

Learning Cross-Domain Landmarks for Heterogeneous Domain Adaptation

Yao-Hung Hubert Tsai¹, Yi-Ren Yeh², Yu-Chiang Frank Wang¹

¹Research Center for IT Innovation, Academia Sinica, Taipei, Taiwan

²Department of Mathematics, National Kaohsiung Normal University, Kaohsiung, Taiwan

y.h.huberttsai@gmail.com, yryeh@nknu.edu.tw, ycwang@citi.sinica.edu.tw

Abstract

While domain adaptation (DA) aims to associate the learning tasks across data domains, heterogeneous domain adaptation (HDA) particularly deals with learning from cross-domain data which are of different types of features. In other words, for HDA, data from source and target domains are observed in separate feature spaces and thus exhibit distinct distributions. In this paper, we propose a novel learning algorithm of Cross-Domain Landmark Selection (CDLS) for solving the above task. With the goal of deriving a domain-invariant feature subspace for HDA, our CDLS is able to identify representative cross-domain data, including the unlabeled ones in the target domain, for performing adaptation. In addition, the adaptation capabilities of such cross-domain landmarks can be determined accordingly. This is the reason why our CDLS is able to achieve promising HDA performance when comparing to state-of-the-art HDA methods. We conduct classification experiments using data across different features, domains, and modalities. The effectiveness of our proposed method can be successfully verified.

1. Introduction

When solving standard machine learning and pattern recognition tasks, one needs to collect a sufficient amount of labeled data for learning purposes. However, it is not always computationally feasible to collect and label data for each problem of interest. To alleviate this concern, domain adaptation (DA) aims to utilize labeled data from a source domain, while only few (or no) labeled data can be observed in the target domain of interest [27]. In other words, the goal of DA is to transfer the knowledge learned from an auxiliary domain, so that the learning task in the target domain can be solved accordingly.

A variety of real-world applications have been benefited from the recent advances of DA (e.g., sentiment analysis [7, 33], Wi-Fi localization [25, 26], visual object classification [43, 30, 34, 8], and cross-language text classification [10, 29]). Generally, most existing DA approaches

like [26, 24, 31, 6] consider that source and target-domain data are collected using the same type of features. Such problems can be referred to *homogeneous* domain adaptation, i.e., cross-domain data are observed in the same feature space but exhibit different distributions.

On the other hand, *heterogeneous* domain adaptation (HDA) [32, 38, 21, 28] deals with a even more challenging task, in which cross-domain data are described by different types of features and thus exhibit distinct distributions (e.g., training and test image data with different resolutions or encoded by different codebooks). Most existing approaches choose to solve HDA by learning feature transformation, which either projects data from one domain to the other, or to project cross-domain data to a common subspace for adaptation purposes [32, 38, 21, 12, 18, 42, 19].

As noted in [39, 23, 40, 41], exploiting unlabeled target-domain instances during adaptation would be beneficial for HDA. Following their settings, we choose to address *semi-supervised* HDA in this paper. To be more precise, a sufficient amount and few labeled data will be observed in the source and target domains, respectively. And, the remaining target-domain data (to be classified) will also be presented during the learning process. In our work, we propose a learning algorithm of *Cross-Domain Landmark Selection (CDLS)* for solving HDA. Instead of viewing all cross-domain data to be equally important during adaptation, our CDLS derives a heterogeneous feature transformation which results in a domain-invariant subspace for associating cross-domain data. In addition, the representative source and target-domain data will be jointly exploited for improving the adaptation capability of our CDLS. Once the adaptation process is complete, one can simply project cross-domain labeled and unlabeled target domain data into the derived subspace for performing recognition. Illustration of our CDLS is shown in Figure 1.

The contributions of this paper are highlighted below:

- We are among the first to exploit heterogenous source and target-domain data for learning cross-domain landmarks, aiming at solving semi-supervised HDA problems.

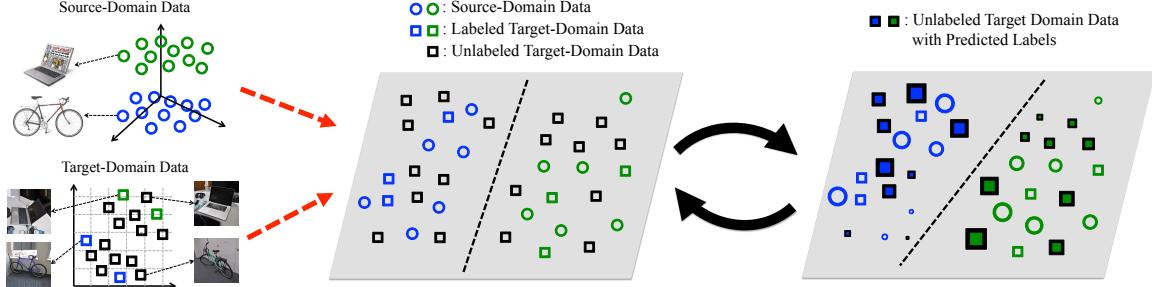


Figure 1. Illustration of Cross-Domain Landmark Selection (CDLS) for heterogeneous domain adaptation (HDA). Note that instances shown in different sizes indicate cross-domain landmarks with different adaptation abilities.

- By learning the adaptability of cross-domain data (including the unlabeled target-domain ones), our CDLS derives a domain-invariant feature space for improved adaptation and classification performances.
- Experiments on classification tasks using data across different features, datasets, and modalities data are considered. Our CDLS is shown to perform favorably against state-of-the-art HDA approaches.

2. Related Work

Most existing HDA methods consider a supervised setting, in which only labeled source and target-domain data are presented during adaptation. Generally, these HDA approaches can be divided into two categories. The first group of them derive a pair of feature transformation (one for each domain), so that source and target-domain data can be projected into a common feature space for adaptation and recognition [32, 38, 12, 39, 23, 41]. For example, Shi *et al.* [32] proposed heterogeneous spectral mapping (HeMap), which selects source-domain instances for relating cross-domain data via spectral embedding. Wang and Mahadevan [38] chose to solve domain adaptation by manifold alignment (DAMA), with the goal of preserving label information during their alignment/adaptation process. Duan *et al.* [12] proposed heterogeneous feature augmentation (HFA) for learning a common feature subspace, in which SVM classifiers can be jointly derived for performing recognition.

On the other hand, the second category of existing HDA methods learn a mapping matrix which transforms the data from one domain to another [21, 18, 19, 42, 40]. For example, Kulis *et al.* [21] proposed asymmetric regularized cross-domain transformation (ARC-t) for associating cross-domain data with label information guarantees. Hoffman *et al.* [18, 19] considered a max-margin domain transformation (MMDT) approach, which adapts the observed prediction models (i.e., SVMs) across domains. Similarly, Zhou *et al.* [42] presented an algorithm of sparse heterogeneous feature representation (SHFR), which relate the predictive structures produced by the prediction models across do-

mains for addressing cross-domain classification problems.

