussian RKHS. The empirical estimation of MMD over samples X and Y is

$$\begin{aligned}\mathrm{MMD}_b^2[\mathcal{F}, X, Y] : = & \frac{1}{m^2}\sum_{i=1}^{m}\sum_{j=1}^{m}k(x_i, x_j) \\ & -\frac{1}{mn}\sum_{i=1}^{m}\sum_{j=1}^{n}k(x_i, y_j) \\ & +\frac{1}{n^2}\sum_{i=1}^{n}\sum_{j=1}^{n}k(y_i, y_j)\end{aligned}$$

[25], [55].

B. Chordal Distance

Let P_S and P_T belong to $\mathcal{S}_D^d$. $\mathcal{P}_S$ and $\mathcal{P}_T$ are the spaces spanned by the columns of P_S and P_T respectively. They both are subspaces of $\mathbb{R}^D$. We state the principal angles between spaces $\mathcal{P}_S$ and $\mathcal{P}_T$ as in [44]. The smallest principal angle $\theta_1(\mathcal{P}_S, \mathcal{P}_T) = \theta_1 \in [0, \frac{\pi}{2}]$ between spaces $\mathcal{P}_S$ and $\mathcal{P}_T$ is defined by $\cos\theta_1 =$

$$\begin{aligned}&\max\ u^T v \\ &\text{s.t. } u \in \mathcal{P}_S, v \in \mathcal{P}_T, \\ &\quad \|u\|_2 = 1, \quad \|v\|_2 = 1.\end{aligned}$$

Assume the maximum of the above optimization is achieved at points $u = u_1$, $v = v_1$. Then $\theta_2(\mathcal{P}_S, \mathcal{P}_T)$ is defined as the smallest angle between the orthogonal complement of $\mathcal{P}_S$ with respect to u_1 and that of P_T with respect to v_1, and so forth, until one of the spaces is empty. Then we have $\cos\theta_k =$

$$\begin{aligned}&\max\ u^T v = u_k^T v_k \\ &\text{s.t. } u \in \mathcal{P}_S, \quad v \in \mathcal{P}_T, \\ &\quad \|u\|_2 = 1, \quad \|v\|_2 = 1, \\ &\quad u_j^T u = 0, \quad j = 1, 2, \ldots, k-1, \\ &\quad v_j^T v = 0, \quad j = 1, 2, \ldots, k-1.\end{aligned}$$

$P_S^T P_T = U(\cos\Theta)V^T$ is the singular value decomposition (SVD), where θ_i's are the principal angles between $\mathcal{P}_S$ and $\mathcal{P}_T$ and $\cos(\Theta)$ is the diagonal matrix $\mathrm{diag}(\cos\theta_1, \cos\theta_2, \ldots, \cos\theta_d)$ [26]. The chordal distance is

$$\begin{aligned}\mathbf{d}(\mathcal{P}_S, \mathcal{P}_T) &= \frac{1}{\sqrt{2}}\|P_S P_S^T - P_T P_T^T\|_F \\ &= \left(\sum_{i=1}^{d}\sin^2\theta_i)\right)^{\frac{1}{2}}\end{aligned}$$

[26]. This distance is also called embedding distance in [22].

V. Homologous Component Analysis

Firstly, we introduce the notion of homology between two transformations. Then we provide a convenient criterion to characterize the homology, and later we propose homologous component analysis model. The optimization scheme will be postponed to section VI.

A. The Proposed Method: HCA

We propose a new notion of homology between two transformations. The homologous relationship constraint used in our proposed method is reasonable because it obeys the principle of transferring the knowledge from the source to improve the performance of classifier in target domain.

Definition 2: Let P_S and P_T belong to the Stiefel manifold $\mathcal{S}_D^d$. Two transformations P_S and P_T are homologous if the vector spaces generated by their columns respectively are identical.

Proposition 1: If P_S and P_T are homologous, then conditional distributions are preserved, i.e. $\Pr_S(\mathbf{y}|P_S^T\mathbf{x}) = \Pr_T(\mathbf{y}|P_T^T\mathbf{x})$.

Proof: Because both P_S and P_T belong to $\mathcal{S}_D^d$, they are injective mappings. This means that $\Pr_S(\mathbf{y}|P_S^T\mathbf{x}) = \Pr_T(\mathbf{y}|P_T^T\mathbf{x})$. □

The homology is related to matroids [27].

Proposition 2: If P_S and P_T are homologous, then they induce the same matroid.

Proof: Because P_S and P_T are homologous, they induce the same vector spaces. Note that vector spaces are matroids. Hence P_S and P_T induces the same matroid. □

The following proposition offers a convenient criterion to determine whether P_S and P_T are homologous. Let $\mathcal{P}_S$ and $\mathcal{P}_T$ be vector spaces spanned by the columns of P_S and P_T respectively.

Proposition 3: P_S and P_T are homologous if and only if the related chordal distance $\mathbf{d}(\mathcal{P}_S, \mathcal{P}_T)$ is zero.

Proof: By definition 2, the homology of P_S and P_T implies $\mathcal{P}_S = \mathcal{P}_T$, hence $\mathbf{d}(\mathcal{P}_S, \mathcal{P}_T) = 0$.

By the positive definiteness of distance, $\mathbf{d}(\mathcal{P}_S, \mathcal{P}_T) = 0 \Leftrightarrow \mathcal{P}_S = \mathcal{P}_T$, hence P_S and P_T are homologous. □

We propose the homologous component analysis model. This model is comprised of two parts: reconstruction error terms, distribution alignment term. Besides, we impose homologous constraint on two transformations P_S and P_T. This can make them related and hence alleviate negative domain adaptation. The optimization problem is

$$\begin{aligned}\min\ & \|X_S - P_S P_S^T X_S\|_F^2 + \|X_T - P_T P_T^T X_T\|_F^2 \\ & +\lambda\mathrm{MMD}_b^2[\mathcal{F}, P_S^T X_S, P_T^T X_T] \\ \text{s.t. } & P_S, P_T \in \mathcal{S}_D^d, \\ & \mathbf{d}(\mathcal{P}_S, \mathcal{P}_T) = 0.\end{aligned}$$

where

$$\begin{aligned}&\mathrm{MMD}_b^2[\mathcal{F}, P_S^T X_S, P_T^T X_T] \\ &\quad = \|\frac{1}{m_S}\sum_{i=1}^{m_S}\phi(P_S^T x_S^i) - \frac{1}{m_T}\sum_{i=1}^{m_T}\phi(P_T^T x_T^i)\|_{\mathcal{H}}^2,\end{aligned}$$

$\phi : \mathbb{R}^D \to \mathcal{H}$ is a map, and λ is a positive trade-off parameter. Here $\mathcal{H}$ is an universal RKHS. For simplicity, we use Gaussian kernel $k(x_i, x_j) = \exp\left(\frac{-\|x_i - x_j\|_2^2}{2\sigma^2}\right)$ in this paper, where $\sigma > 0$. However, in this case, to solve above optimization problem by Lagrange multiplier method is NP-hard. To still align the distributions between the projected data and make alternative optimization possible, we consider

the empirical kernel map obtained by the decomposition $K = (KK^{-\frac{1}{2}})(K^{-\frac{1}{2}}K)$ [17]. But we consider the instance-based instead of feature-based form which is used in [17]. In other words, we consider side information with respect to both source and target domain where side information means the projected source and target data information. As far as to our knowledge, it is the first time that this information is considered. That is,

$$\mathrm{MMD}_b^2[\mathcal{F}, P_S^T X_S, P_T^T X_T] = Tr(\widehat{X}KMK\widehat{X}^T), \quad (1)$$

where $\widehat{X} = [P_S^T X_S, P_T^T X_T]$ is the instance-based form instead of feature-based form, and $\widehat{X} \in \mathbb{R}^{d\times(m_S+m_T)}$ is the joint side information, and K is the kernel matrix over all the samples, i.e.

$$K = \begin{bmatrix} K_{SS} & K_{ST} \\ K_{TS} & K_{TT} \end{bmatrix} \in \mathbb{R}^{(m_S+m_T)\times(m_S+m_T)}, \quad (2)$$

where $K_{SS} = [k(x_S^i, x_S^j)]$, $K_{ST} = [k(x_S^i, x_T^j)]$, $K_{TS} = K_{ST}$, $K_{TT} = [k(x_T^i, x_T^j)]$, and the (i, j)-entry of M is

$$M_{ij} = \begin{cases} \frac{1}{m_S^2}, & x_i, x_j \in \mathbf{X}_S; \\ \frac{1}{m_T^2}, & x_i, x_j \in \mathbf{X}_T; \\ -\frac{1}{m_S m_T}, & \text{otherwise.} \end{cases}$$

Based on [28], we take into account the conditional distribution adaptation. We expect to minimize the distance between conditional distributions $P^S(y_S|\mathbf{x}_S)$ and $P^T(y_T|\mathbf{x}_T)$. However, the labels in target domain is few and this leads to the difficulty to estimate conditional distribution $P^T(y_T|\mathbf{x}_T)$. As mentioned in [28], we resort to explore non-parameter statistics of $P^S(\mathbf{x}_S|y_S)$ and $P^T(\mathbf{x}_T|y_T)$. We also use the *assumption* in [28] that the pseudo class centroids calculated by pseudo labels may reside not far apart from the true class centroids of target domain. Because there is limited amount of available labels in target domain, *pseduo target labels* which predicted by some supervised classifiers (for example, support vector machine) is used to calculate MMD with respect to each class $c \in \{1, \ldots, C\}$, i.e.

$$\mathrm{MMD}_b^2[\mathcal{F}, P_S^T X_S^{(c)}, P_T^T X_T^{(c)}] = \| \frac{1}{m_S^{(c)}} \sum_{i=1}^{m_S^{(c)}} \phi(P_S^T x_S^{c,i}) - \frac{1}{m_T^{(c)}} \sum_{i=1}^{m_T^{(c)}} \phi(P_T^T x_T^{c,i}) \|_{\mathcal{H}}^2,$$

where $X_S^{(c)} = [x_S^{c,1}, \ldots, x_S^{c,m_S^{(c)}}]$ is the c-th class instance matrix in the source domain and $X_T^{(c)} = [x_T^{c,1}, \ldots, x_T^{c,m_T^{(c)}}]$ is the c-th *pseudo class* (i.e. predicted class on unlabeled target samples via a classifier) instance matrix in the target domain respectively. Similarly to (1), we use the following estimation

$$\mathrm{MMD}_b^2[\mathcal{F}, P_S^T X_S^{(c)}, P_T^T X_T^{(c)}] = Tr(\widehat{X}^{(c)} K^{(c)} M^{(c)} K^{(c)} (\widehat{X}^{(c)})^T), \quad (3)$$

where $\widehat{X}^{(c)} = [P_S^T X_S^{(c)}, P_T^T X_T^{(c)}]$,

$$(M^{(c)})_{ij} = \begin{cases} \frac{1}{(m_S^{(c)})^2}, & x_i, x_j \in \mathbf{X}_S^{(c)}; \\ \frac{1}{(m_T^{(c)})^2}, & x_i, x_j \in \mathbf{X}_T^{(c)}; \\ -\frac{1}{m_S^{(c)} m_T^{(c)}}, & x_i \in \mathbf{X}_S^{(c)} \text{ and } x_j \in \mathbf{X}_T^{(c)}; \\ -\frac{1}{m_S^{(c)} m_T^{(c)}}, & x_i \in \mathbf{X}_T^{(c)} \text{ and } x_j \in \mathbf{X}_S^{(c)}; \\ 0, & \text{otherwise,} \end{cases}$$

$\mathbf{X}_S^{(c)} = \{x_S^{c,1}, \ldots, x_S^{c,m_S^{(c)}}\}$, $\mathbf{X}_T^{(c)} = \{x_T^{c,1}, \ldots, x_T^{c,m_T^{(c)}}\}$, and $K^{(c)}$ is the kernel matrix defined on $\mathbf{X}_S^{(c)} \cup \mathbf{X}_T^{(c)}$.

