2015 IEEE International Conference on Computer Vision

# Bi-shifting Auto-Encoder for Unsupervised Domain Adaptation

Meina Kan[1], Shiguang Shan[1,2], Xilin Chen[1]
[1]Key Laboratory of Intelligent Information Processing of Chinese Academy of Sciences (CAS),
Institute of Computing Technology, CAS, Beijing, 100190, China
[2]CAS Center for Excellence in Brain Science and Intelligence Technology
{kanmeina, sgshan, xlchen}@ict.ac.cn

## Abstract

*In many real-world applications, the domain of model learning (referred as source domain) is usually inconsistent with or even different from the domain of testing (referred as target domain), which makes the learnt model degenerate in target domain, i.e., the test domain. To alleviate the discrepancy between source and target domains, we propose a domain adaptation method, named as Bi-shifting Auto-Encoder network (BAE). The proposed BAE attempts to shift source domain samples to target domain, and also shift the target domain samples to source domain. The non-linear transformation of BAE ensures the feasibility of shifting between domains, and the distribution consistency between the shifted domain and the desirable domain is constrained by sparse reconstruction between them. As a result, the shifted source domain is supervised and follows similar distribution as target domain. Therefore, any supervised method can be applied on the shifted source domain to train a classifier for classification in target domain. The proposed method is evaluated on three domain adaptation scenarios of face recognition, i.e., domain adaptation across view angle, ethnicity, and imaging sensor, and the promising results demonstrate that our proposed BAE can shift samples between domains and thus effectively deal with the domain discrepancy.*

## 1. Introduction

For classification problems in computer vision, the most typical technique is to learn a classifier on the training samples with class label and then apply it to classify the testing samples. The basic assumption behind this is that the training samples and testing samples share the same or similar distributions. However, in real world applications, many factors (*e.g.*, pose of faces, illumination, imaging quality, etc) cause the mismatch of distribution between the training samples and testing samples, which usually degenerate the performance of learnt classifier on the testing samples.

The techniques for addressing this challenging domain disparity problems are often referred as domain adaptation [19][14][13][28][12] or generally as transfer learning [25]. Usually, the domain with labeled data, which is also where the classifier is learnt, is called as *source domain*, and the domain where the target task is conducted but with different distribution is called as *target domain*. Rather than the general transfer learning, this work only focuses on the domain adaption, in which the source domain and target domain share the same task but follow different distributions. Depending on whether class labels are available for target domain, domain adaptation can be categorized into two settings [13][30][14], supervised domain adaptation and unsupervised domain adaptation.

In scenario of supervised domain adaptation, labeled data is available in the target domain but the number is usually too small to train a good classifier, while in unsupervised domain adaptation only unlabeled data (but generally in large-scale) is available in the target domain, which is more challenging. This work mainly concentrates on the unsupervised domain adaptation problem, of which the essence is how to employ the unlabeled data of target domain to guide the model learning from the labeled source domain.

An intuitive strategy is re-weighting or re-sampling the samples of source domain to make the re-weighted/re-sampled source domain shares similar distribution as target domain, *e.g.*, sample selection bias [38][18], and particularly covariant shift [31][33][32][15][4], etc. These methods usually need to measure the distance between two distributions, which is also very hard for complex scenarios.

More popular methods attempt to design domain-invariant feature representation [2][5][24][23][30][26][14] [13][10][12]. In [5], a structural correspondence learning method automatically induces correspondences among features from different domains with pivot feature, that behave in the same way for discriminative learning in both domains. In Sampling Geodesic Flow (SGF) [14], a serial of intermediate subspaces between the source and target domains along the Grassmann manifold are sampled to de-

1550-5499/15 $31.00 © 2015 IEEE
DOI 10.1109/ICCV.2015.438

scribe the underlying domain shift, and the projections of labeled source domain data onto these subspaces can be used to learn a classifier applicable for the target domain. Furthermore, in [13], a Geodesic Flow Kernel (GFK) approach is proposed to characterize the domain shift by integrating an infinite number of subspaces. In [12], a subset of labeled instances in source domain that are similar to the target domain is identified as landmarks to bridge the source and target domains by constructing an easier auxiliary domain adaptation task. Based on domain-invariant feature, the learnt model or metric from labeled source domain can be applicable for the classification in target domain.

On the other hand, some methods endeavor to directly optimize a classifier or metric which follows small or no discrepancy with the target domain [8][6][30][27]. In [7], a progressive transductive support vector machine is developed to iteratively label and modify the unlabeled target domain to achieve a wider margin for target domain. In [6], the discriminant classifier is adjusted step by step to the target domain by iteratively deleting the samples from source domain and adding samples from target domain until the final classification function is determined only based on samples from target domain. Most of these methods exploit an iteration scheme to gradually adapt the supervised information of the source domain to the target domain. In [30], the domain-invariant feature and classifier are jointly learnt by optimizing an information-theoretic metric as an proxy to the expected misclassification error on the target domain.

For the existing methods, maximum mean discrepancy, K-L and Bregman divergence are the most commonly used criteria to measure the discrepancy between source and target domains. Recently, low-rank representation constraint is proposed to guide the reduction of discrepancy between domains. In [20], the samples of source domain are mapped into an intermediate representation such that the transformed source domain samples can be linearly reconstructed by target domain samples in lowest rank structure, and then the transformed source samples can be used to learn a classifier for classification in target domain. In [28][29], a common and discriminant subspace is achieved via a low-rank representation constraint, which attempts to ensure that each datum in source domain can be linearly represented by target domain samples. In [21], the samples in source domain are transformed to target domain where each source domain sample can be linearly reconstructed by sparse number of target domain samples. In [11], a high level feature representation is learnt as domain-invariant feature by employing denoising auto-encoder to recover the input instances from both source and target domains, which is expected to characterize the commonality of both domains.

In works [20][28][29], linear transformation is employed to project source domain, target domain or both to a subspace where the samples of one domain can be linearly reconstructed by another domain. The linear reconstruction principle enforces the domain consistency and achieves promising performance. However in many cases the source domain might be quite different from the target domain, and linear transformation can hardly remove the discrepancy completely for further linear reconstruction. The work [11] employs non-linear transformation to model the variations but with no domain consistency constraint, and thus cannot well characterize the commonality of domains. In order to effectively handle the domain discrepancy, we propose a Bi-shifting Auto-Encoder network (BAE) which can shift source domain samples to target domain and also shift the target domain samples to source domain, as shown in Fig. 1. In BAE, the non-linear transformation ensures the feasibility of shifting between domains, and the sparse reconstruction ensures the distribution consistency between the shifted domain and desirable domain. Specifically, our bi-shifting auto-encoder network has one common encoder $\mathbf{f}_c$, two decoders $\mathbf{f}_s$ and $\mathbf{f}_t$ which can map an image to the source and target domain respectively. As a result, the source domain can be shifted to target domain along with its class label, and any supervised method can be applied on shifted source domain to train a classifier for classification in target domain, as the shifted source domain follows similar distribution as target domain.