Nevertheless, the above HDA approaches only considered labeled source and target-domain data during adaption. As noted in literature on homogeneous domain adaptation [13, 26, 22], unlabeled target-domain data can be jointly exploited for adaptation. Such semi-supervised settings have been shown to be effective for DA. Recent HDA approaches like [39, 23, 40, 41] also benefit from this idea. For example, inspired by canonical correlation analysis, Wu *et al.* [39] presented an algorithm of heterogeneous transfer discriminant-analysis of canonical correlations (HTDCC). Their objective is to minimize canonical correlation between inter-class samples, while that between the intra-class ones can be maximized. Li *et al.* [23] extended HFA [12] to a semi-supervised version (i.e., SHFA), and further exploited projected unlabeled target-domain data into their learning process. Recently, Xiao and Guo [40] proposed semi-supervised kernel matching for domain adaptation (SSKMDA), which determines a feature space while preserving cross-domain data locality information. They also proposed a semi-supervised HDA approach of subspace co-projection (SCP) in [41], and chose to minimize the divergence between cross-domain features and their prediction models. However, both SSKMDA and SCP require additional unlabeled source-domain data for HDA.

It is worth noting that, techniques of instance reweighting or landmark selection have been applied for solving domain adaptation problems [20, 9, 15, 1]. Existing methods typically weight or select source-domain data, and only homogeneous DA settings are considered. Moreover, no label information from either domain is taken into consideration during the adaptation process. Based on the above observations, we address semi-supervised HDA by proposing CDLS for learning representative landmarks from cross-domain data. Our CDLS is able to identify the contributions of each landmark when matching cross-domain class-conditional data distributions in the derived feature space. This is why improved HDA performance can be expected (see our experiments in Section 4).

3. Our Proposed Method

3.1. Problem Settings and Notations

Let $\mathcal{D}_S = \{\mathbf{x}_s^i, y_s^i\}_{i=1}^{n_S} = \{\mathbf{X}_S, \mathbf{y}_S\}$ denote source-domain data, where each instance $\mathbf{x}_s \in \mathbb{R}^{d_S}$ is assigned with label $y_s \in \mathcal{L} = \{1, \dots, C\}$ (i.e., a C -cardinality label set). For HDA with a semi-supervised setting, we let $\mathcal{D}_L = \{\mathbf{x}_l^i, y_l^i\}_{i=1}^{n_L} = \{\mathbf{X}_L, \mathbf{y}_L\}$ and $\mathcal{D}_U = \{\mathbf{x}_u^i, y_u^i\}_{i=1}^{n_U} = \{\mathbf{X}_U, \mathbf{y}_U\}$ as labeled and unlabeled target-domain data, respectively. We have $\mathbf{x}_l, \mathbf{x}_u \in \mathbb{R}^{d_T}$ and $y_l, y_u \in \mathcal{L}$. Note that $d_S \neq d_T$, since source and target-domain data are of different types of features. In a nutshell, for solving semi-supervised HDA, we observe d_S -dimensional source-domain data D_S , few labeled d_T -dimensional target-domain data D_L , and all unlabeled target-domain data D_U (to be recognized) during adaptation. Our goal is to predict the labels \mathbf{y}_U for \mathbf{X}_U .

For semi-supervised HDA, only few labeled data are presented in the target domain, while the remaining unlabeled target-domain instances are seen during the adaptation process. Nevertheless, a sufficient number of labeled data will be available in the source domain for performing adaptation. Different from [40, 41], we do *not* require any additional unlabeled source-domain data for adaptation.

3.2. Cross-Domain Landmark Selection

3.2.1 Matching cross-domain data distributions

Inspired by the recent HDA approaches of [21, 18, 19, 42, 40], we learn a feature transformation \mathbf{A} which projects source-domain data to the subspace derived from the target-domain data. In this derived subspace, our goal is to associate cross-domain data with different feature dimensionality and distinct distributions. Such association is achieved by eliminating the domain bias via matching cross-domain data distributions (i.e., marginal distributions $P_T(\mathbf{X}_T)$ and $P_S(\mathbf{A}^\top \mathbf{X}_S)$, and the conditional ones $P_T(\mathbf{y}_T|\mathbf{X}_T)$ and $P_S(\mathbf{y}_S|\mathbf{A}^\top \mathbf{X}_S)$). Unfortunately, direct estimation of $P_T(\mathbf{y}_T|\mathbf{X}_T)$ and $P_S(\mathbf{y}_S|\mathbf{A}^\top \mathbf{X}_S)$ is not possible. As an alternative solution suggested in [24], we choose to match the estimated $P_T(\mathbf{X}_T|\mathbf{y}_T)$ and $P_S(\mathbf{A}^\top \mathbf{X}_S|\mathbf{y}_S)$ by Bayes' Theorem [4].

As noted above, we first project all target-domain data into a m -dimensional subspace via PCA (note that $m \leq \min\{d_S, d_T\}$). In the remaining of this paper, we use $\hat{\mathbf{x}}_l$ and $\hat{\mathbf{x}}_u$ to denote reduced-dimensional labeled and unlabeled target-domain data, respectively. As a result, the linear transformation to be learned is $\mathbf{A} \in \mathbb{R}^{m \times d_S}$. It is worth noting that, this dimension reduction stage is to prevent possible overfitting caused by mapping low-dimensional data in a high-dimensional space. Later in our experiments, we will vary the dimension number m for the completeness of our evaluation.

Recall that, as discussed in Section 2, most existing HDA works consider a supervised setting. That is, only labeled data in the target domain are utilized during training. In order to match cross-domain data distributions for such standard supervised HDA settings, we formulate the problem formulation as follows:

$$\min_{\mathbf{A}} E_M(\mathbf{A}, \mathcal{D}_S, \mathcal{D}_L) + E_C(\mathbf{A}, \mathcal{D}_S, \mathcal{D}_L) + \lambda \|\mathbf{A}\|^2, \quad (1)$$

where E_M and E_C measure the differences between cross-domain marginal and conditional data distributions, respectively. The last term in (1) is penalized by λ to prevent overfitting \mathbf{A} .

To solve (1), we resort to statistics to measure the probability density for describing data distributions. As suggested in [26, 24], we adopt empirical *Maximum Mean Discrepancy* (MMD) [16] to measure the difference between the above cross-domain data distributions. As a result, $E_M(\mathbf{A}, \mathcal{D}_S, \mathcal{D}_L)$ can be calculated as

$$E_M(\mathbf{A}, \mathcal{D}_S, \mathcal{D}_L) = \left\| \frac{1}{n_S} \sum_{i=1}^{n_S} \mathbf{A}^\top \mathbf{x}_s^i - \frac{1}{n_L} \sum_{i=1}^{n_L} \hat{\mathbf{x}}_l^i \right\|^2. \quad (2)$$

For matching cross-domain conditional data distributions, we determine $E_C(\mathbf{A}, \mathcal{D}_S, \mathcal{D}_L)$ as:

$$\begin{aligned} E_C(\mathbf{A}, \mathcal{D}_S, \mathcal{D}_L) &= \sum_{c=1}^C \left\| \frac{1}{n_S^c} \sum_{i=1}^{n_S^c} \mathbf{A}^\top \mathbf{x}_s^{i,c} - \frac{1}{n_L^c} \sum_{i=1}^{n_L^c} \hat{\mathbf{x}}_l^{i,c} \right\|^2 \\ &\quad + \frac{1}{n_S^c n_L^c} \sum_{i=1}^{n_S^c} \sum_{j=1}^{n_L^c} \left\| \mathbf{A}^\top \mathbf{x}_s^{i,c} - \hat{\mathbf{x}}_l^{j,c} \right\|^2, \end{aligned} \quad (3)$$

where $\hat{\mathbf{x}}_l^{i,c}$ indicates the labeled target-domain instance of class c , and n_L^c denotes the number of labeled target-domain instances in that class. Similarly, $\mathbf{x}_s^{i,c}$ is the source-domain data of class c , and n_S^c represents the number of source-domain data of class c .