Combining (1) with (3) leads to joint distribution adaptation, and MMD is calculated as

$$\begin{aligned} \mathrm{MMD}^2 &= \mathrm{MMD}_b^2[\mathcal{F}, P_S^T X_S, P_T^T X_T] \\ &\quad + \sum_{c=1}^{C} \mathrm{MMD}_b^2[\mathcal{F}, P_S^T X_S^{(c)}, P_T^T X_T^{(c)}] \\ &= Tr(\widehat{X}KMK\widehat{X}^T) \\ &\quad + \sum_{c=1}^{C} Tr(\widehat{X}^{(c)} K^{(c)} M^{(c)} K^{(c)} (\widehat{X}^{(c)})^T). \end{aligned} \quad (4)$$

Our model is

$$\begin{aligned} &\min \ \|X_S - P_S P_S^T X_S\|_F^2 + \|X_T - P_T P_T^T X_T\|_F^2 + \lambda \mathrm{MMD}^2 \\ &\text{s.t. } P_S, \quad P_T \in \mathcal{S}_D^d, \\ &\quad \mathbf{d}(\mathcal{P}_S, \mathcal{P}_T) = 0. \end{aligned}$$

We change the constraint $\mathbf{d}(\mathcal{P}_S, \mathcal{P}_T) = 0$ into a penalty term $\mu\mathbf{d}(\mathcal{P}_S, \mathcal{P}_T)$, where $\mu > 0$. Our optimization problem becomes

$$\begin{aligned} \min \ & \|X_S - P_S P_S^T X_S\|_F^2 + \|X_T - P_T P_T^T X_T\|_F^2 \\ & + \lambda \mathrm{MMD}^2 + \mu \mathbf{d}(\mathcal{P}_S, \mathcal{P}_T)^2 \\ \text{s.t.} \quad & P_S, \quad P_T \in \mathcal{S}_D^d. \end{aligned} \quad (5)$$

We give a reformulation of our model (5).

Proposition 4:

$$\begin{aligned} \max \ & Tr(P_S^T X_S X_S^T P_S) + Tr(P_T^T X_T X_T^T P_T) \\ & - \lambda (Tr(\widehat{X}KMK\widehat{X}^T) \\ & + \sum_{c=1}^{C} Tr(\widehat{X}^{(c)} K^{(c)} M^{(c)} K^{(c)} (\widehat{X}^{(c)})^T)) \\ & + \mu Tr(P_S^T P_T P_T^T P_S) \\ \text{s.t.} \ & P_S, \quad P_T \in \mathcal{S}_D^d. \end{aligned} \quad (6)$$

We leave the proof in the Appendix A.

For the optimization, we provide its main procedures in Appendix C. As summarization, we display the complete algorithm procedure in Algorithm 1. The given optimization procedure produces a decreasing sequence

Algorithm 1 Homologous Component Analysis

Input: Source data X_S, source labels y_S, target data X_T, target labels y_{Tl}, $\lambda > 0$, and $\mu > 0$.
Initialize: $P_S^0, P_T^0 \in \mathcal{S}_D^d$.
1. Compute kernel matrix K by (2) and matrix L by (8);
2. Calculate pseudo labels $\widetilde{Y}_{Tu}$ on unlabeled target smaples $\mathbf{X}_{Tu}$ via a classifier which is well trained on labeled samples $\{\mathbf{X}_S \cup \mathbf{X}_{Tl}, \mathbf{y}_S \cup \mathbf{y}_{Tl}\}$.
while not converge **do**
  while not converge **do**
    3. Fix P_S, and solve

$$\min_{P_T \in \mathcal{S}_D^d} -f(P_S, P_T)$$

    by iteration algorithm in the context of Stiefel manifold with (11-15);
    4. Fix P_T, and solve

$$\min_{P_S \in \mathcal{S}_D^d} -f(P_S, P_T)$$

    by iteration algorithm in the context of Stiefel manifold with (10) and (16-19);
  end while
5. Project target data and produce pseudo labels based on classifier which is well trained on projected source data.
end while
Output: Final classifier.

$\{-f(P_S^{(k)}, P_T^{(k)})\}_{k\geq 0}$, i.e. $-f(P_S^{(0)}, P_T^{(0)}) \geq -f(P_S^{(0)}, P_T^{(1)}) \geq -f(P_S^{(1)}, P_T^{(1)}) \geq -f(P_S^{(1)}, P_T^{(2)}) \geq \cdots \geq -f(P_S^{(k)}, P_T^{(k)}) \geq -f(P_S^{(k)}, P_T^{(k+1)}) \geq \cdots$. Hence a local minimizer of $-f(P_S, P_T)$ will be attained.

Besides, we provide complexity analysis. The basic operations are matrix multiplication and matrix inversion from which we can derive complexity. Assume that pseudo labels updates are performed T_{out} iterations, alternating procedure for two transformation are performed T_{alt} iterations and each transformation updates are performed T_{in} iterations. Computing L in Eq. (8) needs $\mathcal{O}((m_S + m_T)^3)$. Next we analyze complexity for updating P_T one iteration (i.e. step (a) in Appendix C.). Eq. (12) needs $\mathcal{O}(D^2 m_T d + D^2 d m_T m_S + D^2 d^2)$. In Eq. (13), its computation needs $\mathcal{O}(Dd^2 + d^3)$ by [21]. Hence it needs $\mathcal{O}(D^2 m_T d + D^2 d m_T m_S + D^2 d^2 + d^3)$ for updating P_T. Similarly, it needs $\mathcal{O}(D^2 m_S d + D^2 d m_T m_S + D^2 d^2 + d^3)$ for updating P_S. The overall complexity of the proposed algorithm is $\mathcal{O}(((m_S + m_T)^3 + D^2(m_S + m_T)d + D^2 d m_T m_S + D^2 d^2 + d^3) T_{in} T_{alt} T_{out})$.

VI. Generalization Bound Analysis

Reference [29] gives a correct statement of the result about the generalization error bound in [16]. Based on their work, we give the following theorem. For simplicity, let labels belong to $\{0, 1\}$.

Theorem 1: Let H be a hypothesis space of VC-dimension d and $\{(x_i, y_i)\}_{i=1}^{m_S}$ be i.i.d samples drawn from $\Pr_S(\mathbf{x}, \mathbf{y})$ and $\{(x_{m_s+j}, y_{m_S+j})\}_{j=1}^{m_{Tl}}$ be i.i.d samples drawn from $\Pr_T(\mathbf{x}, \mathbf{y})$, where $m_{Tl} \ll m_S$. Let $L : \mathbb{R} \times \mathbb{R} \longrightarrow [0, 1]$ be a loss function, $\wp_S$ and $\wp_T$ be two transformations, $d \ll m_S$ and $d_{H\triangle H}$ be the distance induced by the symmetric difference hypothesis space. Let $m = m_S + m_{Tl}$. With probability at least $1 - \delta - \frac{m_{Tl}}{m}$(over the choice of the samples), for every $h \in H$,

$$\begin{aligned}
&\mathbf{E}_{(\wp_T(\mathbf{x}),\mathbf{y})\sim \Pr_T(\wp_T(\mathbf{x}),\mathbf{y})} L(h(\wp_T(\mathbf{x})), \mathbf{y}) \\
&\leq \frac{1}{m}\left(\sum_{i=1}^{m_S} L(h(\wp_S(x_i)), y_i) + \sum_{i=1}^{m_{Tl}} L(h(\wp_T(x_{m_S+i})), y_{m_S+i})\right) \\
&\quad + \sqrt{\frac{d(\log(2m_S/d) + 1) - \log(\delta/4)}{m_S}} \\
&\quad + \frac{1}{2} d_{H\triangle H}(\Pr_S(\wp_S(\mathbf{x}), \mathbf{y}), \Pr_T(\wp_T(\mathbf{x}), \mathbf{y})) + \gamma, \qquad (7)
\end{aligned}$$

where

$$\begin{aligned}
\gamma = \min_{h\in H} &\mathbf{E}_{(\wp_S(\mathbf{x}),\mathbf{y})\sim \Pr_S(\wp_S(\mathbf{x}),\mathbf{y})} L(h(\wp_S(\mathbf{x})), \mathbf{y}) \\
&+ \mathbf{E}_{(\wp_T(\mathbf{x}),\mathbf{y})\sim \Pr_T(\wp_T(\mathbf{x}),\mathbf{y})} L(h(\wp_T(\mathbf{x})), \mathbf{y}).
\end{aligned}$$

We place the proof of theorem 1 in the Appendix B. Semi-supervised domain adaptation is the case where there is few labels are available in target domain. Nevertheless, this is empirically effective. This theorem provides a generalized error bound for semi-supervised domain adaptation for the first time which is related to the ratio of target labeled samples and source labeled samples, and includes the term which has the same structure over source and target labeled samples as empirical risk minimization. In our experiment settings, the ratio m_{Tl}/m is small.

In our proposed method, the MMD item aims to align joint distributions so as to make $d_{H\triangle H} \approx 0$, and minimize the average loss over the labeled samples from different domains in the classification stage. So we indeed reduce the right part in (7).

VII. Experiments

A. Synthetic Data

In this section, we conduct an experiment on generated toy data for homologous component analysis. We generate 100 source samples and 110 target samples i.i.d drawn from Gaussian distribution whose mean is μ_S and covariance is Σ_S and Gaussian distribution whose mean is μ_T and covariance is Σ_T respectively, where

$$\mu_S = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}, \quad \Sigma_S = \begin{bmatrix} 5 & 0 & 0 \\ 0 & 7 & 0 \\ 0 & 0 & 0.5 \end{bmatrix}$$

and

$$\mu_T = \begin{bmatrix} 4 \\ 4 \\ 4 \end{bmatrix}, \quad \Sigma_T = \begin{bmatrix} 8 & 1 & 0 \\ 1 & 7 & 0 \\ 0 & 0 & 0.5 \end{bmatrix}.$$

In source data, 50 samples form 1st to 50th are in class 1 and others of class 2. In target data, 5 samples for 1st to 5th are in class 1 and 5 samples from 56th to 60th are in class two. Labels of other samples in target data are not available. Source and target domain are shown in Fig 2 and Fig 3 respectively.