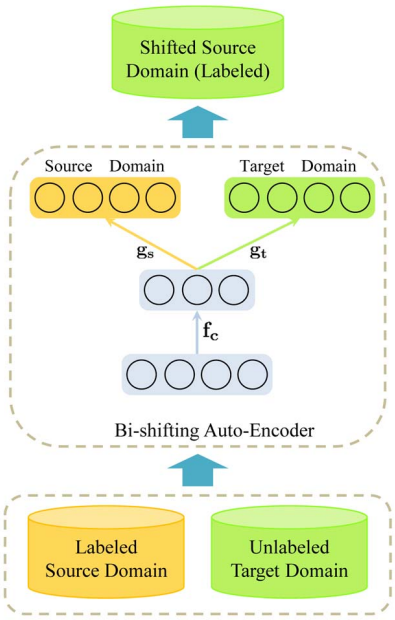


Figure 1. An overview of Bi-shifting Auto-Encoder (BAE). Through BAE, a source domain sample can be transformed to target domain where it can be sparsely and linearly reconstructed by target domain samples, and a target domain sample can be transformed to source domain where it can be sparsely and linearly reconstructed by source domain samples too.

The reminder of this paper is organized as follows. Sec.

2 presents the proposed bi-shifting auto-encoder network and its optimization. Sec. 3 evaluates the proposed method on three domain adaptation face recognition scenarios, *i.e.*, domain adaptation across view angle, ethnicity and imaging sensor. Finally, a conclusion is given in the last section.

## 2. Bi-shifting Auto-Encoder

### 2.1. Notations and Problem

For clear description in the following, we first define some notations. In the whole text, upper-case and lower-case characters represent the matrices and vectors respectively. Unless otherwise specified, the symbols $s$ and $t$ used in the superscript or subscript denotes the source domain and target domain respectively.

In source domain, there are $n_s$ labeled samples in $d$-dimension, denoted as $\mathbf{X}_s = [\mathbf{x}_1^s, \mathbf{x}_2^s, \cdots, \mathbf{x}_{n_s}^s] \in \mathbb{R}^{d\times n_s}$, with their class labels $\mathbf{y}_s = [y_1^s, y_2^s, \cdots, y_{n_s}^s], y_i^s \in \{1, 2, \cdots, c_s\}$, where $\mathbf{x}_i^s \in \mathbb{R}^{d\times 1}$ is the feature representation of the $i$-th source domain sample, $y_i^s$ is its class label, and $c_s$ is the number of classes in the source domain.

In target domain, there are $n_t$ samples in $d$-dimension without class label, denoted as $\mathbf{X}_t = [\mathbf{x}_1^t, \mathbf{x}_2^t, \cdots, \mathbf{x}_{n_t}^t] \in \mathbb{R}^{d\times n_t}$, where $\mathbf{x}_i^t \in \mathbb{R}^{d\times 1}$ is the feature representation of the $i$-th target domain sample.

The problem to deal with is how to learn a model for classification in target domain with only unlabeled samples of target domain $\mathbf{X}_t$ and labeled but distributed differently samples of source domain $(\mathbf{X}_s, \mathbf{y}_s)$.

### 2.2. Auto-Encoder (AE)

For an auto-encoder neural network [3][35] with single hidden layer, it is usually compromised of two parts, encoder and decoder. The encoder, denoted as $\mathbf{f}$, attempts to map the input $\mathbf{x}\in\mathbb{R}^{d\times 1}$ into hidden representations, denoted as $\mathbf{z}\in\mathbb{R}^{r\times 1}$, in which $r$ is the number of neurons in hidden layer. Typically, $\mathbf{f}$ is a nonlinear transform as follows:

$$\mathbf{z} = \mathbf{f}(\mathbf{x}) = s(\mathbf{W}\mathbf{x} + \mathbf{b}), \tag{1}$$

where $\mathbf{W} \in \mathbb{R}^{r\times d}$ is a linear transform , $\mathbf{b} \in \mathbb{R}^{r\times 1}$ is the basis and $s(\cdot)$ is the so-called element-wise "activation function", which is usually non-linear, such as sigmoid function $s(x) = \frac{1}{1+e^{-x}}$ or tanh function $s(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}}$.

The decoder, denoted as $\mathbf{g}$, tries to map the hidden representation $\mathbf{z}$ back to the input $\mathbf{x}$, *i.e.*,

$$\mathbf{x} = \mathbf{g}(\mathbf{z}) = s(\mathbf{W}'\mathbf{z} + \mathbf{b}'), \tag{2}$$

with$\mathbf{W}' \in \mathbb{R}^{d\times r}$ and basis $\mathbf{b}' \in \mathbb{R}^{d\times 1}$.

To optimize the parameters $\mathbf{W}, \mathbf{b}$, $\mathbf{W}'$ and $\mathbf{b}'$, usually the least square error is employed as the cost function:

$$\min_{\mathbf{W},\mathbf{b},\mathbf{W}',\mathbf{b}'} \sum_{i=1}^{N} \|\mathbf{x}_i - \mathbf{g}(\mathbf{f}(\mathbf{x}_i))\|_2^2, \tag{3}$$

where $\mathbf{x}_i$ represents the $i^{th}$ one of $N$ training sample. Due to the non-linearity of encoder and decoder in Eq. (3), it is difficult to solve, and thus the gradient descent algorithm is commonly employed.

### 2.3. Bi-shifting Auto-Encoder (BAE)

The typical auto-encoder in Eq. (3) tries to reconstruct the input itself, which is usually employed for dimension reduction, or feature learning. Nevertheless, our proposed bi-shifting auto-encoder network attempts to shift samples between domains to deal with the domain discrepancy. As shown in Fig. 1, our bi-shifting auto-encoder network consists of one encoder $\mathbf{f}_c$, and two decoders, *i.e.*, $\mathbf{g}_s$ and $\mathbf{g}_t$, which can transform an input sample to source domain and target domain respectively.

Specifically, the encoder $\mathbf{f}_c$ aims to map an input sample $\mathbf{x}$ into hidden feature representation $\mathbf{z}$ which is common to both source and target domains as below:

$$\mathbf{z} \triangleq \mathbf{f}_c(\mathbf{x}) = \sigma(\mathbf{W}_c\mathbf{x} + \mathbf{b}_c) \tag{4}$$

The decoder $\mathbf{g}_s$ intends to map the hidden representation to source domain, and decoder $\mathbf{g}_t$ intends to map the hidden representation to target domain as follows:

$$\begin{aligned} \mathbf{g}_s(\mathbf{z}) &= \sigma(\mathbf{W}_s\mathbf{z} + \mathbf{b}_s), \\ \mathbf{g}_t(\mathbf{z}) &= \sigma(\mathbf{W}_t\mathbf{z} + \mathbf{b}_t), \end{aligned} \tag{5}$$

where $s(\cdot)$ is the element-wise nonlinear activation function, *e.g.*, sigmoid or tanh function, $\mathbf{W}_c$ and $\mathbf{b}_c$ are the parameters for encoder $\mathbf{f}_c$, $\mathbf{W}_s$ and $\mathbf{b}_s$ are the parameters for decoder $\mathbf{g}_s$, $\mathbf{W}_t$ and $\mathbf{b}_t$ are the parameters for decoder $\mathbf{g}_t$.

