Joint cross-domain classification and subspace learning for unsupervised adaptation

Basura Fernando [a,1], Tatiana Tommasi [b,1,*], Tinne Tuytelaars [c]

[a] College of Engineering and Computer Science, The Australian National University, Canberra, Australia
[b] Department of Computer Science, University of North Carolina, Chapel Hill, USA
[c] KU Leuven, ESAT-PSI - iMinds, Kasteelpark Arenberg 10, B-3001 Leuven, Belgium

ARTICLE INFO

ABSTRACT

Domain adaptation aims at adapting the knowledge acquired on a source domain to a new different but related target domain. Several approaches have been proposed for classification tasks in the unsupervised scenario, where no labeled target data are available. Most of the attention has been dedicated to searching a new domain-invariant representation, leaving the definition of the prediction function to a second stage. Here we propose to learn both *jointly*. Specifically we learn the source subspace that best matches the target subspace while at the same time minimizing a regularized misclassification loss. We provide an alternating optimization technique based on stochastic sub-gradient descent to solve the learning problem and we demonstrate its performance on several domain adaptation tasks.

1. Introduction

In real world applications, having a probability distribution mismatch between the training and the test data is more often the rule than an exception. Think about part of speech tagging across different text corpora [5], localization over time with wifi signal distributions that get easily outdated [42], or biological models to be used across different subjects [38]. Computer vision methods are also particularly challenged in this respect: real world conditions may alter the image statistics in many complex ways (lighting, pose, background, motion blur etc.), to not even mention the difference in quality of the acquisition device (e.g. resolution), or the high number of possible artificial modifications obtained by post-processing (e.g. filtering). Due to this large variability, any learning algorithm trained on a *source* set regardless of the final *target* data will most likely produce poor, unsatisfactory results.

Domain adaptation techniques propose to overcome these issues and make use of information coming from both source and target domains during the learning process. In the *unsupervised* case, where no labeled samples are provided for the target, the most extensively studied paradigm consists in assuming the existence of a domain-invariant feature space and searching for it. In general all the techniques based on this idea focus on transforming the representation

of the source and target samples to maximize some notion of similarity between them [12,14,15]. However in this way the classification task is left aside and the prediction model is learned only in a second stage. As thoroughly discussed in [2,29], the choice of the feature representation able to reduce the domain divergence is indeed a crucial factor for the adaptation. Nevertheless it is not the only one. If several representations induce similar marginal distributions for the two domains, would a classifier perform equally well in all of them? Is it enough to encode the labeling information in the used feature space or is it better to learn a cross-domain classification model together with the optimal domain-invariant representation? Here we answer these questions by focusing on unsupervised domain adaptation subspace solutions. We present an algorithm that learns jointly both a low dimensional representation and a reliable classifier by optimizing a trade-off between the source-target similarity and the source training error.

2. Related work

For classification tasks, the goal of domain adaptation is to learn a function from the source domain that predicts the class label of a novel test sample from the target domain [33]. In the literature there are two main scenarios depending on the availability of data annotations: the semi-supervised and the unsupervised setting.

In the *semi-supervised* setting a few labeled samples are provided for the target domain besides a large amount of annotated source data. Existing solutions can be divided into *classifier-based* and *representation-based* methods. The former modify the original

* Corresponding author. Tel.: +91 95906030.
E-mail address: ttommasi@cs.unc.edu, tommasi.t@gmail.com (T. Tommasi).
[1] Work done while at KU Leuven.

formulation of Support Vector Machines (SVM) [10,41] and other statistical classifiers [6]: they adapt a pre-trained model to the target, or learn the source and target classifiers simultaneously. The latter exploit the correspondence between source and target labeled data to impose constraints over the samples through metric learning [25,34], or consider feature augmentation strategies [23] and manifold alignment [40]. Some approaches have also tackled the cases with more than two available domains [9,19] and the unlabeled part of the target has been used for co-regularization [26]. Recently, two methods proposed to combine classifier-based and representation-based solutions. [20] introduced an approach to learn jointly a cross-domain classifier and a transformation that maps the target points into the source domain. Several kernel maps are used to encode the representation in [8], which proposed a domain transfer multiple kernel learning algorithm.