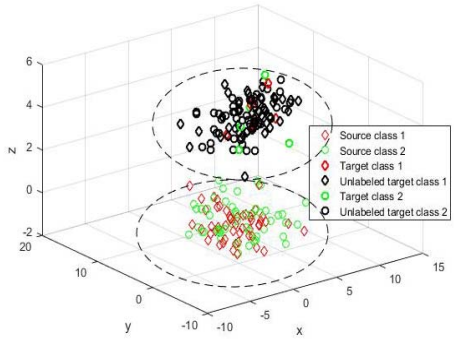


Fig. 2. 3D illustration for source and target domain.

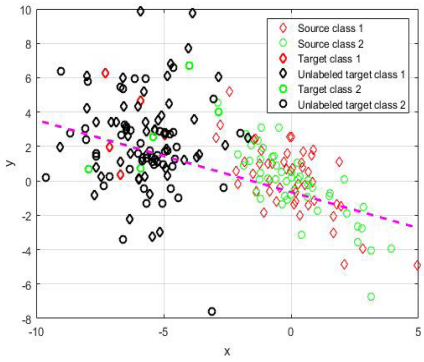


Fig. 3. 2D illustration for projected source and target domain.

The marginal distributions of source and target domain are quite different. In Fig 2, we can explicitly see that data in source and target domain are separated. A good classification decision for source domain may badly fit the target domain. In Fig 3, we find that the variance of both projected source and target domain becomes large when compared to unprojected case. The distributions is approximately aligned. The two resulted optimal homologous transformations makes the projected source and target data similar and hence help to transfer useful knowledge form source domain to target domain. Then we use SVM classifier with linear kernel to do prediction. Pink broken line in Fig 3 represents the decision boundary. Therefore, toy data experiment verifies the effectiveness of the proposed method. Let P_S^* and P_T^* denotes the two resulted optimal transformations respectively. $\mathcal{P}_S^*$ and $\mathcal{P}_T^*$ are the spaces spanned by the columns of P_S^* and P_T^* respectively. In this experiment, the Frobenius distance $\|P_S^* - P_T^*\|_F = 24.15$ and the Chordal distance $\mathbf{d}(\mathcal{P}_S^*, \mathcal{P}_T^*) = 1.51 \times 10^{-3}$. Numerical evidence shows that the two resulted optimal transformations are really different. Hence we avoid the case where we seemingly use two transformations by but actually only use one transformation when solved by optimization problem.

B. Object Recognition

1) Datasets: We use the Office dataset [30] and Caltech10 [18] dataset. Two datasets contain four domains in all to evaluate all domain adaptation methods. Each domains contain 10 common object classes. The Office dataset consists of images from Webcam(W), DSLR(D) images and Amazon(A) images. We represent Caltech10 as C for simplicity. We use the image representations as in [18] for Office and SURF features encoded with a visual dictionary of 800 words for Caltech10 datasets. we show datasets description in Table I.

TABLE I
DATASETS DESCRIPTION AND SAMPLES SELECTION

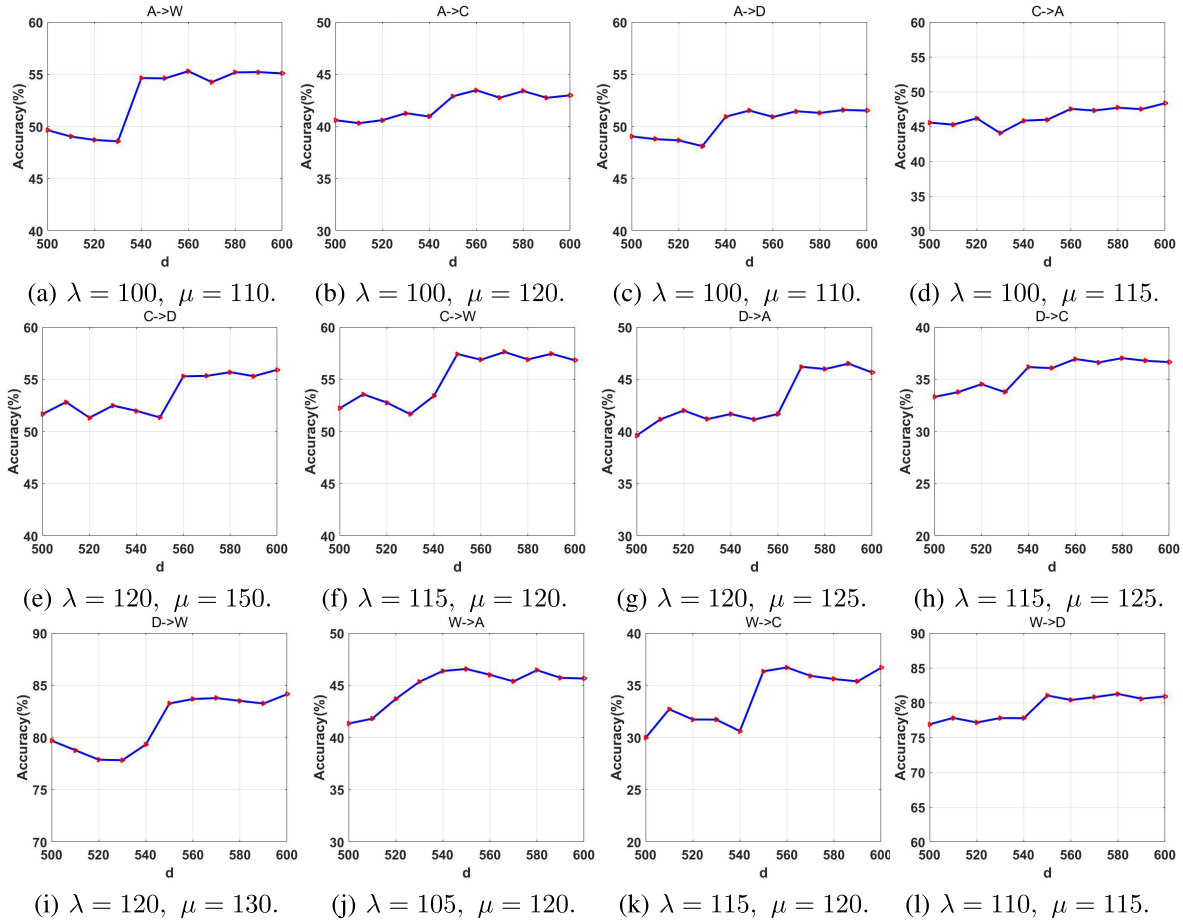
Domain	♯ class	♯ dimension	♯ Samples	n_S/c	n_T/c
Webcam	10	800	295	20	3
DSLR	10	800	157	8	3
Amazon	10	800	958	20	3
Caltech10	10	800	1123	20	3

2) Experiment Setup: We strictly follow the configuration in [18]. Randomly Select 20 samples per class from Amazon, and 8 samples per class from DSLR, Webcam and Caltech if they are source domains, while Rondomly select 3 samples per class if they are target domains, and the rest data in target domain is used for testing. We shown samples selection in Table I, in which n_S/c and n_T/c mean that the number of samples randomly selected from each class when the specific domain is regard as source domain and target domain respectively. For every cross-domain pair, we conduct experiments for 20 trails and the average accuracy is reported.

3) Baseline Methods:

- **SVM**. We perform SVM [31] with linear kernel on labeled samples from the source and target domain and predict the unlabeled samples from the target domain without domain adaptation.
- **SGF**. SGF [33] considers intermediate representations of source domains and target domain by viewing the generative subspaces created from these domains as points on the Grassmanian manifold, and sampling points along the geodesic between them to obtain subspaces which helps domain adaptation.
- **SA**. SA [5] seeks a domain adaptation solution by learning a mapping function which aligns the source subspace with the target one.
- **GFK**. GFK [18] is a kernel-based method, which models domain shift integrating an infinite number of subspaces that characterize changes in geometric and statistical properties from the source to the target domain.
- **CORAL**. CORAL [3] minimizes domain shift by aligning the second-order statistics of source and target distributions.
- **LSDT**. LSDT [2] is a joint learning model combining the sparse coding and subspace representation.
- **JGSA**. JGSA [4] reduces the shift between the source and target domain in both statistical and geometric sense. JGSA learns two coupled projections that project the source domain.
- **FLAMM**. FLAMM [34] is a feature learning algorithm used in conjunction with linear classifiers for domain adaptation.

4) Implementation Details: Experimental results are reported in Table I. In the experiments, we set $\lambda = 0.5$ and μ is tuned from the list $\{100, 150, \ldots, 750, 800\}$. The initial value P_S^0 of the transformation P_S is a randomly generated



(a) $\lambda = 100,\ \mu = 110.$ (b) $\lambda = 100,\ \mu = 120.$ (c) $\lambda = 100,\ \mu = 110.$ (d) $\lambda = 100,\ \mu = 115.$

(e) $\lambda = 120,\ \mu = 150.$ (f) $\lambda = 115,\ \mu = 120.$ (g) $\lambda = 120,\ \mu = 125.$ (h) $\lambda = 115,\ \mu = 125.$

(i) $\lambda = 120,\ \mu = 130.$ (j) $\lambda = 105,\ \mu = 120.$ (k) $\lambda = 115,\ \mu = 120.$ (l) $\lambda = 110,\ \mu = 115.$

Fig. 4. Sensitivity curve of dimensionality d with given λ and μ.

orthogonal matrix whose all entries is in the interval $[1, 10]$ and we set the initial value P_T^0 of the transformation P_T as $P_T^0 = P_S^0$. The kernel width parameter of Gaussian kernel in MMD term is selected from $\{75, 85, 95, \ldots, 125\}$, while in SVM classifier from the list $\{100, 125, 150, \ldots, 300\}$.

For SGF, SVM with linear kernel in the classification stage. For GFK, the dimensionality reduction is performed by same strategy in [18], and we use SVM classifier with constructed kernel in [18] in the classification stage. For CORAL, We set parameter $\lambda = 0.5$ in perturbed empirical covariance matrix for each source and target domain, and use linear SVM classifier in the classification stage. For LSDT, all parameters setting follows [2]. For JGSA, we fix $\lambda = 1$, $\mu = 0.5$ and β is tuned over the list $\{0.1, 0.2, \ldots, 0.5\}$. We use SVM classifier with linear kernel in the classification. For FLAMM, parameters setting follows [34].

5) Parameter Sensitivity Analysis: We give empirical parameters sensitivity analysis.

• **The initial value P_S^0 and P_T^0.** The initial values P_S^0 and P_T^0 are randomly generated. We empirically find that if P_S^0 and P_T^0 is far, that is, $\|P_S^0 - P_T^0\|_F$ is large, the stability of the proposed algorithm is poor. But if they are close, the performance is usually stable. Hence, we empirically set $P_S^0 = P_T^0$ in our experiments.

• **The regularizers λ, μ.** When λ is large, overfitting occurs. Hence we choose $\lambda \in (0, 150]$. The larger value of μ drives two resulted optimal transformations to obey homologous principle. We choose $\mu \in [100, 800]$.

• **Dimensionality d.** The feature dimension is 800. Hence $d \in \{1, 2, \ldots, 800\}$. We show the performance of stability for dimensionality $d \in \{500, 510, \ldots, 600\}$. We plot classification accuracy w.r.t. dimensionality d in Fig 4 (a)-(l). From Fig 4 (a)-(l), we find that the performance of the proposed algorithm becomes stable as dimensionality d are large.