For source domain $\mathbf{X}_s$, on one hand, with encoder $\mathbf{f}_c$ and decoder $\mathbf{g}_s$ they should be mapped to source domain, *i.e.*, $\mathbf{X}_s$ itself. On the other hand, with encoder $\mathbf{f}_c$ and decoder $\mathbf{g}_t$, they should be mapped to target domain. Although it is unknown what the mapped samples look like, they are expected to follow the same distribution as target domain. This kind of distribution consistency between two domains can be characterized by the local structure consistency.

The two domains $\mathbf{X}_s$ and $\mathbf{X}_t$ can be generally considered to lie on two manifolds $M_s$ and $M_t$, and the distance between the two manifolds can be used to describe the domain discrepancy of them. As indicated in [36], the distance between manifolds can be measured by the distance between instances of both manifolds. Given the instances $\mathbf{x}_i^s|_{i=1}^{n_s}$ from $M_s$, assume that we can traverse all samplings of $n_s$ instances from $M_t$, then we can get a sampling $\mathbf{x}_i^{t*}|_{i=1}^{n_s}$ that minimize the distance of $\sum_{i=1}^{n_s} ||\mathbf{x}_i^s - \mathbf{x}_i^{t*}||^2$, which can be used to depict the distance of the two manifolds. However, actually it is impossible to look through all samplings to get the optimal $\mathbf{x}_i^{t*}|_{i=1}^{n_s}$. But we have another sampling $\mathbf{X}_t$ sampled from the same manifold as $\mathbf{x}_i^{t*}|_{i=1}^{n_s}$. This means that each $\mathbf{x}_i^{t*}$ can be reconstructed by using its several

neighbors from $\mathbf{X}_t$, *i.e.*, $\mathbf{x}_i^{t*} = \sum_{k=1}^{r} b_k \mathbf{x}_{ik}^t$ with $\mathbf{x}_{ik}^t$ as one of the $r$ neighbors. $\mathbf{x}_i^{t*} = \sum_{k=1}^{r} b_k \mathbf{x}_{ik}^t$ can be further formulated as $\mathbf{x}_i^{t*} = \mathbf{X}_t \beta_i^t$, and $\beta_i^t$ is a sparse vector with the non-zero values corresponding to the local neighbors. Consequently, $\sum_{i=1}^{n_s} ||\mathbf{x}_i^s - \mathbf{X}_t \beta_i^t||^2$ can be used to describe the distance between the two domains, where $\beta_i^t$ should be locally sparse. For simplicity, we relax $\beta_i^t$ to be sparse without explicit restriction of the locality, since the non-zeros values from sparsity usually tends to be local.

As shown in Fig. 2, to enforce the shifted source domain and target domain share similar distribution, each mapped source domain sample should be sparely reconstructed by several neighbors from target domain, formulated as below:

$$\mathbf{g}_t(\mathbf{f}_c(\mathbf{x}_i^s)) = \mathbf{X}_t \boldsymbol{\beta}_i^t, \quad s.t., |\boldsymbol{\beta}_i^t|_0 < \tau, \tag{6}$$

where $\boldsymbol{\beta}_i^t$ is the sparse coefficients for the reconstruction of shifted source domain sample. Eq. (6) enforces that each local structure of shifted source domain is consistent with that of target domain, which ensures that the shifted source domain follow similar distribution as target domain. The overall objective for the samples of source domain $\mathbf{X}_s$ can be formulated as below, with $\mathbf{B}_t = \left[\boldsymbol{\beta}_1^t, \cdots, \boldsymbol{\beta}_{n_s}^t\right]$:

$$\begin{aligned} \min_{\mathbf{f}_c, \mathbf{g}_s, \mathbf{g}_t, \boldsymbol{\beta}_i^t} & ||\mathbf{X}_s - \mathbf{g}_s(\mathbf{f}_c(\mathbf{X}_s))||_2^2 + ||\mathbf{X}_t \mathbf{B}_t - \mathbf{g}_t(\mathbf{f}_c(\mathbf{X}_s))||_2^2 \\ & s.t., |\boldsymbol{\beta}_i^t|_0 < \tau, \end{aligned} \tag{7}$$

Here, the $\mathbf{g}_s(\mathbf{f}_c(\mathbf{X}_s))$ and $\mathbf{g}_t(\mathbf{f}_c(\mathbf{X}_s))$ represent the matrices of $[\mathbf{g}_s(\mathbf{f}_c(\mathbf{x}_1^s)), \mathbf{g}_s(\mathbf{f}_c(\mathbf{x}_2^s)), \cdots, \mathbf{g}_s(\mathbf{f}_c(\mathbf{x}_{n_s}^s))]$ and $[\mathbf{g}_t(\mathbf{f}_c(\mathbf{x}_1^s)), \mathbf{g}_t(\mathbf{f}_c(\mathbf{x}_2^s)), \cdots, \mathbf{g}_t(\mathbf{f}_c(\mathbf{x}_{n_s}^s))]$ respectively for concise representation. The same simplifications are used hereinafter if without misunderstanding.

Similarly, for the samples of target domain $\mathbf{X}_t$, on one hand, with encoder $\mathbf{f}_c$ and decoder $\mathbf{g}_t$ they should be mapped to the target domain, *i.e.*, $\mathbf{X}_t$ itself. On the other hand, with encoder $\mathbf{f}_c$ and decoder $\mathbf{g}_s$ they should be mapped to the source domain, where they are constrained to be sparsely reconstructed by several neighbors from source domain, so as to ensure a similar distribution between the source domain and shifted target domain. The overall objective for the samples of target domain $\mathbf{X}_t$ can be formulated as below with $\mathbf{B}_s = \left[\boldsymbol{\beta}_1^s, \cdots, \boldsymbol{\beta}_{n_t}^s\right]$:

$$\begin{aligned} \min_{\mathbf{f}_c, \mathbf{g}_s, \mathbf{g}_t, \boldsymbol{\beta}_i^s} & ||\mathbf{X}_s \mathbf{B}_s - \mathbf{g}_s(\mathbf{f}_c(\mathbf{X}_t))||_2^2 + ||\mathbf{X}_t - \mathbf{g}_t(\mathbf{f}_c(\mathbf{X}_t))||_2^2 \\ & s.t., |\boldsymbol{\beta}_i^s|_0 < \tau, \end{aligned} \tag{8}$$