In (3), the first term calculates the difference between class-wise means for matching the approximated class-conditional distributions (see [24] for detailed derivations), while the second term enforces the embedding of transformed within-class data [37]. We note that, since only labeled cross-domain data are utilized in (1), its solution for HDA can be considered as a supervised version of our CDLS (denoted as CDLS_{sup} in Section 4).

3.2.2 From supervised to semi-supervised HDA

To adopt the information of unlabeled target-domain data for improved adaptation, we advocate the learning of the landmarks from cross-domain data when deriving the aforementioned domain-invariant feature subspace.

Extended from (1), our proposed algorithm cross-domain landmark selection (CDLS) exploits heterogenous data across domains, with the ability to identify the adaptation ability of each instance with a properly assigned weight. Instances in either domain with a nonzero weight will be considered as a landmark.

With the above goal, the objective function of our CDLS can be formulated as follows:

$$\begin{aligned} \min_{\mathbf{A}, \boldsymbol{\alpha}, \boldsymbol{\beta}} & E_M(\mathbf{A}, \mathcal{D}_S, \mathcal{D}_L, \mathbf{X}_U, \boldsymbol{\alpha}, \boldsymbol{\beta}) + \\ & E_C(\mathbf{A}, \mathcal{D}_S, \mathcal{D}_L, \mathbf{X}_U, \boldsymbol{\alpha}, \boldsymbol{\beta}) + \lambda \|\mathbf{A}\|^2, \\ \text{s.t. } & \{\alpha_i^c, \beta_i^c\} \in [0, 1], \frac{\boldsymbol{\alpha}^{c\top} \mathbf{1}}{n_S^c} = \frac{\boldsymbol{\beta}^{c\top} \mathbf{1}}{n_U^c} = \delta, \end{aligned} \quad (4)$$

where $\boldsymbol{\alpha} = [\boldsymbol{\alpha}^1; \dots; \boldsymbol{\alpha}^c; \dots; \boldsymbol{\alpha}^C] \in \mathbb{R}^{n_S}$ and $\boldsymbol{\beta} = [\boldsymbol{\beta}^1; \dots; \boldsymbol{\beta}^c; \dots; \boldsymbol{\beta}^C] \in \mathbb{R}^{n_U}$ are the weights observed for all labeled and unlabeled data in source and target domains, respectively. We have $\boldsymbol{\alpha}^c = [\alpha_1^c; \dots; \alpha_{n_S^c}^c]$ and $\boldsymbol{\beta}^c = [\beta_1^c; \dots; \beta_{n_U^c}^c]$, where n_S^c and n_U^c denote the total numbers of the corresponding instances of or predicted as class c in the associated domain.

In (4), $\delta \in [0, 1]$ controls the portion of cross-domain data in each class to be utilized for adaptation. If $\delta = 0$, our CDLS would turn into its supervised version as described in Section 3.2.1. While we fix $\delta = 0.5$ in our work, additional analysis on the selection and effect of δ will be provided in our experiments. We also note that, since the number of labeled target-domain instances \mathbf{X}_L is typically small in semi-supervised HDA, all of such data will be viewed as the most representative landmarks to be utilized during adaptation (i.e., no additional weights required for \mathbf{X}_L).

The E_M in (4) matches marginal cross-domain data distributions for HDA. With $\hat{\mathbf{x}}_l$ and $\hat{\mathbf{x}}_u$ indicating reduced-dimensional labeled and unlabeled target-domain data, we calculate E_M by:

$$E_M(\mathbf{A}, \mathcal{D}_S, \mathcal{D}_L, \mathbf{X}_U, \boldsymbol{\alpha}, \boldsymbol{\beta}) = \left\| \frac{1}{\delta n_S} \sum_{i=1}^{n_S} \alpha_i \mathbf{A}^\top \mathbf{x}_s^i - \frac{1}{n_L + \delta n_U} \left(\sum_{i=1}^{n_L} \hat{\mathbf{x}}_l^i + \sum_{i=1}^{n_U} \beta_i \hat{\mathbf{x}}_u^i \right) \right\|^2. \quad (5)$$

To match cross-domain conditional data distributions via E_C , we apply SVM trained from labeled cross-domain data to predict the pseudo-labels $\tilde{\mathbf{y}}_u^i$ for $\hat{\mathbf{x}}_u^i$ (as described later in Section 3.3.3). With $\{\tilde{\mathbf{y}}_u^i\}_{i=1}^{n_U}$ assigned for \mathbf{X}_U , the E_C term in (4) can be expressed as:

$$E_C(\mathbf{A}, \mathcal{D}_S, \mathcal{D}_L, \mathbf{X}_U, \boldsymbol{\alpha}, \boldsymbol{\beta}) = \sum_{c=1}^C E_{cond}^c + \frac{1}{e^c} E_{embed}^c, \quad (6)$$

where

$$\begin{aligned} E_{cond}^c &= \left\| \frac{1}{\delta n_S^c} \sum_{i=1}^{n_S^c} \alpha_i \mathbf{A}^\top \mathbf{x}_s^{i,c} - \frac{1}{n_L^c + \delta n_U^c} \left(\sum_{i=1}^{n_L^c} \hat{\mathbf{x}}_l^{i,c} + \sum_{i=1}^{n_U^c} \beta_i \hat{\mathbf{x}}_u^{i,c} \right) \right\|^2, \\ E_{embed}^c &= \sum_{i=1}^{n_S^c} \sum_{j=1}^{n_U^c} \left\| \alpha_i \mathbf{A}^\top \mathbf{x}_s^{i,c} - \hat{\mathbf{x}}_l^{j,c} \right\|^2 + \sum_{i=1}^{n_L^c} \sum_{j=1}^{n_U^c} \left\| \hat{\mathbf{x}}_l^{i,c} - \beta_j \hat{\mathbf{x}}_u^{j,c} \right\|^2 + \\ &\quad \sum_{i=1}^{n_U^c} \sum_{j=1}^{n_S^c} \left\| \beta_i \hat{\mathbf{x}}_u^{i,c} - \alpha_j \mathbf{A}^\top \mathbf{x}_s^{j,c} \right\|^2, \end{aligned}$$

It can be seen that, E_C term in (6) is extended from (3) by utilizing unlabeled target-domain data with pseudo-labels. Similar to (3), we match cross-domain class-conditional distributions and preserve local embedding of transformed data of each class using via E_{cond}^c and E_{embed}^c , respectively. The normalization term in (6) is calculated as $e^c = \delta n_S^c n_L^c + \delta n_L^c n_U^c + \delta^2 n_U^c n_S^c$.