6) Analysis on Resulted Optimal Transformations: To avoid the case like in [19] where we seemingly use two transformations but actually just use only one transformation solved by optimization, we need to check the Frobenius distance between two transformations. We report the record of chordal distance $\mathbf{d}(\mathcal{P}_S, \mathcal{P}_T)$ and Frobenius distance $\|P_S - P_T\|_F$ in Table III and Table IV. We find that the two resulted optimal transformations approximately obey the homologous principle and they are really different in every recognition task. This signifies that proposed method is different from works mentioned in related works in section II on the object recognition experiment. Two transformation can help to transfer more useful knowledge form source domain to target domain than actually one common transformation in experiments. Experimental results which outperform previous works using only one transformation may reveal the effectiveness of homology.

7) The Effectiveness of Homology: In Table II, performance of the proposed method with homologous constraint

TABLE II
RECOGNITION ACCURACY (%) OF SINGLE SOURCE DOMAIN ADAPTATION

Directions	Compared methods								Our method	
Source → Target	SVM	SGF	SA	GFK	CORAL	LSDT	JGSA	FLAMM	HCA ($\mu = 0$)	HCA ($\mu > 0$)
A→W	51.8±0.5	54.0±0.2	52.5±1.1	56.7±1.2	52.3±0.8	56.8±0.3	56.5±0.7	50.4±0.3	53.2±0.7	**56.9±0.4**
A→C	32.1±0.1	36.9±0.9	40.2±1.0	39.1±0.9	31.7±1.1	41.5±0.1	40.8±0.5	35.5±1.1	41.6±0.4	**43.8±0.3**
A→D	45.9±0.5	46.5±0.4	47.4±1.0	50.8±0.2	46.5±0.7	51.7±0.7	50.9±0.5	46.2±0.7	48.6±0.8	**51.8±0.7**
C→A	45.2±0.7	41.7±1.0	46.5±0.8	46.8±0.7	47.5±0.5	46.9±1.2	45.4±1.2	47.1±0.2	46.7±0.5	**48.4±0.7**
C→D	55.8±0.4	49.8±0.5	49.8±0.7	55.3±1.0	45.7±0.9	55.9±0.5	54.7±0.4	50.7±0.4	53.4±0.4	**56.1±0.8**
C→W	50.3±1.0	54.1±0.1	52.7±1.5	56.9±0.6	47.9±0.4	56.8±0.1	57.1±0.6	52.5±0.6	55.1±0.4	**57.7±0.5**
D→A	40.2±0.3	44.5±0.3	43.1±1.3	46.1±0.4	42.3±1.2	45.8±0.7	44.8±0.9	44.9±1.3	44.8±0.5	**46.5±0.6**
D→C	31.1±1.2	32.3±1.4	35.5±0.9	33.5±0.8	29.5±0.7	36.8±1.3	36.1±1.2	32.5±0.4	34.6±0.8	**37.1±0.2**
D→W	55.1±0.9	78.4±0.7	82.3±1.2	80.1±1.3	83.2±0.4	82.7±1.4	81.2±0.7	79.4±0.6	81.7±0.8	**84.2±0.7**
W→A	45.6±0.4	42.8±0.5	43.0±2.0	45.8±0.4	37.9±0.2	45.9±0.8	46.1±1.4	45.1±0.7	43.9±0.7	**46.7±0.9**
W→C	31.2±0.8	32.6±1.3	36.2±0.5	31.7±0.7	30.2±0.9	36.3±0.2	32.5±0.6	31.3±0.6	35.4±1.0	**37.1±0.2**
W→D	55.1±1.1	78.8±0.2	78.8±1.2	76.2±0.5	81.9±0.1	76.2±0.5	77.8±0.4	78.4±1.2	80.3±0.6	**82.1±0.4**

TABLE III
DISTANCE RECORD

Distance type	A→W	A→C	A→D	C→A	C→D	C→W
Chordal distance ($\times 10^{-4}$)	4.21	5.73	1.47	1.13	2.45	3.71
Frobenius distance	37.45	22.65	31.58	24.81	19.54	24.18

TABLE IV
DISTANCE RECORD

Distance type	D→W	D→W	D→W	W→A	W→C	W→D
Chordal distance ($\times 10^{-4}$)	2.23	3.41	5.92	2.34	1.92	4.27
Frobenius distance	19.76	40.09	17.25	28.84	37.15	32.71

(i.e. $\mu > 0$) is stronger than it without homologous constraint (i.e. $\mu = 0$). Hence homology may help to extract related and transferable components from source and target domain.

8) Comparison With SA [5]: Both our proposed approach and SA belong to subspace based domain adaptation approaches. Both of these approaches involve extracting components from source and target samples and simultaneously makes extracted components from source and target domain related in some way such that transferability is considered. However, there exist some differences between them: (1) Our proposed approach establishes the relatedness between extracted components from source and target domain via homoloy while SA establishes the relatedness via learning a linear transformation function that aligns the source subspace coordinate system to the target one. (2) Our proposed approach explicitly considers distribution alignment for extracted components from source and target domain while SA doesn't consider it. From Table II, we can see that the performance of our proposed approach outperforms SA on all tasks. This may be attributed to the homology and extended MMD Eq. (1).

C. Handwritten Digits Data

1) Datasets: We use three handwritten digit datasets to conduct experiments. The MNIST [35] dataset has 70,000 samples including 60,000 training samples and 10,000 test samples with each image size of 28×28. The USPS [35] dataset consists of 7,291 training images and 2,007 test images of size 16 × 16. The SEMEION [36] dataset includes 2593 images of size 16×16. We normalize and z-score the SEMEION data. The size of digit images in MNIST is resized into 16×16 to keep dimension consistent.

TABLE V
DATASET DESCRIPTION AND SAMPLES SELECTION

Domain	♯ class	♯ dimension	♯ Samples	n_S/c	n_T/c
MNIST	10	256	1800	100	10
USPS	10	256	2000	100	10
SEMEION	10	256	1593	100	10

2) Experiment Setup: In our experiment, we adopt the experiment setting in [2] where 100 samples per class are selected from MNIST, USPS and SEMEION if they are source domains, while 10 samples per class are selected if they are target domains. We perform cross-domain adaptation on three mentioned datasets. We have six pairs of cross-domain tasks in all. We describe them for source to target domain in order. They are (1) MNIST to USPS, (2) MNIST to SEMEION, (3) USPS to MNIST, (4) USPS to SEMEION, (5) SEMEION to MNIST and (6) SEMEION to USPS respectively. For every cross-domain pair, we conduct experiment for 10 trails and the average accuracy is reported.

3) Baseline Methods: To verify the effectiveness of the proposed methods, we compare them with existing methods:

- **SVM**. We perform SVM [31] on labeled samples from the source and target domain and predict the unlabeled samples from the target domain without domain adaptation.
- **SA**. SA [5] seeks a domain adaptation solution by learning a mapping function which aligns the source subspace with the target one.
- **TCA**. TCA [17] method tries to learn some transfer components cross domains in a reproducing kernel Hilbert space via maximum mean discrepancy.
- **GFK**. GFK [18] models domain shift by integrating an infinite number of subspaces that characterize changes in geometric and statistical properties from the source to the target domain.

TABLE VI
RECOGNITION ACCURACY (%) IN HANDWRITING DIGIT TASKS

Domains		Compared methods								Our method	
Source	Target	SVM	SA	TCA	GFK	CORAL	VDA	JGSA	LSDT	HCA ($\mu = 0$)	HCA ($\mu > 0$)
MNIST	USPS	72.1±0.3	76.3±1.3	66.4±0.7	81.2±0.2	51.2±1.0	75.6±0.6	78.2±0.9	77.4±0.2	80.3±0.9	**82.5±0.2**
USPS	MNIST	68.5±0.2	56.6±1.6	57.1±0.2	74.5±0.5	64.8±0.6	69.3±0.4	68.9±1.2	69.2±0.9	73.2±0.5	**75.0±0.3**
MNIST	SEMEION	50.2±0.1	45.4±2.2	50.1±0.5	69.8±1.2	10.9±0.7	66.5±0.7	65.1±0.8	67.6±0.7	68.5±0.5	**71.2±0.2**
SEMEION	MNIST	65.5±0.9	52.9±2.3	55.6±0.2	72.3±0.6	25.6±0.4	68.8±0.2	69.1±0.9	68.4±0.4	70.2±0.6	**74.1±0.7**
USPS	SEMEION	63.8±0.8	60.6±1.2	57.9±1.3	75.8±0.8	48.8±0.6	65.5±1.1	66.4±1.2	66.8±0.6	73.1±0.4	**76.3±0.4**
SEMEION	USPS	80.7±0.5	80.7±1.8	69.2±0.9	81.7±0.8	52.4±0.3	85.7±0.29	83.5±1.4	83.2±1.2	81.8±1.1	**85.8±0.5**

- **CORAL**. CORAL [3] minimizes domain shift by aligning the second-order statistics of source and target distributions.
- **LSDT**. LSDT [2] is a joint learning model combining the sparse coding and subspace representation. KLSDT extends the LSDT to a kernel-based framework for tacking non-linear subspace shifts in reproducing kernel Hilbert space.
- **JGSA**. JGSA [4] reduces the shift between the source and target domain in both statistical and geometric sense. JGSA learns two coupled projections that project the source domain and target domain data into low-dimensional subspaces where the geometrical shift and distribution shift are reduced simultaneously. But it requires the coupled projections are very closed in the sense of Frobebius norm. Hence it produces two almost equal projections. Our proposed methods make essential difference.
- **VDA**. VDA [20] tries to reduce the difference between joint distribution and conditional distribution across domains.

4) Implementation Details: All experimental results are reported in Table VI. For the proposed methods, the initial value of orthogonal matrix P_S^0 is randomly generated orthogonal matrix whose all entries is in the interval $[1, 5]$ and we empirically set $P_T^0 = P_S^0$. The kernel width parameter of Gaussian kernel in MMD item is selected from $\{75, 85, 95, \ldots, 125\}$, while in SVM classifier from the list $\{100, 125, 150, \ldots, 300\}$. The trade-off parameters λ and μ are selected from the list $\{100, 105, \ldots, 150\}$ and the list $\{105, 110, \ldots, 155\}$ respectively. For TCA, linear SVM in the classification stage. For GFK, the dimensionality parameter is determined by same strategy in [18] and we use SVM classifier with constructed kernel in [18] in the classification stage. For CORAL, We set parameter $\lambda = 0.5$ in perturbed empirical covariance matrix for each source and target domain, and use linear SVM classifier in the classification stage. For LSDT, parameters setting follows [2]. For JGSA, we fix $\lambda = 2$, $\mu = 1$, and β is tuned over the list $\{0.1, 0.2, \ldots, 0.5\}$. SVM with Gaussian kernel in the classification stage, whose width is tuned over the list $\{100, 110, \ldots, 150\}$. For VDA, we follow the parameters setting in [20].

5) Parameter Sensitivity Analysis: We conduct empirical parameters sensitivity analysis.