The $\mathbf{L}0$-norm problem in Eq. (7) and Eq. (8) are non-convex and hard to solve, so they are relaxed to $\mathbf{L}1$-norm as most existing methods do. Therefore, the objective of the bi-shifting auto-encoder can be formulated as following:

$$\begin{aligned} \min_{\mathbf{f}_c \mathbf{g}_s, \mathbf{g}_t, \mathbf{B}_s, \mathbf{B}_t} & ||\mathbf{X}_s - \mathbf{g}_s(\mathbf{f}_c(\mathbf{X}_s))||_2^2 + ||\mathbf{X}_t \mathbf{B}_t - \mathbf{g}_t(\mathbf{f}_c(\mathbf{X}_s))||_2^2 \\ & + ||\mathbf{X}_s \mathbf{B}_s - \mathbf{g}_s(\mathbf{f}_c(\mathbf{X}_t))||_2^2 + ||\mathbf{X}_t - \mathbf{g}_t(\mathbf{f}_c(\mathbf{X}_t))||_2^2 \\ & + \gamma \left( \sum_{i=1}^{n_s} |\boldsymbol{\beta}_i^t|_1 + \sum_{i=1}^{n_t} |\boldsymbol{\beta}_i^s|_1 \right). \end{aligned} \tag{9}$$

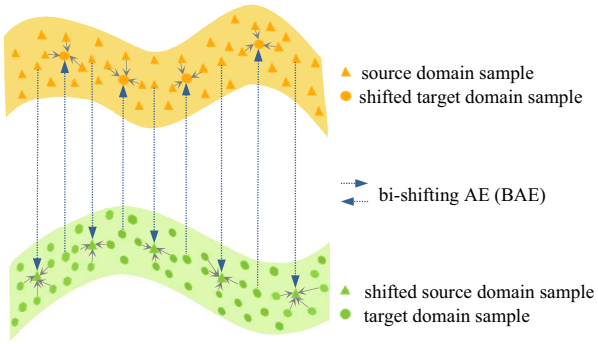


Figure 2. Illustration of sparse reconstruction constraint for distribution consistency. If each shifted source domain sample (green triangle) can be sparsely reconstructed by several local neighbors from target domain (green circle), they tends to follow similar local structure, meaning similar distribution in whole. Similarly, each shifted target domain sample (yellow circle) is constrained to be sparsely reconstructed by source domain neighbors (yellow triangle), to enforce them follow similar distributions.

Here, $\gamma$ is a parameter to control the sparsity, *i.e.*, a larger $\gamma$ leads to less samples selected for the sparse reconstruction, and smaller $\gamma$ leads to more samples selected for the sparse reconstruction. Empirically, the first four terms need to be normalized to similar scale to avoid the dominance of some terms. In Eq. (9), the nonlinear mapping function of the encoder and decoders ensures the feasibility of shifting samples from one domain to another, while the sparse reconstruction constraint promises the shifted domain follows similar distribution as the desirable domain.

With Eq. (9), a bi-shifting auto-encoder can be achieved to map any input sample to source and target domains respectively. Especially, the labeled source domain samples, $(\mathbf{X}_s, \mathbf{y}_s)$, can be shifted to target domain along with its class label as $(\mathbf{G}_t, \mathbf{y}_s)$, $\mathbf{G}_t \triangleq \mathbf{g}_t(\mathbf{f}_c(\mathbf{X}_s))$. The mapped source domain samples $\mathbf{G}_t$ share similar distribution as target domain, so any supervised method can be applied to learn a classifier for classification in target domain. In this work, Fisher Linear discriminant analysis (LDA) [1] is employed for supervised dimension reduction and the nearest neighbor classifier for recognition.

## 2.4. Optimization

Eq. (9) is hard to solve due to the complex non-linearity of the encoder and decoder, so the alternating optimization approach is employed to iteratively solve the network $\mathbf{f}_c, \mathbf{g}_s, \mathbf{g}_t$ and sparse reconstruction coefficients $\mathbf{B}_s, \mathbf{B}_t$.

**STEP 1:** given $\mathbf{f}_c, \mathbf{g}_s$ and $\mathbf{g}_t$, optimize $\mathbf{B}_s$ and $\mathbf{B}_t$.

When $\mathbf{f}_c, \mathbf{g}_s$ and $\mathbf{g}_t$ are fixed, the objective in Eq. (9) can

be reformulated as below:

$$\min_{\mathbf{B}_s,\mathbf{B}_t} ||\mathbf{X}_t\mathbf{B}_t - \mathbf{G}_t||_2^2 + ||\mathbf{X}_s\mathbf{B}_s - \mathbf{G}_s||_2^2 + \gamma\left(\sum_{i=1}^{n_s}|\boldsymbol{\beta}_i^t|_1 + \sum_{i=1}^{n_t}|\boldsymbol{\beta}_i^s|_1\right) \tag{10}$$

with $\mathbf{g}_t(\mathbf{f}_c(\mathbf{X}_s)) \triangleq \mathbf{G}_t = \left[\mathbf{g}_1^t, \cdots, \mathbf{g}_{n_s}^t\right]$ and $\mathbf{g}_s(\mathbf{f}_c(\mathbf{X}_t)) \triangleq \mathbf{G}_s = \left[\mathbf{g}_1^s, \cdots, \mathbf{g}_{n_t}^s\right]$. In Eq. (10), $\mathbf{B}_s$ and $\mathbf{B}_t$ are independent of each other, so they can be optimized independently. Namely, $\mathbf{B}_s = \left[\boldsymbol{\beta}_1^s, \cdots, \boldsymbol{\beta}_{n_t}^s\right]$ can be optimized as:

$$\begin{aligned}&\min_{\boldsymbol{\beta}_i^s} ||\mathbf{X}_s\mathbf{B}_s - \mathbf{G}_s||_2^2 + \gamma\sum_{i=1}^{n_t}|\boldsymbol{\beta}_i^s|_1 \\ &\Leftrightarrow \min_{\boldsymbol{\beta}_i^s}\sum_{i=1}^{n_t}||\mathbf{X}_s\boldsymbol{\beta}_i^s - \mathbf{g}_i^s||_2^2 + \gamma\sum_{i=1}^{n_t}|\boldsymbol{\beta}_i^s|_1\end{aligned} \tag{11}$$

As seen, $\boldsymbol{\beta}_1^s, \beta_2^s, \cdots, \boldsymbol{\beta}_{n_t}^s$ in Eq. (11) are also independent of each other, which means each $\boldsymbol{\beta}_i^s$ can be further separately solved as a lasso problem:

$$\min_{\boldsymbol{\beta}_i^s} ||\mathbf{X}_s\boldsymbol{\beta}_i^s - \mathbf{g}_i^s||_2^2 + \gamma|\boldsymbol{\beta}_i^s|_1 \tag{12}$$

Eq. (12) can be easily solved by using forward stepwise regression like algorithm, *i.e.*, the least angle regression [9].