With both E_M and E_C defined, we address semi-supervised HDA by solving (4). This allows us to learn the proper weights $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ for the representative instances from both domains (i.e., cross-domain landmarks). As a result, our derived feature transformation \mathbf{A} will result in a feature subspace with improved adaptation capability. This will be verified later by our experiments in Section 4.

3.3. Optimization

We now discuss how we solve the optimization problem of (4). Similar optimization process can be applied to solve the supervised version of (1), which is presented in the Supplementary.

It can be seen that, the semi-supervised learning problem of (4) is a non-convex joint optimization problem with respect to \mathbf{A} , $\boldsymbol{\alpha}$, and $\boldsymbol{\beta}$. By advancing the technique of iterative optimization, we alternate between the learning of feature transformation \mathbf{A} and landmark weights $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$. The latter also involves the prediction of pseudo-labels for unlabeled target-domain data. We now describe the optimization process below.

3.3.1 Optimizing A

Given the weights of $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ fixed, we take the derivative of (4) with respect to \mathbf{A} and set it equal to zero. The closed form solution of \mathbf{A} can be derived as:

$$\mathbf{A} = \left(\mathbf{I}_{d_S} + \mathbf{X}_S \mathbf{H}_S \mathbf{X}_S^\top \right)^{-1} \left(\mathbf{X}_S \left(\mathbf{H}_L \hat{\mathbf{X}}_L^\top + \mathbf{H}_U \hat{\mathbf{X}}_U^\top \right) \right), \quad (7)$$

where \mathbf{I}_{d_S} is a d_S -dimensional identity matrix, matrices $\{\mathbf{X}_S \in \mathbb{R}^{d_S \times n_S}, \hat{\mathbf{X}}_L \in \mathbb{R}^{m \times n_L}, \hat{\mathbf{X}}_U \in \mathbb{R}^{m \times n_U}\}$ represent source, transformed labeled and unlabeled target-domain instances, respectively. Each entry $(\mathbf{H}_S)_{i,j}$ in $\mathbf{H}_S \in \mathbb{R}^{n_S \times n_S}$ denotes the derivative coefficient associated with $\mathbf{x}_s^i \top \mathbf{x}_s^j$. Similar remarks can be applied to the matrices $\mathbf{H}_L \in \mathbb{R}^{n_S \times n_L}$ and $\mathbf{H}_U \in \mathbb{R}^{n_S \times n_U}$. Detailed derivations are available in the Supplementary.

Algorithm 1 Cross-Domain Landmark Selection (CDLS)	
Input:	Labeled source and target-domain data $\mathcal{D}_S = \{\mathbf{x}_s^i, \mathbf{y}_s^i\}_{i=1}^{n_S}$, $\mathcal{D}_L = \{\mathbf{x}_l^i, \mathbf{y}_l^i\}_{i=1}^{n_L}$; unlabeled target-domain data $\{\mathbf{x}_u^i\}_{i=1}^{n_U}$; feature dimension m ; ratio δ ; parameter λ
1:	Derive an m -dim. subspace via PCA from $\{\mathbf{x}_l^i\}_{i=1}^{n_L}$ and $\{\mathbf{x}_u^i\}_{i=1}^{n_U}$
2:	Initialize \mathbf{A} by (1) and pseudo-labels $\{\tilde{\mathbf{y}}_u^i\}_{i=1}^{n_U}$
3:	while not converge do
4:	Update transformation \mathbf{A} by (7)
5:	Update landmark weights $\{\alpha, \beta\}$ by (9)
6:	Update pseudo-labels $\{\tilde{\mathbf{y}}_u^i\}_{i=1}^{n_U}$ by Section 3.3.3
7:	end while
Output:	Predicted labels $\{\mathbf{y}_u^i\}_{i=1}^{n_U}$ of $\{\mathbf{x}_u^i\}_{i=1}^{n_U}$

3.3.2 Optimizing α and β

With transformation \mathbf{A} observed, we reformulate (4) as:

$$\begin{aligned} & \min_{\alpha, \beta} \frac{1}{2} \alpha^\top \mathbf{K}_{S,S} \alpha + \frac{1}{2} \beta^\top \mathbf{K}_{U,U} \beta - \alpha^\top \mathbf{K}_{S,U} \beta \\ & \quad - \mathbf{k}_{S,L}^\top \alpha + \mathbf{k}_{U,L}^\top \beta \\ \text{s.t. } & \{\alpha_i^c, \beta_i^c\} \in [0, 1], \frac{\alpha^{c\top} \mathbf{1}}{n_S^c} = \frac{\beta^{c\top} \mathbf{1}}{n_U^c} = \delta. \end{aligned} \quad (8)$$

In (8), each entry $(\mathbf{K}_{S,S})_{i,j}$ in $\mathbf{K}_{S,S} \in \mathbb{R}^{n_S \times n_S}$ is the coefficient associated with $(\mathbf{A}^\top \mathbf{x}_s^i)^\top \mathbf{A}^\top \mathbf{x}_s^j$, and each entry $(\mathbf{k}_{S,L})_i$ in $\mathbf{k}_{S,L} \in \mathbb{R}^{n_S}$ denotes the sum of the coefficients of $(\mathbf{A}^\top \mathbf{x}_s^i)^\top \hat{\mathbf{x}}_l^j$ over all $\hat{\mathbf{x}}_l^j$. Similar remarks can be applied to $\mathbf{K}_{S,U} \in \mathbb{R}^{n_S \times n_U}$, $\mathbf{K}_{U,U} \in \mathbb{R}^{n_U \times n_U}$, and $\mathbf{k}_{U,L} \in \mathbb{R}^{n_U}$ (see the Supplementary for detailed derivations).