• **The initial value P_S^0 and P_T^0.**

In our experiments, we find that the final classification performance depends on the initial values of two transformations P_S and P_T. If they are closed in the sense of Frobenius norm, the final classification performance is relatively well. However, if they are far, the final classification is usually not good enough. The main cause is that different initial values may leads to different local minimizers. Not all local minimizers can construct good classifiers. To set them equal values is empirically to get a good result.

TABLE VII
NUMERICAL RECORD OF DISTANCES

Distance Type	Chordal Distance ($\times 10^{-3}$)	Frobenius Distance
MNIST → USPS	3.54	15.09
USPS → MNIST	2.78	21.35
MNIST → SEMEION	1.79	24.57
SEMEION → MNIST	1.48	19.86
USPS → SEMEION	1.52	27.46
SEMEION → USPS	3.40	25.67

• **Regularizers** λ, μ. When λ and μ are large, the MMD term and chordal distance is small. When $\lambda \to 0$, MMD term may be very large and overfitting occurs. When $\mu \to 0$, chordal distance may be very large and in this case the our model produce two transformations but they are not homologous, overfitting also occurs. We empirically set regularizers $\lambda \in \{75, 85, \ldots, 115\}$ and $\mu \in \{90, 95, \ldots, 120\}$ in our experiment respectively.

• **Dimensionality** d. The feature dimension is 256. Hence $d \in \{1, 2, \ldots, 256\}$. We empirically find that the proposed algorithm is effective for large dimensionality. We plot classification accuracy w.r.t dimensionality d in Fig 5 (a)-(f) and choose $d \in \{100, 105, \ldots, 150\}$.

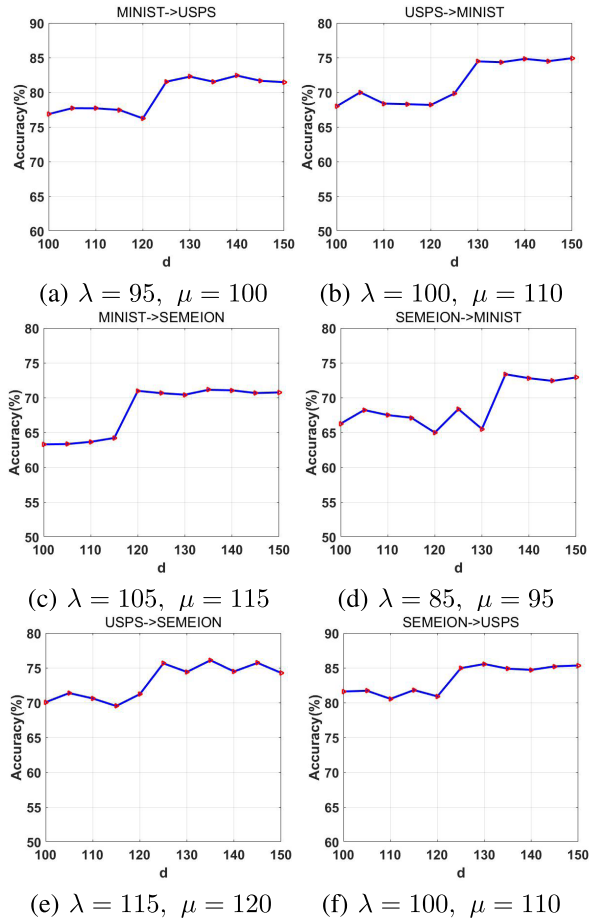
6) Analysis on Resulted Optimal Transformations: To avoid the case like in [19] where we seemingly use two transformations but actually just use only one transformation solved by optimization, we need to check the Frobenius distance between two transformations. We report the numerical record of chordal distance $\mathbf{d}(\mathcal{P}_S, \mathcal{P}_T)$ and Frobenius distance $\|P_S - P_T\|_F$ in Table VII.

The numerical evidence signifies that the two resulted optimal transformations are really different and homologous in our trained models. This signifies that the proposed algorithm is different from the mentioned works in section II on handwritten digits recognition experiment. Experimental results which outperform previous works using only one transformation are partially attributed to the effectiveness of homology.

7) The Effectiveness of Homology: In Table VI, we find that performance of the proposed method with homologous

TABLE VIII
RECOGNITION ACCURACY (%) ON OFFICE-HOME DATASET WITH RESNET-50 FEATURES

Method	Tasks											
	Ar→Cl	Ar→Pr	Ar→Rw	Cl→Ar	Cl→Pr	Cl→Rw	Pr→Ar	Pr→Cl	Pr→Rw	Rw→Ar	Rw→Cl	Rw→Pr
ResNet-50	34.9	50.0	58.0	37.4	41.9	46.2	38.5	31.2	60.4	53.9	41.2	59.9
DAN	43.6	57.0	67.9	45.8	56.5	60.4	44.0	43.6	67.7	63.1	51.5	74.3
DANN	45.6	59.3	70.1	47.0	58.5	60.9	46.1	43.7	68.5	63.2	51.8	76.8
JAN	45.9	61.2	68.9	50.4	59.7	61.0	45.8	43.4	70.3	63.9	52.4	76.8
CDAN	49.0	69.3	74.5	**54.4**	66.0	**68.4**	55.6	48.3	75.9	**68.4**	55.4	80.5
SA	49.5	68.6	74.8	49.6	61.8	64.1	51.8	44.4	72.2	61.8	49.7	76.5
SVM	50.7	69.0	75.6	49.9	63.0	64.2	50.5	44.6	73.2	62.8	50.4	78.0
HCA ($\mu = 0$)	49.2	67.8	74.5	49.3	63.2	64.7	50.1	46.6	72.9	60.2	52.8	77.4
HCA ($\mu > 0$)	**51.2**	**69.8**	**76.2**	53.7	**67.1**	66.5	**55.9**	**49.0**	**76.6**	67.6	**55.7**	**81.1**



(a) $\lambda = 95,\ \mu = 100$ (b) $\lambda = 100,\ \mu = 110$
(c) $\lambda = 105,\ \mu = 115$ (d) $\lambda = 85,\ \mu = 95$
(e) $\lambda = 115,\ \mu = 120$ (f) $\lambda = 100,\ \mu = 110$

Fig. 5. Sensitivity curve of dimensionality d with given λ and μ.

constraint (i.e. $\mu > 0$) is stronger than it without homologous constraint (i.e. $\mu = 0$). This may be attributed to the fact that homology helps to extract related and transferable components from source and target domain. We also notice that when $\mu = 0$, performance of HCA is weaker than SVM on some tasks, which means that there exits negative transfer phenomenon, i.e. performance degrades after domain adaptation. However, the performance is enhanced when we take homologous constraint into account. Hence homology is also helpful for alleviating negative transfer.

8) Comparison With SA [5]: From Table VI, we can see that the performance of our proposed approach outperforms SA on all tasks. This may be attributed to the homology and extended MMD.

TABLE IX
RECOGNITION ACCURACY (%) ON VISDA DATASET WITH RESNET-50 FEATURES

Method	Synthetic→Real
JAN	61.6
GTA	**69.5**
CDAN	66.8
SVM	50.0
SA	55.6
JGSA	61.4
HCA ($\mu = 0$)	59.3
HCA ($\mu > 0$)	62.5

D. Large Datasets

We evaluate the proposed approach on two large datasets: Office-Home dataset [58] and VisDA dataset.[1] The Office-Home dataset contains 15,500 images in 65 object classes which form four extremely dissimilar domains: Artistic images (Ar), Clip Art (Cl), Product images (Pr), and Real-World images (Rw). VisDA dataset is a visual challenging synthetic-to-real dataset which contains two very distinct domains: Synthetic, renderings of 3D models from different angles and with different lightning conditions; Real, natural images. It consists of over 280,000 images across 12 classes in the training, validation and test domains. We follow the standard domain adaptation protocol in [62].

On Office-Home dataset, we compare the proposed approach with state-of-the-art deep domain adaptation ones: ResNet-50 [59], DAN [55], DANN [60], JAN [61], CDAN [62]. We directly cite published experimental results of these approaches in [62]. We also compare our proposed approach with SA [5]. Experimental results are reported in Table VIII.

On VisDA dataset, we compare the proposed approach with state-of-the-art domain adaptation ones: JAN [61], GTA [63], CDAN [62], SA [5] and JGSA [4]. JAN [61], GTA [63] and CDAN [62] are deep learning based domain approaches. We directly cite published experimental results of these approaches in [62]. JGSA is a recent shallow domain adaptation approach. Experimental results are reported in Table IX.

[1] http://ai.bu.edu/visda-2017/

TABLE X
NUMERICAL RECORD OF DISTANCES

Metric	Tasks												
	Ar→Cl	Ar→Pr	Ar→Rw	Cl→Ar	Cl→Pr	Cl→Rw	Pr→Ar	Pr→Cl	Pr→Rw	Rw→Ar	Rw→Cl	Rw→Pr	Synthetic→Real
Chordal Distance ($\times 10^{-3}$)	10.54	7.19	5.12	8.72	9.87	3.42	6.47	13.27	4.98	10.25	7.88	11.29	6.17
Frobenius Distance	30.61	21.35	28.64	33.86	45.27	39.61	26.75	19.42	38.71	46.21	34.86	40.27	60.34

1) Parameters Setting: We empirically set $d = 1500$ on Office-Home dataset. We use PCA over all samples from source and target domain to initialize P_T^0 and set $P_S^0 = P_T^0$. In all experiments on Office-Home dataset, λ and μ are tuned from the list $\{30, 40, \cdots, 100\}$. Kernel width parameter of Gaussian kernel in MMD term is selected from $\{10, 10^2, 10^3, 10^4\}$. We use SVM [31] with Gaussian kernel for no domain adaptation situation.

We empirically set $d = 1600$ on VisDA dataset. We use PCA over all samples from source and target domain to initialize P_T^0 and set $P_S^0 = P_T^0$. In all experiments on VisDA dataset, λ and μ are tuned over the list $\{40, 50, \ldots, 100\}$. Kernel width parameter of Gaussian kernel in MMD term is selected from $\{10^2, 10^3, 10^4, 10^5\}$. We use SVM [31] with linear kernel for no domain adaptation. For JGSA, we fix $\lambda = 1$ and $\mu = 0.8$, and γ is tuned over the list $\{0.2, 0.4, \ldots, 1\}$.

2) Parameters Sensitivity: We conduct empirical parameters sensitivity analysis.

• **The initial value P_S^0 and P_T^0.** In our experiments, we find that the final classification performance depends on the initial values of two transformations P_S and P_T. If they are closed in the sense of Frobenius norm, the final classification performance is relatively well. However, if they are far, the final classification is usually not good enough. We empirically set them as the same value.

• **Regularizers λ, μ.** When λ and μ are large, the MMD term is small and homologous constraint is satisfied. When $\lambda \to 0$, MMD term may be very large and overfitting occurs. When $\mu \to 0$, this amounts to discarding the homologous constraint and performance degrades. We empirically set regularizers $\lambda \in \{30, 50, \ldots, 110\}$ and $\mu \in \{50, 60, \ldots, 100\}$ in our experiment respectively.