Similarly, each $\boldsymbol{\beta}_i^t$ in $\mathbf{B}_t$ is also independent of each other and can be separately optimized as below:

$$\min_{\boldsymbol{\beta}_i^t} ||\mathbf{X}_t\boldsymbol{\beta}_i^t - \mathbf{g}_i^t||_2^2 + \gamma|\boldsymbol{\beta}_i^t|_1. \tag{13}$$

The problem in Eq. (13) can be also easily solved by using the least angle regression algorithm [9].

**STEP 2:** given $\mathbf{B}_s$ and $\mathbf{B}_t$, optimize $\mathbf{f}_c, \mathbf{g}_s$ and $\mathbf{g}_t$.

When $\mathbf{B}_s$ and $\mathbf{B}_t$ are fixed, the objective in Eq. (9) can be reformulated as below:

$$\begin{aligned}\min_{\mathbf{f}_c,\mathbf{g}_s,\mathbf{g}_t} &||\mathbf{X}_s - \mathbf{g}_s(\mathbf{f}_c(\mathbf{X}_s))||_2^2 + ||\widetilde{\mathbf{X}}_t - \mathbf{g}_t(\mathbf{f}_c(\mathbf{X}_s))||_2^2 \\ &+ ||\widetilde{\mathbf{X}}_s - \mathbf{g}_s(\mathbf{f}_c(\mathbf{X}_t))||_2^2 + ||\mathbf{X}_t - \mathbf{g}_t(\mathbf{f}_c(\mathbf{X}_t))||_2^2\end{aligned} \tag{14}$$

with $\mathbf{X}_t\mathbf{B}_t \triangleq \widetilde{\mathbf{X}}_t$ and $\mathbf{X}_s\mathbf{B}_s \triangleq \widetilde{\mathbf{X}}_s$. Eq. (14) can be easily optimized by gradient descent as the typical auto-encoder.

**STEP 3:** Repeat step 1 and 2 until $\mathbf{f}_c, \mathbf{g}_s, \mathbf{g}_t, \mathbf{B}_s$ and $\mathbf{B}_t$ converge or a maximum number of iterations is exceeded.

Before the alternation, the network $\mathbf{f}_c$, $\mathbf{g}_s$ and $\mathbf{g}_t$ are initialized by optimizing the following objective:

$$\min_{\mathbf{f}_c,\mathbf{g}_s,\mathbf{g}_t,\mathbf{B}_s,\mathbf{B}_t} ||\mathbf{X}_s - \mathbf{g}_s(\mathbf{f}_c(\mathbf{X}_s))||_2^2 + ||\mathbf{X}_t - \mathbf{g}_t(\mathbf{f}_c(\mathbf{X}_t))||_2^2 \tag{15}$$

## 3. Experiments

In this section, we evaluate the proposed method by comparing with existing methods on three face recognition scenarios: 1) *domain adaptation across view angle*, where the source and target domains are from different view angles; 2) *domain adaptation across ethnicity*, where the source and target domains are from different ethnicities, Mongolian and Caucasian; 3) *domain adaptation across imaging sensor*, where the images in source and target domains are captured under two different sensors, visual light (VIS) and near-infrared light (NIR) sensors. Several competitive approaches are briefly described as below. For all methods, their parameters are tuned to report the best results, and Linear Discriminant Analysis (LDA) is employed for supervised learning or initialization for fair comparison.

**PCA** [34]. Principal Component Analysis is a typical unsupervised method, taken as the baseline by being directly conducted on target domain.

**Source LDA** [1]. Fisher's Linear Discriminant analysis is a widely-used supervised approach for feature extraction. The LDA trained on the source domain with no adaptation is also tested as a baseline, denoted as "Source LDA".

**ITL** [30]. Information Theoretical Learning aims at identifying a discriminative subspace where source and target domains are similarly distributed, by optimizing an information theoretic metric as an proxy to the expected misclassification error on target domain. ITL is initialized with random matrix, PCA and LDA respectively, and dimension of the metric is tuned so as to report the best performance.

**SGF** [14]. In Sampling Geodesic Flow approach, a series of intermediate common representations are created by projecting the data onto the sampled intermediate subspaces. With the projected intermediate representation, LDA and 1-NN classifier is employed for classification. The number of sampled subspaces, dimension of subspace, and the dimension of LDA are tuned to report the best results.

**GFK** [14]. Geodesic Flow Kernel models domain shift by integrating an infinite number of subspaces that characterize changes in geometric and statistical properties from the source to target domain. For GFK, the supervised LDA subspace is used as the source subspace, PCA subspace is used as the target subspace, and 1-NN classifier is employed for final classification. The dimension of source subspace and target subspace are tuned to report the best results.

**Landmarks** [12]. This approach automatically discov-

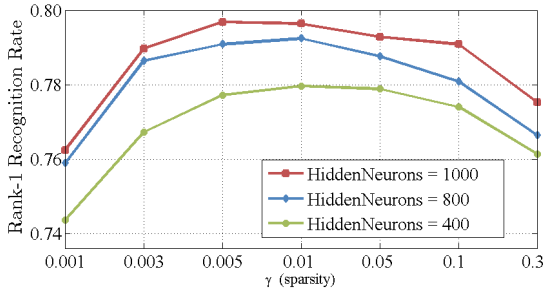


Figure 3. Performance of the proposed BAE w.r.t. different number of hidden neurons and different sparsity $\gamma$, with Caucasian as source domain and Mongolian as target domain respectively.

Table 1. Evaluation of domain adaptation across view angle on the MultiPIE dataset.

| | -45° →0° | -30° →15° | -15° →30° | 0° →45° | -45° →-15° | -15° →15° | 15° →45° | Average |
|---|---|---|---|---|---|---|---|---|
| Source LDA [1] | 0.665 | 0.693 | 0.669 | 0.617 | 0.703 | 0.719 | 0.637 | 0.672 |
| Target PCA [34] | 0.615 | 0.632 | 0.583 | 0.541 | 0.651 | 0.632 | 0.541 | 0.599 |
| ITL [30] | 0.745 | 0.727 | 0.653 | 0.597 | 0.721 | 0.714 | 0.634 | 0.684 |
| SGF+LDA[14] | 0.716 | 0.714 | 0.669 | 0.629 | 0.735 | 0.748 | 0.629 | 0.691 |
| GFK (PCA,LDA)[13] | 0.751 | 0.754 | 0.699 | 0.615 | 0.767 | 0.761 | 0.624 | 0.710 |
| Landmarks (PCA,LDA)[12] | 0.747 | 0.759 | 0.701 | 0.640 | 0.763 | 0.763 | 0.642 | 0.716 |
| LTSL (LDA)[29] | 0.736 | 0.735 | 0.698 | 0.642 | 0.752 | 0.767 | 0.640 | 0.710 |
| AE+LDA[11] | 0.735 | 0.708 | 0.702 | 0.656 | 0.746 | 0.739 | 0.649 | 0.705 |
| **BAE+LDA(Ours)** | **0.795** | **0.794** | **0.763** | **0.698** | **0.803** | **0.796** | **0.693** | **0.763** |

ers the landmarks and use them to bridge the source domain to target domain by constructing provably easier auxiliary domain adaptation tasks. Same as GFK, the supervised LDA subspace is used as the source subspace, PCA subspace is used as the target subspace, and 1-NN is employed for final classification. The dimension of source subspace and target subspace, the scale of which to select the landmarks and the step for scale are tuned to report the best results.