With the above formulation, one can apply existing *Quadratic Programming* (QP) solvers and optimize the following problem instead:

$$\min_{z_i \in [0,1], \mathbf{Z}^\top \mathbf{V} = \mathbf{W}} \frac{1}{2} \mathbf{Z}^\top \mathbf{B} \mathbf{Z} + \mathbf{b}^\top \mathbf{Z}, \quad (9)$$

where

$$\mathbf{Z} = \begin{pmatrix} \alpha \\ \beta \end{pmatrix}, \mathbf{B} = \begin{pmatrix} \mathbf{K}_{S,S} & -\mathbf{K}_{S,U} \\ -\mathbf{K}_{S,U}^\top & \mathbf{K}_{U,U} \end{pmatrix}, \mathbf{b} = \begin{pmatrix} -\mathbf{k}_{S,L} \\ \mathbf{k}_{U,L} \end{pmatrix},$$

$$\mathbf{V} = \begin{bmatrix} \mathbf{V}_S & \mathbf{0}_{n_S \times C} \\ \mathbf{0}_{n_U \times C} & \mathbf{V}_U \end{bmatrix} \in \mathbb{R}^{(n_S+n_U) \times 2C} \text{ with}$$

$$(\mathbf{V}_S)_{ij} = \begin{cases} 1 & \text{if } \mathbf{x}_s^i \in \text{class } j \\ 0 & \text{otherwise} \end{cases}$$

$$(\mathbf{V}_U)_{ij} = \begin{cases} 1 & \text{if } \mathbf{x}_u^i \text{ predicted as class } j \\ 0 & \text{otherwise} \end{cases},$$

$$\mathbf{W} \in \mathbb{R}^{1 \times 2C} \text{ with } (\mathbf{W})_c = \begin{cases} \delta n_S^c & \text{if } c \leq C \\ \delta n_U^{c-C} & \text{if } c > C \end{cases}.$$



Figure 2. Example images of Caltech-256 and Office dataset.

3.3.3 Label prediction for \mathbf{X}_U

Given the feature transformation and the observed landmarks, we train linear SVMs [5] using all transformed labeled cross-domain data with the associated weights. As a result, the labels of unlabeled target-domain data $\{\hat{\mathbf{x}}_u^i\}_{i=1}^{n_U}$ can be predicted accordingly.

To start our optimization process, we apply the supervised version of CDLS (i.e., Section 3.2.1) to initialize the transformation \mathbf{A} and the pseudo-labels $\{\tilde{\mathbf{y}}_u^i\}_{i=1}^{n_U}$. The pseudo code of our CDLS is summarized in Algorithm 1.

4. Experiments

4.1. Datasets and Settings

We first address the task of cross-domain object recognition, and consider the use of **Office** + **Caltech-256** (C) datasets [30, 17]. The **Office** dataset contains images collected from three sub-datasets : Amazon (A) (i.e., images downloaded from the Internet), Webcam (W) (i.e., low-resolution images captured by web cameras), and DSLR (D) (i.e., high-resolution images captured by digital SLR cameras). This dataset has objects images of 31 categories, and **Caltech-256** contains 256 object categories. Among these objects, 10 overlapping categories are selected for our experiments. Figure 2 shows example images of the category of *laptop_computer*. Following [14], two different types of features are considered for cross-domain object recognition: *DeCAF*₆ [11] and *SURF* [3]. We note that, the latter is described in terms of Bag-of-Words (BOW) features via k-means clustering. The feature dimensions of *DeCAF*₆ and *SURF* are 4096 and 800, respectively.

We also solve the problem of cross-lingual text categorization, and apply the **Multilingual Reuters Collection** [35, 2] dataset for experiments. This dataset contains about 11,000 articles from 6 categories in 5 languages (i.e., English, French, German, Italian, and Spanish). To extract the features from these articles, we follow the settings in [12, 42, 23] in which all the articles are represented by BoW with TF-IDF, followed by PCA for dimension reduction (with 60% energy preserved). The reduced dimensions with respect to different language categories English, French, German, Italian, and Spanish are 1,131, 1,230, 1,417, 1,041, and 807, respectively.

For simplicity, we fix the ratio $\delta = 0.5$ for all our ex-

Table 1. Classification results (%) with standard deviations for cross-feature object recognition.

S.T	SVM _t	DAMA	MMDT	SHFA	CDLS _{sup}	CDLS
<i>SURF to DeCAF₆</i>						
A, A	87.3±0.5	87.4±0.5	89.3±0.4	88.6±0.3	86.7±0.6	91.7±0.2
W, W	87.1±1.1	87.2±0.7	87.3±0.8	90.0±1.0	88.5±1.4	95.2±0.9
C, C	76.8±1.1	73.8±1.2	80.3±1.2	78.2±1.0	74.8±1.1	81.8±1.1
<i>DeCAF₆ to SURF</i>						
A, A	43.4±0.9	38.1±1.1	40.5±1.3	42.9±1.0	45.6±0.7	46.4±1.0
W, W	57.9±1.0	47.4±2.1	59.1±1.2	62.2±0.7	60.9±1.1	63.1±1.1
C, C	29.1±1.5	18.9±1.3	30.6±1.7	29.4±1.5	31.6±1.5	31.8±1.2

periments. We follow [21, 18, 19] and set $\lambda = \frac{1}{2}$ for regularizing \mathbf{A} . We also fix $m = 100$ as the reduced dimensionality for the derived feature subspace. Later, we will provide analyses on convergence and parameter sensitivity in Section 4.4 for verifying the robustness of our CDLS.

4.2. Evaluations

For evaluation, we first consider the performance of the baseline approach of SVM_t, which indicates the use of SVMs learned from only labeled target-domain data $\hat{\mathbf{X}}_L$. To compare our approach with state-of-the-art HDA methods, we consider HeMap [32], DAMA [38], MMDT [18, 19], and SHFA [23]. As noted earlier, since both SSKMDA [40] and SCP [41] require additional unlabeled data from the source domain for adaptation, we do not include their results for comparisons. Finally, we consider the supervised version of our CDLS (denoted as CDLS_{sup}, see Section 3.2.1), which simply derives feature transformation without learning cross-domain landmarks.

4.2.1 Object recognition across feature spaces

To perform cross-feature object recognition on **Office + Caltech-256** (C), two different types of features are available: *SURF* vs. *DeCAF₆*. If one of them is applied for describing the source-domain data, the other will be utilized for representing those in the target domain. For source-domain data with *SURF* features, we randomly choose 20 images per category as labeled data. As for the target domain with *DeCAF₆*, we randomly choose 3 images per object category as labeled data, and the rest images in each category as the unlabeled ones to be recognized. The same procedure is applied to the uses of *DeCAF₆* and *SURF* for describing source and target-domain data, respectively.

Table 1 lists the average classification results with 10 random trials. Since the number of images in DSLR (D) is much smaller than those in other domains (i.e., Amazon (A), Webcam (W), and Caltech-256 (C)), we only report the recognition performance on A, W, and C. Also, the performance of HeMap was much poorer than all other approaches (e.g., 11.7% on C for *DeCAF₆* to *SURF*), so we do not include its performance in the table.

Table 2. Classification results (%) with standard deviations for object recognition across feature representations and datasets.