• **Dimensionality d.** The feature dimension is 2048. Hence $d \in \{1, 2, \ldots, 2048\}$. We find that the proposed algorithm is effective for large dimensionality d when we use PCA for initialization. The cause is that PCA initialization means leveraging prior information of data and a small d will lead to much information loss of transferable knowledge from two domains after projection. When d is large, performance degrades.

3) Analysis on Resulted Optimal Transformations: We report the numerical record of chordal distance $\mathbf{d}(\mathcal{P}_S, \mathcal{P}_T)$ and Frobenius distance $\|P_S - P_T\|_F$ on Office-Home dataset and VisDA dataset in Table X. The numerical evidence signifies that the two resulted optimal transformations are really different and homologous in our trained models. The good performance on most of tasks are partially attributed to the effectiveness of homology.

4) The Effectiveness of Homology: In Table VIII and Table IX, we find that performance of the proposed method with homologous constraint (i.e. $\mu > 0$) is stronger than it without homologous constraint (i.e. $\mu = 0$). This may be attributed to the fact that homology helps to extract related and transferable components from source and target domain. We also notice that when $\mu = 0$, performance of HCA is weaker than SVM on some tasks, which means that there exits negative transfer phenomenon, i.e. performance degrades after domain adaptation. However, the performance is enhanced when we take homologous constraint into account. Hence homology is also helpful for alleviating negative transfer problem.

5) Comparison With SA [5]: From Table VIII and Table IX, we can see that the performance of our proposed approach outperforms SA on all tasks. This may be attributed to the homology and extended MMD.

VIII. CONCLUSION

In this paper, we propose homologous component analysis for domain adaptation, in which we resort to homologous principle and utilize side information (i.e. the projected data information with respect to source domain and target domain) in source domain to help distributions alignment. Besides, we give a generalized error bound analysis for domain adaptation for the semi-supervised domain adaptation setting. Two transformations can help to reduce this upper bound more than only one common transformation. Extensive experiments on synthetic and real data show the effectiveness of the proposed method and numerical evidence reveals that the two resulted optimal transformations outperform the conventional one transformation domain adaptation methods.

APPENDIX A
PROOF OF PROPOSITION 4

In view of constraint condition, both P_S and P_T are orthogonal matrix of size $D \times d$, i.e. $P_S^T P_S = I$, $P_T^T P_T = I$. Therefore we have

$$\begin{aligned}
&\|X_S - P_S P_S^T X_S\|_F^2 + \|X_T - P_T P_T^T X_T\|_F^2 \\
&\qquad + \lambda \mathrm{MMD}^2 + \mu \mathbf{d}(\mathcal{P}_S, \mathcal{P}_T)^2 \\
&\quad = \|X_S\|_F^2 - Tr(P_S^T X_S X_S^T P_S) \\
&\qquad + \|X_T^T\|_F^2 - Tr(P_T^T X_T X_T^T P_T) \\
&\qquad + \lambda (Tr(\widehat{X} K M K \widehat{X}^T) \\
&\qquad + \sum_{c=1}^{C} Tr(\widehat{X}^{(c)} K^{(c)} M^{(c)} K^{(c)} (\widehat{X}^{(c)})^T)) \\
&\qquad + \mu \left(d - Tr(P_S^T P_T P_T^T P_S) \right).
\end{aligned}$$

Because the constant terms $\|X_S\|_F^2$, $\|X_T\|_F^2$ and μd have no impact on optimization, we drop out them and actually maximize

$$Tr(P_S^T X_S X_S^T P_S) + Tr(P_T^T X_T X_T^T P_T) - \lambda(Tr(\widehat{X} K M K \widehat{X}^T) + \sum_{c=1}^{C} Tr(\widehat{X}^{(c)} K^{(c)} M^{(c)} K^{(c)} (\widehat{X}^{(c)})^T)) + \mu Tr(P_S^T P_T P_T^T P_S).$$

Appendix B
Proof of Theorem 1

We denote the event by $\mathcal{A}$

$$\begin{aligned}&\mathbf{E}_{(\wp_T(\mathbf{x}),\mathbf{y})\sim \Pr_T(\wp_T(\mathbf{x}),\mathbf{y})} L(h(\wp_T(\mathbf{x})),\mathbf{y})\\ &\quad\le \mathbf{E}_{(\wp_S(\mathbf{x}),\mathbf{y})\sim \Pr_S(\wp_S(\mathbf{x}),\mathbf{y})} L(h(\wp_S(\mathbf{x})),\mathbf{y})\\ &\qquad+\frac{1}{2} d_{H\triangle H}(\Pr_S(\wp_S(\mathbf{x}),\mathbf{y}), \Pr_T(\wp_T(\mathbf{x}),\mathbf{y}))\\ &\qquad+\gamma,\end{aligned}$$

the event by $\mathcal{B}$

$$\begin{aligned}&\mathbf{E}_{(\wp_S(\mathbf{x}),\mathbf{y})\sim \Pr_S(\wp_S(\mathbf{x}),\mathbf{y})} L(h(\wp_S(\mathbf{x})),\mathbf{y})\\ &\quad\le \frac{1}{m_S}\sum_{i=1}^{m_S} L(h(\wp_S(x_i)), y_i)\\ &\qquad+\sqrt{\frac{d(\log(2m_S/d)+1)-\log(\delta/4)}{m_S}},\end{aligned}$$

the event by $\mathcal{C}$

$$\begin{aligned}&\mathbf{E}_{(\wp_S(\mathbf{x}),\mathbf{y})\sim \Pr_S(\wp_S(\mathbf{x}),\mathbf{y})} L(h(\wp_S(\mathbf{x})),\mathbf{y})\\ &\le \frac{1}{m}(\sum_{i=1}^{m_S} L(h(\wp_S(x_i)), y_i)+\sum_{i=1}^{m_{Tl}} L(h(\wp_T(x_{m_S+i})), y_{m_S+i}))\\ &\quad+\sqrt{\frac{d(\log(2m_S/d)+1)-\log(\delta/4)}{m_S}},\end{aligned}$$

where $m = m_S + m_{Tl}$, and the event by $\widetilde{\mathcal{C}}$

$$\begin{aligned}&\mathbf{E}_{(\wp_S(\mathbf{x}),\mathbf{y})\sim \Pr_S(\wp_S(\mathbf{x}),\mathbf{y})} L(h(\wp_S(\mathbf{x})),\mathbf{y})\\ &\le \frac{1}{m_S}(\sum_{i=1}^{m_S} L(h(\wp_S(x_i)), y_i)+\sum_{i=1}^{m_{Tl}} L(h(\wp_T(x_{m_S+i})), y_{m_S+i}))\\ &\quad+\sqrt{\frac{d(\log(2m_S/d)+1)-\log(\delta/4)}{m_S}},\end{aligned}$$

respectively.

According to the intermediate result in the appendix in [29], $\Pr(\mathcal{A}) = 1$. We have $\Pr(\mathcal{B}) \ge 1-\delta$ because of [37]. Since $\widetilde{\mathcal{C}} \supseteq \mathcal{B}$, $\Pr(\widetilde{\mathcal{C}}) \ge \Pr(\mathcal{B}) \ge 1-\delta$. Let

$$\begin{aligned}\mathbf{z} &:= \frac{1}{m}(\sum_{i=1}^{m_S} L(h(\wp_S(x_i)), y_i)\\ &\quad+\sum_{i=1}^{m_{Tl}} L(h(\wp_T(x_{m_S+i})), y_{m_S+i})),\\ A &:= \frac{m_S}{m}(\mathbf{E}_{(\wp_S(\mathbf{x}),\mathbf{y})\sim \Pr_S(\wp_S(\mathbf{x}),\mathbf{y})} L(h(\wp_S(\mathbf{x})),\mathbf{y})\\ &\quad-\sqrt{\frac{d(\log(2m_S/d)+1)-\log(\delta/4)}{m_S}}),\end{aligned}$$

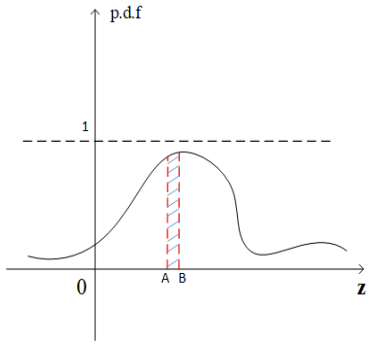


Fig. 6. The probability distribution function.

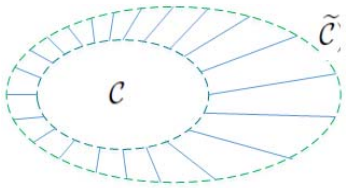


Fig. 7. The relationship between the event $\mathcal{C}$ and $\widetilde{\mathcal{C}}$.

$$\begin{aligned}B &:= \mathbf{E}_{(\wp_S(\mathbf{x}),\mathbf{y})\sim \Pr_S(\wp_S(\mathbf{x}),\mathbf{y})} L(h(\wp_S(\mathbf{x})),\mathbf{y})\\ &\quad-\sqrt{\frac{d(\log(2m_S/d)+1)-\log(\delta/4)}{m_S}}.\end{aligned}$$

We give the coarse illustration of $\mathbf{z}$ and A and B, which just helps to understand. Because $m > m_S$, $\mathcal{C} \subseteq \widetilde{\mathcal{C}}$. See Fig. 6, note that the value of L lies in $[0, 1]$, the area of the shaded area $\Delta(S; A, B)$ satisfies

$$\begin{aligned}\Delta(S; A, B) &= \Pr(A \le \mathbf{z} \le B) = \Pr(\widetilde{\mathcal{C}}\backslash\mathcal{C}) = \Pr(\widetilde{\mathcal{C}}) - \Pr(\mathcal{C})\\ &\le (1-\frac{m_S}{m})(\mathbf{E}_{(\wp_S(\mathbf{x}),\mathbf{y})\sim \Pr_S(\wp_S(\mathbf{x}),\mathbf{y})} L(h(\wp_S(\mathbf{x})),\mathbf{y})\\ &\quad-\sqrt{\frac{d(\log(2m_S/d)+1)-\log(\delta/4)}{m_S}})\\ &\le 1-\frac{m_S}{m} = \frac{m_{Tl}}{m},\end{aligned}$$

we have $\Pr(\mathcal{C}) + \frac{m_{Tl}}{m} \ge \Pr(\widetilde{\mathcal{C}})$. See Fig. 7 for intuition. The shaded area describes the probability gap.

Hence

$$\begin{aligned}\Pr(\mathcal{A}\cap\mathcal{C}) &= \Pr(\mathcal{A}) + \Pr(\mathcal{C}) - \Pr(\mathcal{A}\cup\mathcal{C})\\ &\ge \Pr(\mathcal{A}) + \left(\Pr(\widetilde{\mathcal{C}}) - \frac{m_{Tl}}{m}\right) - \Pr(\mathcal{A}\cup\mathcal{C})\\ &\ge 1 + \left(1-\delta-\frac{m_{Tl}}{m}\right) - 1\\ &\ge 1-\delta-\frac{m_{Tl}}{m}.\end{aligned}$$

Denote the event (7) by $\mathcal{D}$. Since $\mathcal{D} \supseteq \mathcal{A}\cap\mathcal{C}$, $\Pr(\mathcal{D}) \ge 1-\delta-\frac{m_{Tl}}{m}$, which is the desired result.