**LTSL** [29]. Low-rank transfer subspace learning projects both source and target data to a generalized subspace where each target sample can be represented by source samples with low-rank constraint. For LTSL, the version with LDA objective is used for comparison. The dimension of PCA, LDA, and the LTSL projection are tuned to report the best results.

**AE** [11]. The auto-encoder attempts to reconstruct the input itself and is trained with all samples from both source and target domains. The hidden representation is expected to characterize some commonality of both domains. As in [11], the hidden representation is used as the domain-invariant feature, and LDA + 1-NN classifier is applied for classification. The number of neurons in hidden layer and the dimension of LDA are tuned to report the best results.

**BAE (Ours)**. Our bi-shifting auto-encoder attempts to shift samples between domains, and the shifted source domain, *i.e.*, $\mathbf{G}_t = \mathbf{g}_t(\mathbf{f}_c(\mathbf{X}_s))$, share similar distribution as target domain. Base on $(\mathbf{G}_t, \mathbf{y}_s)$, LDA + 1-NN classifier is applied for classification. For all experiments, the number of hidden neurons of BAE is set as 1000, the sparsity parameter $\gamma$ and the dimension of LDA are tuned. The influence of the parameters in scenario of domain adaption across ethnicity are investigated in Fig. 3, of which the experimental setting is same as that in section 3.3. As seen from Fig. 3, a moderate sparsity $\gamma$ is favorable, as large $\gamma$ means too few samples selected for the reconstruction leading to information loss while small $\gamma$ means too many samples selected for the reconstruction leading to loose enforce of the domain consistency. For both AE and BAE, sigmoid activation function is employed.

Table 2. An exemplar illustration of training and testing settings.

| Scenario | Training | | Testing | |
|---|---|---|---|---|
| | Source | Target | Gallery | Probe |
| View angle | -45° | 0° | 0° | 0° |
| Ethnicity | Mon | Cau | Cau | Cau |
| | Cau | Mon | Mon | Mon |
| Imaging sensor | VIS | NIR | VIS | NIR |
| | NIR | VIS | NIR | VIS |

In all experiments, the face images are aligned according to the manually labeled eye locations, and then normalized to 40x32 pixels on MultiPIE [16], XM2VTS [22] and OFD [37] datasets, but to 32x32 pixels on BUAA dataset [17] as suggested. After this normalization, each image is represented as a column vector by stacking its raw pixels.

For all methods, the training samples consists of two subsets, labeled source domain and unlabeled target domain, and the testing samples are compromised two subsets, gallery set with known identities and probe set of which each sample needs to classify as one identity of the gallery. The probe samples are from the target domain, but the gallery samples can be from either source domain or target domain. The performance is evaluated in terms of rank-1 recognition rate, *i.e.*, the percentage of correctly identified probe samples. For experiments of adaptation across view angle and ethnicity, the gallery samples are from target domain and for experiments of adaptation across imaging sensor, the gallery samples are from source domain. The settings of the three experiments are shown in Table 2.

### 3.1. Domain adaptation across view angle

Domain adaptation across view angle endeavors to adapt the knowledge from one view to another. For this evaluation, the MultiPIE dataset [16] is exploited. It contains images of 337 subjects under various poses, illuminations and expressions. Specifically, a subset including images from all subjects at 7 poses (-45°, -30°, -15°, 0°, 15°, 30°, 45°), with 3 expressions (Neutral, Smile, Disgust), under no flush illumination from 4 collecting sessions is selected as the evaluation dataset. This evaluation dataset is divided into 7 subsets according to view angle. For each view angle, the images of first 200 subjects with about 7 randomly selected images per subject are used for training, and the images of the remaining 137 subjects are used for testing. Among

Table 3. Evaluation of domain adaptation across ethnicity.

| | Cau→Mon | Mon→Cau | Average |
|---|---|---|---|
| Source LDA [1] | 0.679 | 0.676 | 0.678 |
| ITL [30] | 0.801 | 0.775 | 0.788 |
| SGF+LDA[14] | 0.790 | 0.751 | 0.771 |
| GFK (PCA,LDA)[13] | 0.738 | 0.721 | 0.730 |
| Landmarks (PCA,LDA)[12] | 0.718 | 0.763 | 0.741 |
| LTSL (LDA)[29] | 0.791 | 0.793 | 0.792 |
| AE+LDA[11] | 0.784 | 0.786 | 0.785 |
| **BAE+LDA(Ours)** | **0.892** | **0.826** | **0.859** |

Table 4. Evaluation of domain adaptation across imaging sensor.

| | VIS→NIR | NIR→VIS | Average |
|---|---|---|---|
| Source LDA [1] | 0.816 | 0.779 | 0.798 |
| ITL [30] | 0.858 | 0.877 | 0.868 |
| SGF+LDA[14] | 0.841 | 0.832 | 0.837 |
| GFK (PCA,LDA)[13] | 0.850 | 0.867 | 0.859 |
| Landmarks (PCA,LDA)[12] | 0.859 | 0.871 | 0.865 |
| LTSL (LDA)[29] | 0.868 | 0.878 | 0.873 |
| AE+LDA[11] | 0.827 | 0.846 | 0.837 |
| **BAE+LDA(Ours)** | **0.904** | **0.920** | **0.912** |

the testing images, 1 and 4 images per subject are randomly selected as the gallery and probe images respectively. In summary, for each view angle, 1,383 images from 200 subjects are used as the training set, 137 images from the rest 137 subjects are used as gallery, and 553 images from 137 subjects are used as probes. For each evaluation, the training set with class label from a source view and the training set without label from a target view are used for training, the gallery and probe sets from target view are used for testing.

The evaluation results are shown in Table 1. As seen from these comparisons, PCA and Source LDA performs the worst as no supervised information or no adaptation is employed. As expected, the domain adaptation methods, *e.g.*, SGF and ITL, achieve much better performance as they exploit the knowledge from both source and target domain via common subspace or domain-invariant feature representation. Furthermore, the GFK and Landmarks outperform SGF, benefited from integrating an infinite number of subspaces. Although LTSL is a linear method, it also performs promising benefited from the low-rank constraint. The Auto-Encoder methods performs better than Source LDA, but slightly worse than GFK, Landmarks and LTSL, as it does not reduce the domain disparity explicitly and thus can not promise a discriminative commonality. Compared with these method, our BAE performs the best with an improvement up to 4.7% on average, as the non-linearity of auto-encoder coupled with sparse representation constraint can ensure that the shifted source domain follow similar distribution as target domain. The distribution of different layers of BAE is shown in Fig. 4 and some shifted source domain samples can be found in Fig. 5.