S.T	SVM _t	DAMA	MMDT	SHFA	CDLS _{sup}	CDLS
<i>SURF to DeCAF₆</i>						
A, D	91.5±1.2	92.1±1.0	93.4±1.1	92.0±1.2	96.1±0.7	
W, D	90.9±1.1	91.2±0.9	91.5±0.8	92.4±0.9	91.0±1.1	95.1±0.8
C, D	91.0±1.3	93.1±1.2	93.8±1.0	91.9±1.3		94.9±1.5
<i>DeCAF₆ to SURF</i>						
A, D	53.2±1.5	53.5±1.3	56.1±1.0	54.3±1.3	58.4±0.8	
W, D	54.4±1.0	51.6±2.2	54.0±1.3	57.6±1.1	57.8±1.0	60.5±1.0
C, D	51.9±2.1	56.7±1.0	57.3±1.1	54.8±1.0		59.4±1.2

From Table 1, it can be seen that the methods of DAMA and CDLS_{sup} did not always achieve comparable performance as the baseline approach of SVM_t did. A possible explanation is that these two HDA approaches only related cross-domain heterogeneous data by deriving feature transformation for alignment purposes. For other HDA ones like MMDT, and SHFA, they additionally adapted the prediction models across-domain and thus produced improved results. Nevertheless, our CDLS consistently achieved the best performance across all categories. This supports the use of our HDA approach for cross-feature object recognition.

4.2.2 Object recognition across features and datasets

In this part of the experiments, we have source and target-domain data collected from distinct datasets and also described by different types of features. We adopt the same settings in Section 4.2.1 for data partition, while the only difference is that we choose DSLR as the target domain with others as source domains.

Table 2 lists the average classification results from 10 random trials. From this table, we see that existing HDA approaches generally produced slightly improved results than the use of SVM_t. Again, our proposed CDLS is observed to perform favorably against all HDA methods. It is worth repeating that, our improvement over CDLS_{sup} verifies the idea of exploiting information from unlabeled samples to select representative cross-domain landmarks for HDA.

4.2.3 Cross-lingual text categorization

To perform cross-lingual text categorization, we view articles in different languages as data in different domains. In our experiments, we have {English, French, German, or Italian} as the source domain, and randomly selected 100 articles from each as labeled data. On the other hand, we have Spanish as the target domain; for the target-domain data, we vary the number of labeled articles (from {5, 10, 15, 20}) per category, and randomly choose 500 articles from the remaining ones (per category) as the unlabeled data to be categorized.

Figure 3 illustrates the results over different numbers of labeled target-domain data. From this figure, it can be seen

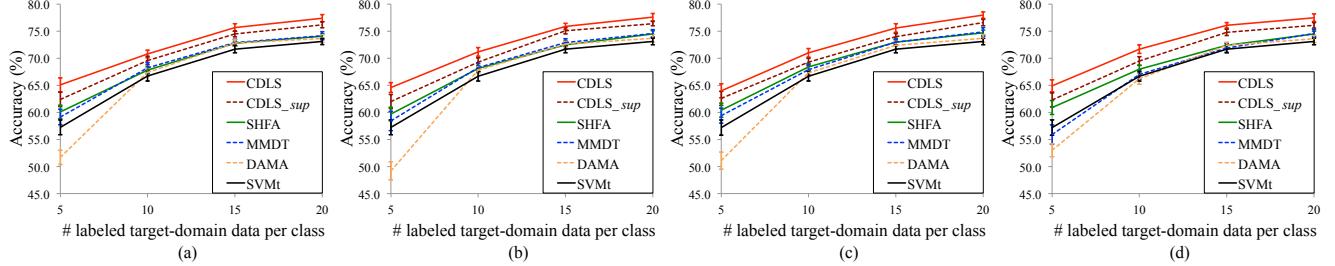


Figure 3. Classification results on cross-lingual text categorization using Multilingual Reuters Collection. Note that Spanish is viewed as the target domain, while the source domains are selected from (a) English, (b) French, (c) German, and (d) Italian, respectively.

Table 3. Classification results with standard deviations for cross-lingual text categorization with Spanish as the target domain.

Source Articles	# labeled target domain data / category = 10						# labeled target domain data / category = 20					
	SVMt	DAMA	MMDT	SHFA	CDLS _{sup}	CDLS	SVMt	DAMA	MMDT	SHFA	CDLS _{sup}	CDLS
English	67.4±0.7	68.2±1.0	67.8±0.7	69.6±0.8	70.8±0.7		73.6±0.7	74.2±0.8	74.1±0.4	76.2±0.6	77.4±0.7	
French	67.8±0.6	68.3±0.8	68.1±0.6	69.3±0.8	71.2±0.8		73.7±0.7	74.6±0.7	74.5±0.6	76.4±0.4	77.6±0.7	
German	66.7±0.9	67.3±0.6	67.9±1.0	68.4±0.7	69.3±0.7	71.0±0.8	73.1±0.6	73.7±0.7	74.9±0.8	74.7±0.5	76.6±0.6	78.0±0.6
Italian	66.2±1.0	67.0±0.9	68.0±0.7	69.5±0.8	71.7±0.8		73.6±0.7	74.6±0.7	74.5±0.5	76.1±0.5	77.5±0.7	

that the performances of all approaches were improved with the increase of the number of labeled target-domain data, while the difference between HDA approaches and SVM_t became smaller. We also list the categorization results in Table 3 (with 10 and 20 labeled target-domain instances per category). From the results presented in both Figure 3 and Table 3, the effectiveness of our CDLS for cross-lingual text categorization can be successfully verified.

4.3. Remarks on Cross-Domain Landmarks

Recall that the weights α and β observed for instances from each domain indicates its adaptation capability. That is, a labeled source-domain instance with a large α_i^c and an unlabeled target-domain instance with a large β_i^c would indicate that they are related to the labeled target-domain data of the same class c .

In Figure 4, we visualize the embedding for the adapted data and cross-domain landmarks using t-distributed stochastic neighbor embedding (t-SNE) [36]. Moreover, in Figure 5, we show example images and landmarks observed with different weights from two categories *laptop_computer* and *bike* on Caltech-256 → DSLR (with SURF to $DeCAF_6$). It can be seen that, the source-domain images with larger weights were not only more representative in terms of the category information, they were also visually more similar to the target-domain images of the class of interest. The same remarks can also be applied to the unlabeled target-domain data viewed as landmarks. It is also worth noting that, unlabeled target-domain images with smaller weights would be more likely to be mis-classified. For example, the image bounded by the red rectangle in Figure 5 was recognized as a similar yet different object category of *monitor*.

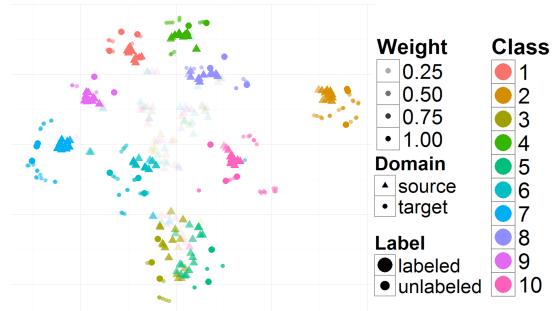


Figure 4. Landmarks (with different weights) observed from source and target-domain data in the recovered subspace.