Appendix C
Optimization

The main difficulty lies on that it is hard to find a closed-form solution for optimization problem (6) by means of

Lagrange multiplier method. So we consider an iterative scheme to find a local minima. Based on [21], we optimize (6) by using alternating direction minimizing method (ADMM [56], [57]) in the context of Stiefel manifolds.

Let

$$\begin{aligned} f(P_S, P_T) = & Tr(P_S^T X_S X_S^T P_S) + Tr(P_T^T X_T X_T^T P_T) \\ & -\lambda(Tr(\widehat{X} L \widehat{X}^T) + \sum_{c=1}^{C} Tr(\widehat{X}^{(c)} L^{(c)} (\widehat{X}^{(c)})^T)) \\ & +\mu Tr(P_S^T P_T P_T^T P_S), \end{aligned}$$

where

$$\begin{aligned} L &= KMK \\ &= \begin{bmatrix} L_{SS} & L_{ST} \\ L_{TS} & L_{TT} \end{bmatrix} \in \mathbb{R}^{(m_S+m_T)\times(m_S+m_T)}, \end{aligned} \tag{8}$$

and

$$\begin{aligned} L^{(c)} &= K^{(c)} M^{(c)} K^{(c)} \\ &= \begin{bmatrix} L_{SS}^{(c)} & L_{ST}^{(c)} \\ L_{TS}^{(c)} & L_{TT}^{(c)} \end{bmatrix} \in \mathbb{R}^{(m_S^{(c)}+m_T^{(c)})\times(m_S^{(c)}+m_T^{(c)})}, \end{aligned} \tag{9}$$

The derivative of f with respect to P_S and P_T are

$$\begin{aligned} \frac{\partial f}{\partial P_S} = & (X_S X_S^T + \mu P_T P_T^T) P_S \\ & -\lambda(X_S(L_{SS} X_S^T P_S + 2 L_{ST} X_T^T P_T) \\ & + \sum_{c=1}^{C} X_S^{(c)} (L_{SS}^{(c)} (X_S^{(c)})^T P_S + 2 L_{ST}^{(c)} (X_T^{(c)})^T P_T), \end{aligned} \tag{10}$$

and

$$\begin{aligned} \frac{\partial f}{\partial P_T} = & (X_T X_T^T + \mu P_S P_S^T) P_T \\ & -\lambda(X_T (L_{TT} X_T^T P_T + 2 L_{TS} X_S^T P_S) \\ & + \sum_{c=1}^{C} X_T^{(c)} (L_{TT}^{(c)} (X_T^{(c)})^T P_T + 2 L_{TS}^{(c)} (X_S^{(c)})^T P_S)), \end{aligned} \tag{11}$$

where block matrices in (8) and (9) are used.

Note that we deal with maximization instead of minimization. Equivalently, we minimize $-f(P_S, P_T)$. Two main steps are included when we use ADMM in the context of Stiefel manifolds.

a) *Update* P_T: Minimize $-f(P_S, \cdot)$ on the Stiefel manifold $\mathcal{S}_D^d$. Let

$$A_T := \frac{\partial f}{\partial P_T} P_T^T - P_T \left(\frac{\partial f}{\partial P_T} \right)^T, \tag{12}$$

where $\frac{\partial f}{\partial P_T}$ is computed as (11). We construct a sequence $\{Y_T(\tau)\}_{\tau \geq 0}$, i.e.

$$Y_T(\tau^{new}) = P_T - \frac{\tau^{old}}{2} A_T (P_T + Y_T(\tau^{old})), \tag{13}$$

where

$$Y_T(\tau^{old}) = Q_T^{old} P_T, \tag{14}$$

$$Q_T^{old} := (I + \frac{\tau^{old}}{2} A_T)^{-1} (I + \frac{\tau^{old}}{2} A_T). \tag{15}$$

As shown in [21], this generated sequence $\{Y_t(\tau)\}_{\tau \geq 0}$ converges to a local minimizer of $-f(P_S, \cdot)$.

b) *Update* P_S: Minimize $-f(\cdot, P_T)$ on the Stiefel manifold $\mathcal{S}_D^d$. Let

$$A_S := \frac{\partial f}{\partial P_S} P_S^T - P_S \left(\frac{\partial f}{\partial P_S} \right)^T, \tag{16}$$

where $\frac{\partial f}{\partial P_S}$ is computed as (10). We construct a sequence $\{Y_S(\tau)\}_{\tau \geq 0}$, i.e.

$$Y_S(\tau^{new}) = P_S - \frac{\tau^{old}}{2} A_S (P_S + Y_S(\tau^{old})), \tag{17}$$

where

$$Y_S(\tau^{old}) = Q_S^{old} P_S, \tag{18}$$

$$Q_S^{old} := (I + \frac{\tau^{old}}{2} A_S)^{-1} (I + \frac{\tau^{old}}{2} A_S). \tag{19}$$

As shown in [21], this generated sequence $\{Y_S(\tau)\}_{\tau \geq 0}$ converges to a local minimizer of $-f(\cdot, P_T)$.

REFERENCES

[1] S. J. Pan and Q. Yang, "A survey on transfer learning," *IEEE Trans. Knowl. Data Eng.*, vol. 22, no. 10, pp. 1345–1359, Oct. 2010.

[2] L. Zhang, W. Zuo, and D. Zhang, "LSDT: Latent sparse domain transfer learning for visual adaptation," *IEEE Trans. Image Process.*, vol. 25, no. 3, pp. 1177–1191, Mar. 2016.

[3] B. Sun, J. Feng, and K. Saenko, "Return of frustratingly easy domain adaptation," in *Proc. AAAI*, Mar. 2015, pp. 2058–2065.

[4] J. Zhang, W. Li, and P. Ogunbona, "Joint geometrical and statistical alignment for visual domain adaptation," in *Proc. CVPR*, Jul. 2017, pp. 5150–5158.

[5] B. Fernando, A. Habrard, M. Sebban and T. Tuytelaars, "Unsupervised visual domain adaptation using subspace alignment," in *Proc. ICCV*, Dec. 2014, pp. 2960–2967.

[6] I. H. Jhuo, D. Liu, D. T. Lee, and S. F. Chang, "Robust visual domain adaptation with low-rank reconstruction," in *Proc. CVPR*, Jun. 2012, pp. 2168–2175.

[7] B. Li, Q. Yang, and X. Xue, "Transfer learning for collaborative filtering via a rating-matrix generative model," in *Proc. ICML*, Jun. 2009, pp. 617–624.

[8] L. Zhao, E. W. Xiang, E. Zhong, Z. Lu, and Q. Yang, "Active transfer learning for cross-system recommendation," in *Proc. AAAI*, Apr. 2013, pp. 1205–1211.

[9] L. Zhao, S. J. Pan, and Q. Yang, "A unified framework of active transfer learning for cross-system recommendation," *Artif. Intell.*, vol. 245, pp. 38–55, Apr. 2017.

[10] U. Sapkota, T. Solorio, M. Montes, and S. Bethard, "Transfer learning for collaborative filtering via a rating-matrix generative model," in *Proc. ACL*, Jun. 2016, pp. 2226–2235.

[11] T. Joey, S. J. Pan, I. W. Tsang, and S. S. Ho, "Transfer learning for cross-language text categorization through active correspondences construction," in *Proc. AAAI*, Feb. 2016, pp. 2400–2406.

[12] H. S. Bhatt, M. Sinha, and S. Roy, "Cross-domain text classification with multiple domains and disparate label sets," in *Proc. ACL*, Aug. 2016, pp. 1641–1650.

[13] A. Barreto, W. Dabney, R. Munos, J. J. Hunt, T. Schaul, and D. Silver, "Successor features for transfer in reinforcement learning," in *Proc. NIPS*, 2017, pp. 4056–4066.

[14] F. L. D. Silva and A. H. R. Costa, "Accelerating multiagent reinforcement learning through transfer learning," in *Proc. AAAI*, Feb. 2017, pp. 5034–5035.

[15] M. E. Taylor and P. Stone, "Transfer learning for reinforcement learning domains: A survey," *J. Mach. Learn. Res.*, vol. 10, pp. 1633–1685, Jul. 2009.

[16] S. Ben-David, T. Lu, T. Luu, and D. Pál, "Impossibility theorems for domain adaptation," in *Proc. ICAIS*, Sep. 2010, pp. 129–136.

[17] S. J. Pan, I. W. Tsang, J. T. Kwok, and Q. Yang, "Domain adaptation via transfer component analysis," *IEEE Trans. Neural Netw.*, vol. 22, no. 2, pp. 199–210, Feb. 2011.