### 3.2. Domain adaptation across ethnicity

In many cases, the training samples are from one ethnicity, but the testing samples to classify are from another ethnicity. To explore domain adaptation across ethnicity, the XM2VTS dataset [22] consisting of mainly Caucasian and the Oriented Face Dataset (OFD) [37] consisting of mainly Mongolian are used. The XM2VTS dataset contains 3,440 images of 295 subjects taken over a period of four months with different pose and illumination variations. Eight images with slight pose variation per subject are randomly selected for evaluation. Specifically, for each subject, 4 of the 8 images are randomly selected to form the training set, and the remaining images form the testing set: for each subject, 1 image is enrolled into gallery and the left 3 are used as probes. For OFD dataset, a subset consisting of 800 subjects with 4 images per subject under slight lighting variations are used. The images of the first 400 subjects are used as training data, and images of the rest 400 subjects are used for testing. Specifically, 1 image per subject is randomly selected to form the gallery, and the rest 3 images of each subject are used as the probes.

In summary, for XM2VTS, 1,180 images from 295 subjects are used as training data, 295 images from 295 subjects are used as gallery, and 885 images from 295 subjects are used as probe. For OFD, 1,600 images from 400 subjects are used as training data, 400 images from 400 subjects (one per subject) are used as gallery, and 1,200 images from 400 subjects are used as probes. For both datasets, if one is used as source domain, the training set of this dataset is used along with class label; if the dataset is used as the target domain, the training set is used without class label, the gallery and probe set are used for testing.

The evaluation results are shown in Table 3. As seen, Source LDA performs the worst, SGF, GFK, AE, and Landmarks perform much better as expected. Besides, ITL and LTSL perform even better and the reason we guess is that the discrepancy caused by ethnicity is smaller than view angle which is easier to handle by linear model. Our BAE can further improve the performance up to 5.9% compared with the best performer LTSL. This demonstrates that the discrepancy between the shifted source domain from our BAE and target domain is smaller than that of those domain-

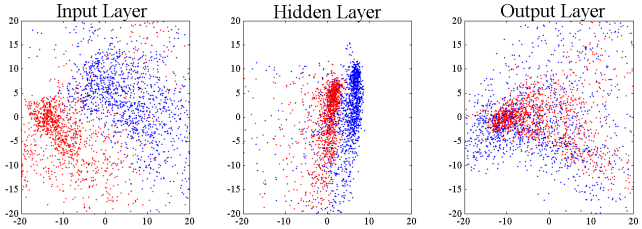


Figure 4. The distribution of the first two dimensions from Principal Component Analysis (PCA) projections of input layer, hidden layer, and output layer of BAE on MultiPIE (-30° →15°).

invariant feature from the other methods. Some shifted source domain samples can be found in Fig. 6(a) and Fig. 6(b). Although it is unknown what the shifted source domain samples should look like when transformed into another ethnicity, the shifted source domain samples shown in Fig. 6(a) and Fig. 6(b) seem reasonable as they look alike the target domain samples with crucial characteristics kept as the source domain, such as eyebrows, eyes, and so on.

### 3.3. Domain adaptation across imaging sensor

For face recognition, another important factor that can cause the distribution different is the imaging sensor, *e.g.*, the images captured from visual light (VIS) sensor look different from the images captured from near-infrared light (NIR) sensor. The BUAA dataset [17] is used for evaluating domain adaptation across imaging sensor. The BUAA dataset has 150 subjects, with VIS images and NIR images captured simultaneously. To simulate a real-world scenario, 675 randomly selected VIS images and another 675 NIR images with different variations in pose or expression are used as the evaluation set. Overall, the 675 VIS images and another 675 NIR images from 150 subjects are used as for training and also for testing, in which VIS used as gallery and NIR used as probe, or vice visa.

The evaluation results are shown in Table 4. Similar conclusion can be obtained as that from domain adaptation across ethnicity. Source LDA performs the worst, AE, SGF, GFK, ITL and Landmarks perform better benefited from the consideration of reducing the discrepancy between domains. Furthermore, our BAE outperform all the other methods, demonstrating the effectiveness of shifting samples between domains. Some shifted source domain samples can be found in Fig. 6(c) and Fig. 6(d).

## 4. Conclusions

In this work, we propose a bi-shifting auto-encoder network (BAE), which attempts to shift the samples from one domain to another domain. The nonlinearity of BAE make it feasible to shift the samples between domains which may depart far from each other, and the sparse representation constraint ensures that the shifted source domain from BAE share similar structure as the desirable target domain. As evaluated on three face domain adaptation scenarios, *i.e.*, domain adaptation across view angle, domain adaptation across ethnicity and domain adaptation across imaging sensor respectively, the proposed BAE outperforms the existing methods, and demonstrate that BAE can shift samples between domains and thus effectively deal with the domain discrepancy.

Figure 5. Exemplar of input image (top) and shifted image (bottom) from BAE, with input and target view angle as (-45° →0°), (-30° →15°), (-15° →30°), (0° →45°), (-15° →15°), and (15° →45°) respectively.

## Acknowledgements

This work was partially supported by 973 Program under contract No. 2015CB351802, Natural Science Foundation of China under contracts Nos. 61173065, 61222211, 61402443 and 61390511.

## References

[1] P. N. Belhumeur, J. P. Hespanha, and D. J. Kriegman. Eigenfaces vs. fisherfaces: recognition using class specific linear projection. *IEEE Transactions on Pattern Analysis and Machine Intelligence (T-PAMI)*, 19(7):711–720, 1997.

[2] S. Ben-David, J. Blitzer, K. Crammer, and F. Pereira. Analysis of representations for domain adaptation. In *Advances in Neural Information Processing Systems (NIPS)*, volume 19, pages 137–144, 2007.

[3] Y. Bengio. Learning deep architectures for ai. *Foundations and trends® in Machine Learning*, 2(1):1–127, 2009.

[4] S. Bickel, M. Brückner, and T. Scheffer. Discriminative learning under covariate shift. *The Journal of Machine Learning Research (JMLR)*, 10:2137–2155, 2009.

[5] J. Blitzer, R. McDonald, and F. Pereira. Domain adaptation with structural correspondence learning. In *Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 120–128, 2006.

[6] L. Bruzzone and M. Marconcini. Domain adaptation problems: a dasvm classification technique and a circular validation strategy. *IEEE Transactions on Pattern Analysis and Machine Intelligence (T-PAMI)*, 32(5):770–787, 2010.