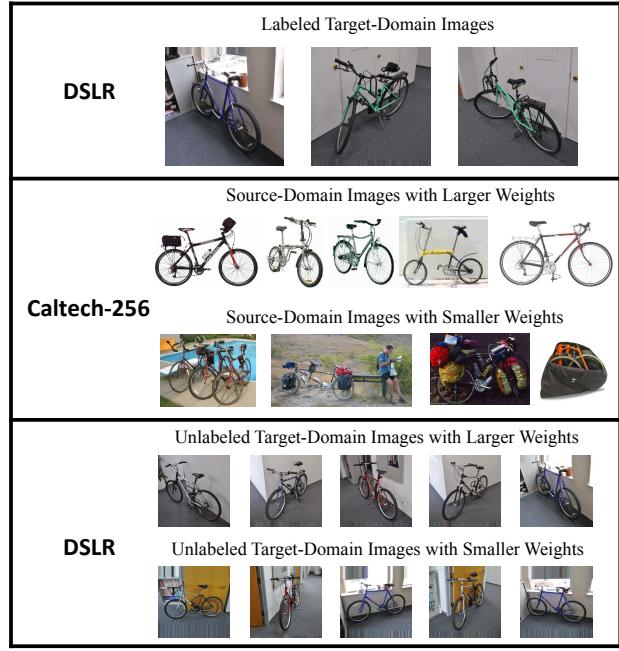
4.4. Convergence and Parameter Sensitivity

To further analyze the convergence property of our CDLS and assess its parameter sensitivity, we conduct additional experiments on cross-domain object recognition using C to D (with SURF to $DeCAF_6$). As discussed in Section 3.3, we alternate between the variables to be optimized when solving our CDLS.

We report the recognition performance with respect to the iteration number in Figure 6(a), in which we observe that the performance converged within 10 iterations. This observation is consistent to those of our other HDA experiments in this paper. As for the parameters of interest, we first discuss the selection of the dimension m of the reduced feature space. We vary the dimension number m and perform cross-domain recognition on the above cross-domain pairs. The results are presented in Figure 6(b). From this figure, we see that the performance increased when m was larger, while such performance improvements became marginal for m beyond 100. Thus, $m = 100$ would be a preferable choice for our experiments.



(a)



(b)

Figure 5. Example images and landmarks (with the observed weights) for Caltech-256 → DSLR (with SURF to $DeCAF_6$). The object categories are (a) *laptop_computer* and (b) *bike*. Note that the image bounded by the red rectangle was mis-classified as *monitor*.

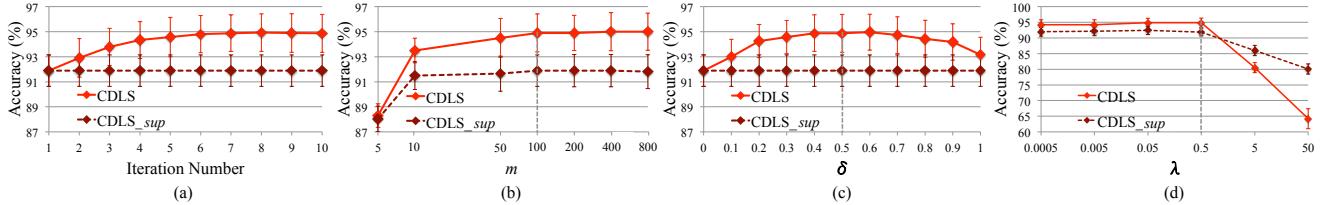


Figure 6. Analysis on convergence and parameter sensitivity using Caltech-256 with SURF to DSLR with $DeCAF_6$. The parameters of interests are (a) iteration number, (b) reduced dimensionality m , (c) ratio δ , and (d) regularization parameter λ .

Recall that the ratio δ in (4) controls the portion of cross-domain landmarks to be exploited for adaptation. Figure 6(c) presents the performance of CDLS with different δ values. It is worth repeating that, CDLS would be simplified as the supervised version CDLS_{sup} if $\delta = 0$, as illustrated by the flat dotted line in Figure 6(c). As expected, using all ($\delta = 1$) or none ($\delta = 0$) of the cross-domain data as landmarks would not be able to achieve satisfactory HDA performance. Therefore, the choice of $\delta = 0.5$ would be reasonable in our experiments.

Finally, Figure 6(d) presents the results of CDLS and CDLS_{sup} with varying λ values for regularizing \mathbf{A} . From this figure, it can be seen that fixing $\lambda = \frac{1}{2}$ as suggested by [21, 18, 19] would also be a reasonable choice.

5. Conclusion

We proposed *Cross-Domain Landmark Selection* (CDLS) for performing heterogeneous domain adaptation (HDA). In addition to the ability to associate heterogeneous data across domains in a semi-supervised setting, our CDLS is able to learn representative cross-domain landmarks for deriving a proper feature subspace for adaptation and classification purposes. Since the derived feature subspace matches cross-domain data distribution while eliminating the domain differences, we can simply project labeled cross-domain data to this domain-invariant subspace for recognizing the unlabeled target-domain instances. Finally, we conducted experiments on three classification tasks across different features, datasets, and modalities. Our CDLS was shown to perform favorably against state-of-the-art HDA approaches.

Acknowledgements

This work was supported in part by the Ministry of Science and Technology of Taiwan under Grants MOST103-2221-E-001-021-MY2, MOST104-2221-E-017-016, and MOST104-2119-M-002-039.