[18] B. Gong, Y. Shi, F. Sha, and K. Grauman, "Geodesic flow kernel for unsupervised domain adaptation," in *Proc. CVPR*, Jun. 2012, pp. 2066–2073.
[19] J. Zhang, W. Li, and P. Ogunbona, "Joint geometrical and statistical alignment for visual domain adaptation," in *Proc. CVPR*, Jul. 2017, pp. 5150–5158.
[20] J. Tahmoresnezhad and S. Hashemi, "Visual domain adaptation via transfer feature learning," *Knowl. Inf. Syst.*, vol. 50, no. 2, pp. 585–605, Jun. 2017.
[21] Z. Wen and W. Yin, "A feasible method for optimization with orthogonality constraints," *Math. Program.*, vol. 142, no. 1, pp. 397–434, 2013.
[22] B. Wang *et al.*, "Locality preserving projections for Grassmann manifold," in *Proc. IJCAI*, Aug. 2017, pp. 2893–2900.
[23] A. Gretton, K. M. Borgwardt, M. J. Rasch, B. Schölkopf, and A. Smola, "A kernel two-sample test," *J. Mach. Learn. Res.*, vol. 13, no. 3, pp. 723–773, 2012.
[24] I. Steinwart, "On the influence of the kernel on the consistency of support vector machines," *J. Mach. Learn. Res.*, vol. 2, pp. 67–93, Mar. 2002.
[25] M. Baktashmotlagh, M. Harandi, and M. Salzmann, "On the influence of the kernel on the consistency of support vector machines," *J. Mach. Learn. Res.*, vol. 17, no. 1, pp. 3760–3789, Apr. 2016.
[26] A. Barg and D. Y. Nogin, "Bounds on packings of spheres in the Grassmann manifold," *IEEE Trans. Inf. Theory*, vol. 48, no. 9, pp. 2450–2454, Sep. 2002.
[27] J. G. Oxley, *Matroid Theory*, vol. 3. New York, NY, USA: Oxford University Press, 2006.
[28] M. Long, J. Wang, G. Ding, S. J. Pan, and P. S. Yu, "Adaptation regularization: A general framework for transfer learning," *IEEE Trans. Knowl. Data Eng.*, vol. 26, no. 5, pp. 1076–1089, May 2014.
[29] J. Blitzer, K. Crammer, A. Kulesza, F. Pereira, and J. Wortman, "Learning bounds for domain adaptation," in *Proc. NIPS*, 2007, pp. 129–136.
[30] K. Saenko, B. Kulis, M. Fritz, and T. Darrell, "Adapting visual category models to new domains," in *Proc. ECCV*, Sep. 2010, pp. 213–226.
[31] C. C. Chang and C. J. Lin, "LIBSVM: A library for support vector machines," *ACM Trans. Intell. Syst. Technol.*, vol. 2, no. 3, pp. 1–27, 2011.
[32] I. Kuzborskij and F. Orabona, "Stability and hypothesis transfer learning," in *Proc. ICML*, vol. 2013, pp. III-942–III-950.
[33] R. Gopalan, R. Li, and R. Chellappa, "Unsupervised adaptation across domain shifts by generating intermediate data representations," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 36, no. 11, pp. 2288–2302, Nov. 2014.
[34] W. Jiang, C. Deng, W. Liu, F. Nie, F. L. Chung, and H. Huang, "Theoretic analysis and extremely easy algorithms for domain adaptive feature learning," in *Proc. IJCAI*, 2017, pp. 1958–1964.
[35] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proc. IEEE*, vol. 86, no. 11, pp. 2278–2324, Nov. 1998.
[36] A. Frank and A. Asuncion. (2010). *UCI Machine Learning Repository*. [Online]. Available: http://archive.ics.uci.edu/ml
[37] V. Vapnik, "The nature of statistical learning theory," in *Proc. AI*, vol. 1995, pp. 988–999.
[38] V. M. Patel, R. Gopalan, R. Li, and R. Chellappa, "Visual domain adaptation: A survey of recent advances," *IEEE Signal Process. Mag.*, vol. 32, no. 3, pp. 53–69, May 2015.
[39] M. Xiao and Y. Guo, "Feature space independent semi-supervised domain adaptation via kernel matching," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 37, no. 1, pp. 54–66, Jan. 2014.
[40] N. Japkowicz and S. Stephen, "The class imbalance problem: A systematic study," *Intell. Data Anal.*, vol. 6, no. 5, pp. 429–449, Oct. 2002.
[41] H. Shimodaira, "Improving predictive inference under covariate shift by weighting the log-likelihood function," *J. Statist. Planning Inference*, vol. 90, no. 2, pp. 227–244, 2000.
[42] J. J. Heckman, "Sample selection bias as a specification error," *Econometrica*, vol. 47, no. 1, pp. 153–161, 1979.
[43] B. Zadrozny, "Learning and evaluating classifiers under sample selection bias," in *Proc. ICML*, Jul. 2004, pp. 114–121.
[44] B. Li, M. Ayazohlu, T. Mao, O. I. Camps, and M. Sznaier, "Activity recognition using dynamic subspace angles," in *Proc. CVPR*, Jun. 2011, pp. 3193–3200.
[45] B. Schólkopf, A. Smola, and K.-R. Müller, "Nonlinear component analysis as a kernel eigenvalue problem," *Neural Comput.*, vol. 10, no. 5, pp. 1299–1319, Jul. 1998.
[46] C.-A. Hou, Y.-H. H. Tsai, Y.-R. Yeh, and Y.-C. F. Wang, "Unsupervised domain adaptation with label and structural consistency," *IEEE Trans. Image Process.*, vol. 25, no. 12, pp. 5552–5562, Dec. 2016.
[47] Z. Ding and Y. Fu, "Robust transfer metric learning for image classification," *IEEE Trans. Image Process.*, vol. 26, no. 2, pp. 660–670, Feb. 2017.
[48] H. Lu, C. Shen, Z. Cao, Y. Xiao, and A. van den Hengel, "An embarrassingly simple approach to visual domain adaptation," *IEEE Trans. Image Process.*, vol. 27, no. 7, pp. 3403–3417, Jun. 2018.
[49] S. Li, S. Song, G. Huang, Z. Ding, and C. Wu, "Domain invariant and class discriminative feature learning for visual domain adaptation," *IEEE Trans. Image Process.*, vol. 27, no. 9, pp. 4260–4273, Sep. 2018.
[50] K. Bousmalis *et al.*, "Using simulation and domain adaptation to improve efficiency of deep robotic grasping," in *Proc. ICRA*, May 2018, pp. 4243–4250.
[51] F. Mahmood, R. Chen, and N. J. Durr, "Unsupervised reverse domain adaptation for synthetic medical images via adversarial training," *IEEE Trans. Med. Imaging*, vol. 37, no. 12, pp. 2572–2581, Dec. 2018.
[52] A. Rozantsev, M. Salzmann, and P. Fua, "Beyond sharing weights for deep domain adaptation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 41, no. 4, pp. 801–814, Apr. 2018.
[53] M. Wang and W. Deng, "Deep visual domain adaptation: A survey," *Neurocomputing*, vol. 312, pp. 135–153, Oct. 2018.
[54] G. Csurka, "A comprehensive survey on domain adaptation for visual applications," in *Domain Adaptation in Computer Vision Applications*. Cham, Switzerland: Springer, 2017, pp. 1–35.
[55] M. Long, Y. Cao, J. Wang, and M. I. Jordan, "Learning transferable features with deep adaptation networks," in *Proc. ICML*, Jul. 2015, pp. 97–105.
[56] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Found. Trends Mach. Learn.*, vol. 3, no. 1, pp. 1–122, Jan. 2011.
[57] S. Zhang and J. Xin, "Minimization of transformed L_1 penalty: Theory, difference of convex function algorithm, and robust application in compressed sensing," *Math. Program., Ser. B*, vol. 169, no. 1, pp. 307–336, 2018.
[58] H. Venkateswara, J. Eusebio, S. Chakraborty, and S. Panchanathan, "Deep hashing network for unsupervised domain adaptation," in *Proc. CVPR*, Jul. 2017, pp. 5018–5027.
[59] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. CVPR*, May 2016, pp. 770–778.
[60] Y. Ganin *et al.*, "Domain-adversarial training of neural networks," *J. Mach. Learn. Res.*, vol. 17, no. 1, pp. 2030–2096, May 2015.
[61] M. Long, J. Wang, and M. I. Jordan, "Deep transfer learning with joint adaptation networks," in *Proc. ICML*, pp. 2208–2217, 2017.
[62] M. Long, Z. Cao, J. Wang, and M. I. Jordan, "Conditional adversarial domain adaptation," in *Proc. NIPS*, 2018, pp. 1647–1657.
[63] S. Sankaranarayanan, Y. Balaji, C. D. Castillo, and R. Chellappa, "Generate to adapt: Aligning domains using generative adversarial networks," in *Proc. CVPR*, Jul. 2018, pp. 8503–8512.
[64] M. Long, H. Zhu, J. Wang, and M. I. Jordan, "Unsupervised domain adaptation with residual transfer networks," in *Proc. NIPS*, 2016, pp. 136–144.
[65] Y. Li, K. Swersky, and R. Zemel, "Generative moment matching networks," in *Proc. ICML*, 2015, pp. 1718–1727.

Youfa Liu received the B.S. degree from South Central University for Nationalities in 2014 and the M.S. degree from the Wuhan Institute of Physics and Mathematics, Chinese Academy of Sciences, in 2017. He is currently pursuing the Ph.D. degree with the School of Computer Science, Wuhan University. His current research interests include machine learning and computer vision.

Weiping Tu received the B.S. degree from Southwest Jiaotong University, Chengdu, in 1996, and the M.S and Ph.D. degrees in communication and information system from Wuhan University, Wuhan, China, in 2002 and 2011, respectively. She is currently an Associate Professor with the School of Computer Science, Wuhan University. Her research interest includes speech/image signal processing and communication.

Bo Du (M'10–SM'15) received the Ph.D. degree in photogrammetry and remote sensing from the State Key Laboratory of Information Engineering in Surveying, Mapping and Remote Sensing, Wuhan University, Wuhan, China, in 2010.

He is currently a Professor with the School of Computer Science, Wuhan University. He has over 60 research papers published in the IEEE TRANSACTIONS ON GEOSCIENCE AND REMOTE SENSING (TGRS), the IEEE TRANSACTIONS ON IMAGE PROCESSING (TIP), the IEEE JOURNAL OF SELECTED TOPICS IN EARTH OBSERVATIONS AND APPLIED REMOTE SENSING (JSTARS), and the IEEE GEOSCIENCE AND REMOTE SENSING LETTERS (GRSL). Thirteen of them are ESI hot papers or highly cited papers. His major research interests include pattern recognition, hyperspectral image processing, machine learning, and signal processing.

Dr. Du was a recipient of the Distinguished Paper Award from IJCAI 2018, the Best Paper Award of the IEEE Whispers 2018, the Champion Award of the IEEE Data Fusion Contest 2018, the Best Reviewer Award from the IEEE GRSS for his service to the IEEE JOURNAL OF SELECTED TOPICS IN EARTH OBSERVATIONS AND APPLIED REMOTE SENSING (JSTARS) in 2011, and ACM rising star awards for his academic progress in 2015. He was the Session Chair of the International Geoscience and Remote Sensing Symposium (IGARSS) 2018/2016 and the 4th IEEE GRSS Workshop on Hyperspectral Image and Signal Processing: Evolution in Remote Sensing (WHISPERS). He also serves as a Reviewer for 20 Science Citation Index (SCI) magazines, including IEEE TGRS, TIP, JSTARS, and GRSL.

Lefei Zhang (S'11–M'14) received the B.S. and Ph.D. degrees from Wuhan University, Wuhan, China, in 2008 and 2013, respectively.

He was a Big Data Institute Visitor with the Department of Statistical Science, University College London, in 2016, and a Hong Kong Scholar with the Department of Computing, The Hong Kong Polytechnic University, in 2017. He is currently a Professor with the School of Computer Science, Wuhan University. His research interests include pattern recognition, image processing, and remote sensing. He is also a Reviewer/PC Member of numerous journals/conferences, including the IEEE TPAMI, TIP, TGRS, AAAI, IJCAI, and ICDM.

Dacheng Tao (F'15) is currently a Professor of computer science and an ARC Laureate Fellow with the School of Information Technologies, Faculty of Engineering and Information Technologies, and the Inaugural Director of the UBTECH Sydney Artificial Intelligence Centre, The University of Sydney, Darlington, NSW, Australia. He mainly applies statistics and mathematics to artificial intelligence and data science. His research results have expounded in one monograph and over 200 publications at prestigious journals and prominent conferences, such as the IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE, the IEEE TRANSACTIONS ON IMAGE PROCESSING, the IEEE TRANSACTIONS ON NEURAL NETWORKS AND LEARNING SYSTEMS, the *International Journal of Computer Vision*, the *Journal of Machine Learning Research*, NIPS, ICML, CVPR, ICCV, ECCV, ICDM, and ACM SIGKDD.

Prof. Tao is a fellow of the Australian Academy of Science, AAAS, IAPR, OSA, and SPIE. He was a recipient of several best paper awards, such as the Best Theory/Algorithm Paper Runner Up Award from the IEEE ICDM 2007, the Best Student Paper Award from the IEEE ICDM 2013, the Distinguished Paper Award in the 2018 IJCAI, the 2014 ICDM 10-Year Highest-Impact Paper Award, the 2017 IEEE Signal Processing Society Best Paper Award, the 2015 Australian Scopus-Eureka Prize, and the 2018 IEEE ICDM Research Contributions Award.