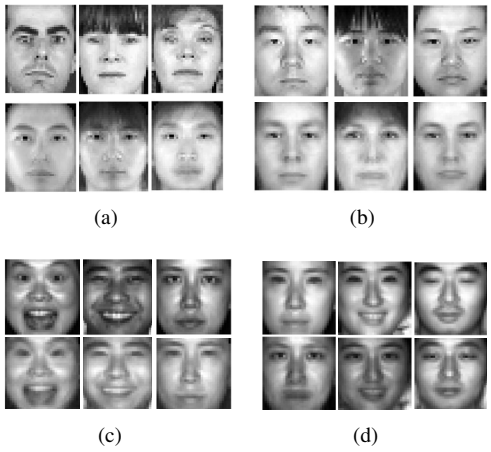


Figure 6. Exemplar of input image (top) and shifted image (bottom) from BAE across ethnicity and imaging sensor, with input and target domain as (a) Cau→Mon, (b) Mon→Cau, (c) VIS→NIR, and (d) NIR→VIS, respectively.

[7] Y. Chen, G. Wang, and S. Dong. Learning with progressive transductive support vector machine. *Pattern Recognition Letters (PRL)*, 24(12):1845–1855, 2003.

[8] L. Duan, D. Xu, I. Tsang, and J. Luo. Visual event recognition in videos by learning from web data. *IEEE Transactions on Pattern Analysis and Machine Intelligence (T-PAMI)*, 34(9):1667–1680, 2012.

[9] B. Efron, T. Hastie, I. Johnstone, and R. Tibshirani. Least angle regression. *Annals of Statistics*, 39(4):407–499, 2004.

[10] B. Geng, D. Tao, and C. Xu. Daml: Domain adaptation metric learning. *IEEE Transactions on Image Processing (T-IP)*, 20(10):2980–2989, 2011.

[11] X. Glorot, A. Bordes, and Y. Bengio. Domain adaptation for large-scale sentiment classification: A deep learning approach. In *International Conference on Machine Learning (ICML)*, pages 513–520, 2011.

[12] B. Gong, K. Grauman, and F. Sha. Connecting the dots with landmarks: Discriminatively learning domain-invariant features for unsupervised domain adaptation. In *International Conference on Machine Learning (ICML)*, pages 222–230, 2013.

[13] B. Gong, Y. Shi, F. Sha, and K. Grauman. Geodesic flow kernel for unsupervised domain adaptation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2066–2073, 2012.

[14] R. Gopalan, R. Li, and R. Chellappa. Domain adaptation for object recognition: an unsupervised approach. In *IEEE International Conference on Computer Vision (ICCV)*, pages 999–1006, 2011.

[15] A. Gretton, A. Smola, J. Huang, M. Schmittfull, K. Borgwardt, and B. Schölkopf. Covariate shift by kernel mean matching. *Dataset shift in machine learning*, pages 131–160, 2009.

[16] R. Gross, I. Matthews, J. Cohn, T. kanada, and S. Baker. The cmu multi-pose, illumination, and expression (multi-pie) face database. Technical report, Carnegie Mellon University Robotics Institute. TR-07-08, 2007.

[17] D. Huang, J. Sun, and Y. Wang. The buaa-visnir face database instructions, 2012.

[18] J. Huang, A. J. Smola, A. Gretton, K. M. Borgwardt, and B. Schölkopf. Correcting sample selection bias by unlabeled data. In *Advances in Neural Information Processing Systems (NIPS)*, 2006.

[19] H. D. III and D. Marcu. Domain adaptation for statistical classifiers. *Journal of Artificial Intelligence Research (JAIR)*, pages 101–126, 2006.

[20] I.-H. Jhuo, D. Liu, D. T. Lee, and S.-F. Chang. Robust visual domain adaptation with low-rank reconstruction. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2168–2175, 2012.

[21] M. Kan, J. Wu, S. Shan, and X. Chen. Domain adaptation for face recognition: Targetize source domain bridged by common subspace. *International Journal of Computer Vision*, 109(1-2):94–109, 2014.

[22] K. Messer, M. Matas, J. Kittler, J. Lttin, and G. Maitre. Xm2vtsdb: The extended m2vts database. In *International Conference on Audio and Video-based Biometric Person Authentication (AVBPA)*, pages 72–77, 1999.

[23] S. J. Pan, J. T. Kwok, and Q. Yang. Transfer learning via dimensionality reduction. In *AAAI Conference on Artificial Intelligence (AAAI)*, pages 677–682, 2008.

[24] S. J. Pan, I. W. Tsang, J. T. Kwok, and Q. Yang. Domain adaptation via transfer component analysis. *IEEE Transactions on Neural Networks (T-NN)*, 22(2):199–210, 2011.

[25] S. J. Pan and Q. Yang. A survey on transfer learning. *IEEE Transactions on Knowledge and Data Engineering (T-KDE)*, 22(10):1345–1359, 2010.

[26] Q. Qiu, V. M. Patel, P. Turaga, and R. Chellappa. Domain adaptive dictionary learning. In *European Conference on Computer Vision (ECCV)*, pages 631–645, 2012.

[27] R. Raina, A. Battle, H. Lee, B. Packer, and A. Y. Ng. Self-taught learning: transfer learning from unlabeled data. In *International Conference on Machine Learning (ICML)*, pages 759–766, 2007.

[28] M. Shao, C. Castillo, Z. Gu, and Y. Fu. Low-rank transfer subspace learning. In *IEEE International Conference on Data Mining (ICDM)*, pages 1104–1109, 2012.

[29] M. Shao, D. Kit, and Y. Fu. Generalized transfer subspace learning through low-rank constraint. *International Journal of Computer Vision (IJCV)*, 109(1-2):74–93, 2014.

[30] Y. Shi and F. Sha. Information-theoretical learning of discriminative clusters for unsupervised domain adaptation. In *International Conference on Machine Learning (ICML)*, 2012.

[31] Shimodaira and Hidetoshi. Improving predictive inference under covariate shift by weighting the log-likelihood function. *Journal of Statistical Planning and Inference*, 90(2):227–244, 2000.

[32] M. Sugiyama, S. Nakajima, H. Kashima, P. V. Buenau, and M. Kawanabe. Direct importance estimation with model selection and its application to covariate shift adaptation. In *Advances in Neural Information Processing Systems (NIPS)*, volume 20, pages 1433–1440, 2008.

[33] M. Sugiyamai, M. Krauledat, and K.-R. Müller. Covariate shift adaptation by importance weighted cross validation. *The Journal of Machine Learning Research (JMLR)*, 8:985–1005, 2007.

[34] M. A. Turk and A. P. Pentland. Face recognition using eigenfaces. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, volume 591, pages 586–591, 1991.

[35] P. Vincent, H. Larochelle, I. Lajoie, Y. Bengio, and P.-A. Manzagol. Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion. *The Journal of Machine Learning Research (JMLR)*, 9999:3371–3408, 2010.

[36] R. Wang, S. Shan, X. Chen, and W. Gao. Manifold-manifold distance with application to face recognition based on image set. In *IEEE International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2008.

[37] U. XianJiaotong. http://www.aiar.xjtu.edu.cn/groups/face/Chinese/HomePage.htm, 2006.

[38] Zadrozny and Bianca. Learning and evaluating classifiers under sample selection bias. In *Proceedings of International Conference on Machine Learning (ICML)*, pages 114–114, 2004.