References

- [1] R. Aljundi, R. Emonet, D. Muselet, and M. Sebban. Landmarks-based kernelized subspace alignment for unsupervised domain adaptation. In *Computer Vision and Pattern Recognition (CVPR)*, 2015. [2](#)
- [2] M. Amini, N. Usunier, and C. Goutte. Learning from multiple partially observed views—an application to multilingual text categorization. In *Neural Information Processing Systems (NIPS)*, 2009. [5](#)
- [3] H. Bay, T.uytelaars, and L. Van Gool. Surf: Speeded up robust features. In *European Conference on Computer Vision (ECCV)*, 2006. [5](#)
- [4] C. M. Bishop. Pattern recognition and machine learning. Springer, 2006. [3](#)
- [5] C.-C. Chang and C.-J. Lin. Libsvm: A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 2011. [5](#)
- [6] R. Chattopadhyay, W. Fan, I. Davidson, S. Panchanathan, and J. Ye. Joint transfer and batch-mode active learning. In *International Conference on Machine Learning (ICML)*, 2013. [1](#)
- [7] M. Chen, K. Q. Weinberger, and J. Blitzer. Co-training for domain adaptation. In *Neural Information Processing Systems (NIPS)*, 2011. [1](#)
- [8] B. Chidlovskii, G. Csurka, and S. Gangwar. Assembling heterogeneous domain adaptation methods for image classification. In *Cross Language Evaluation Forum (CLEF)*, 2014. [1](#)
- [9] W.-S. Chu, F. De la Torre, and J. F. Cohn. Selective transfer machine for personalized facial action unit detection. In *Computer Vision and Pattern Recognition (CVPR)*, 2013. [2](#)
- [10] W. Dai, Y. Chen, G.-R. Xue, Q. Yang, and Y. Yu. Translated learning: Transfer learning across different feature spaces. In *Neural Information Processing Systems (NIPS)*, 2008. [1](#)
- [11] J. Donahue, Y. Jia, O. Vinyals, J. Hoffman, N. Zhang, E. Tzeng, and T. Darrell. Decaf: A deep convolutional activation feature for generic visual recognition. In *International Conference on Machine Learning (ICML)*, 2014. [5](#)
- [12] L. Duan, D. Xu, and I. Tsang. Learning with augmented features for heterogeneous domain adaptation. In *International Conference on Machine Learning (ICML)*, 2012. [1, 2, 5](#)
- [13] L. Duan, D. Xu, and I. W. Tsang. Domain adaptation from multiple sources: A domain-dependent regularization approach. In *IEEE Transactions on Neural Networks and Learning Systems (T-NNLS)*, 2012. [2](#)
- [14] N. Farajidavar, T. de Campos, and J. Kittler. Transductive transfer machine. In *Asian Conference of Computer Vision (ACCV)*, 2014. [5](#)
- [15] 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)*, 2013. [2](#)
- [16] A. Gretton, K. M. Borgwardt, M. Rasch, B. Schölkopf, and A. J. Smola. A kernel method for the two-sample-problem. In *Neural Information Processing Systems (NIPS)*, 2006. [3](#)
- [17] G. Griffin, A. Holub, and P. Perona. Caltech-256 object category dataset. 2007. [5](#)
- [18] J. Hoffman, E. Rodner, J. Donahue, T. Darrell, and K. Saenko. Efficient learning of domain-invariant image representations. In *International Conference on Learning Representation (ICLR)*, 2013. [1, 2, 3, 6, 8](#)
- [19] J. Hoffman, E. Rodner, J. Donahue, B. Kulis, and K. Saenko. Asymmetric and category invariant feature transformations for domain adaptation. In *International Journal of Computer Vision (IJCV)*, 2014. [1, 2, 3, 6, 8](#)
- [20] J. Huang, A. Gretton, K. M. Borgwardt, B. Schölkopf, and A. J. Smola. Correcting sample selection bias by unlabeled data. In *Neural Information Processing Systems (NIPS)*, 2006. [2](#)
- [21] B. Kulis, K. Saenko, and T. Darrell. What you saw is not what you get: Domain adaptation using asymmetric kernel transforms. In *Computer Vision and Pattern Recognition (CVPR)*, 2011. [1, 2, 3, 6, 8](#)
- [22] A. Kumar, A. Saha, and H. Daume. Co-regularization based semi-supervised domain adaptation. In *Neural Information Processing Systems (NIPS)*, 2010. [2](#)
- [23] W. Li, L. Duan, D. Xu, and I. W. Tsang. Learning with augmented features for supervised and semi-supervised heterogeneous domain adaptation. In *IEEE Transactions on Pattern Analysis and Machine Intelligence (T-PAMI)*, 2014. [1, 2, 5, 6](#)
- [24] M. Long, J. Wang, G. Ding, J. Sun, and P. S. Yu. Transfer feature learning with joint distribution adaptation. In *International Conference on Computer Vision (ICCV)*, 2013. [1, 3](#)
- [25] S. J. Pan, J. T. Kwok, Q. Yang, and J. J. Pan. Adaptive localization in a dynamic wifi environment through multi-view learning. In *Association for the Advancement of Artificial Intelligence (AAAI)*, 2007. [1](#)
- [26] S. J. Pan, I. W. Tsang, J. T. Kwok, and Q. Yang. Domain adaptation via transfer component analysis. In *IEEE Transactions on Neural Networks and Learning Systems (T-NNLS)*, 2011. [1, 2, 3](#)
- [27] S. J. Pan and Q. Yang. A survey on transfer learning. In *IEEE Transactions on Knowledge and Data Engineering (TKDE)*, 2010. [1](#)
- [28] V. M. Patel, R. Gopalan, R. Li, and R. Chellappa. Visual domain adaptation: A survey of recent advances. In *IEEE Signal Processing Magazine*, 2015. [1](#)
- [29] P. Prettenhofer and B. Stein. Cross-language text classification using structural correspondence learning. In *Association for Computational Linguistics (ACL)*, 2010. [1](#)
- [30] K. Saenko, B. Kulis, M. Fritz, and T. Darrell. Adapting visual category models to new domains. In *European Conference on Computer Vision (ECCV)*, 2010. [1, 5](#)

- [31] S. Shekhar, V. M. Patel, H. Nguyen, and R. Chellappa. Generalized domain-adaptive dictionaries. In *Computer Vision and Pattern Recognition (CVPR)*, 2013. 1
- [32] X. Shi, Q. Liu, W. Fan, P. S. Yu, and R. Zhu. Transfer learning on heterogenous feature spaces via spectral transformation. In *International Conference on Data Mining (ICDM)*, 2010. 1, 2, 6
- [33] V. Sindhwani, P. Niyogi, and M. Belkin. A co-regularization approach to semi-supervised learning with multiple views. In *International Conference on Machine Learning (ICML) workshop*, 2005. 1
- [34] E. Tzeng, J. Hoffman, T. Darrell, and K. Saenko. Simultaneous deep transfer across domains and tasks. In *International Conference in Computer Vision (ICCV)*, 2015. 1
- [35] N. Ueffing, M. Simard, S. Larkin, and J. H. Johnson. NRC’s PORTAGE system for WMT 2007. In *Association for Computational Linguistics (ACL) workshop*, 2007. 5
- [36] L. Van der Maaten and G. Hinton. Visualizing data using t-SNE. In *Journal of Machine Learning Research (JMLR)*, 2008. 7
- [37] C. Wang and S. Mahadevan. Manifold alignment without correspondence. In *International Joint Conference on Artificial Intelligence (IJCAI)*, 2009. 3
- [38] C. Wang and S. Mahadevan. Heterogeneous domain adaptation using manifold alignment. In *International Joint Conference on Artificial Intelligence (IJCAI)*, 2011. 1, 2, 6
- [39] X. Wu, H. Wang, C. Liu, and Y. Jia. Cross-view action recognition over heterogeneous feature spaces. In *International Conference in Computer Vision (ICCV)*, 2013. 1, 2
- [40] M. Xiao and Y. Guo. Feature space independent semi-supervised domain adaptation via kernel matching. In *IEEE Transactions on Pattern Analysis and Machine Intelligence (T-PAMI)*, 2015. 1, 2, 3, 6
- [41] M. Xiao and Y. Guo. Semi-supervised subspace co-projection for multi-class heterogeneous domain adaptation. In *European Conference on Machine Learning (ECML)*, 2015. 1, 2, 3, 6
- [42] J. T. Zhou, I. W. Tsang, S. J. Pan, and M. Tan. Heterogeneous domain adaptation for multiple classes. In *International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2014. 1, 2, 3, 5
- [43] Y. Zhu, Y. Chen, Z. Lu, S. J. Pan, G.-R. Xue, Y. Yu, and Q. Yang. Heterogeneous transfer learning for image classification. In *Association for the Advancement of Artificial Intelligence (AAAI)*, 2011. 1