In the more challenging *unsupervised* setting, all the available target samples are unlabeled. Many unsupervised domain adaptive approaches resort to estimating the data distributions and minimizing a distance measure between them, while re-weighting/selecting the samples [13,38]. The Maximum Mean Discrepancy (MMD) [16] maps two sets of data to a reproducing Kernel Hilbert Space and it has been largely used as distance measure between two domain distributions. Although endowed with nice properties, the choice of the kernel and kernel parameters are critical and, if non-optimal, can lead to a very poor estimate of the distribution distance [17]. Dictionary learning methods have also been used with the goal of defining new representations that overcome the domain shift [30,35]. A reconstruction approach was proposed in [24]: the source samples are mapped into an intermediate space where each of them can be represented as a linear combination of the target domain samples.

Another promising direction for unsupervised domain adaptation is that of *subspace modeling*. This is based on the idea that source and target share a latent subspace where the domain shift is removed or reduced. As for dictionary learning, the approaches presented in this framework are mostly linear, but can be easily extended to non-linear spaces through explicit feature mappings [39]. In [4] Canonical Correlation Analysis (CCA) has been applied to find a coupled domain-invariant subspace. Principal Component Analysis (PCA) and other eigenvalue methods are also widely used for subspace generation. For instance, Transfer Component Analysis (TCA, [32]) is a dimensionality reduction approach that searches a latent space where the variance of the data is preserved as much as possible and the distance between the distributions is reduced. Transfer Subspace Learning (TSL, [37]) couples PCA and other subspace learning methods with a Bregman divergence-based regularization which measures the distance between the distribution in the projected space. Alternatively, the algorithm introduced in [1] uses MMD in the subspace to search for a domain invariant projection matrix. Other methods exploited multiple intermediate subspaces to link the source and the target data. This idea was introduced in [15] where the path across the domains is defined as a geodesic curve over a Grassmann manifold. This strategy has been further extended in [14] where all the intermediate subspaces are integrated to define a cross-domain similarity measure. Despite the intuitive characterization of the problem, it is not clear why all the subspaces along this path should yield meaningful representations. Recently the Subspace Alignment (SA) method [12] demonstrated that it is possible to map directly the source to the target subspace without necessarily passing through intermediate steps.

Overall, the main focus of the unsupervised methods proposed in the literature is on the domain invariance of the final data representation and less attention has been dedicated to its discriminative power. First attempts in this direction have been done in [14] by substituting the use of PCA over the source subspace with Partial Least Squares (PLS), and in [32] where SSTCA chooses the representation by maximizing its dependence on the data labels. Our work fits in this context. We aim at extending the integration of classifier-based with representation-based solutions in the unsupervised

setting where no access to the target labels is available, not even for hyperparameter cross validation. Differently from all the described unsupervised approaches we go beyond searching only a domain invariant feature space and we want to optimize also a cross-domain classification model. We propose an algorithm that combines effectively subspace and max-margin learning and exploits the source discriminative information better than just encoding it in the representation. Our approach does not need an estimate of the source and target data distributions and relies on a simple measure of domain shift. Finally, in previous work the performance of the adaptive methods have been often evaluated by tuning the model parameters on the target data [12,24] or by fixing them to default values [21]. Here we choose a more fair setup for unsupervised domain adaptation and we show that our approach outperforms different existing subspace adaptive methods by exploiting exclusively the source annotations. We name our algorithm *Joint cross-domain Classification and Subspace Learning* (JCSL).

In the following sections we define the notation that will be used in the rest of the paper (Section 3) and we briefly review the theory of learning from different domains together with the subspace domain shift measure used in [12] from which we took inspiration. We then introduce our approach (Section 4) followed by an extensive experimental analysis that shows its effectiveness on several domain adaptation tasks (Section 5). We conclude with a final discussion and sketching possible directions for future research (Section 6).

3. Problem setup and background

Let us consider a classification problem where the data instances are in the form $(\mathbf{x}_i, y_i)$. Here $\mathbf{x}_i \in \mathbb{R}^D$ is the feature vector for the i-th sample and $y_i \in \{1, \ldots, K\}$ is the corresponding label. We assume that n_s labeled training samples are drawn from a source distribution $\mathcal{D}_s = P(\mathbf{x}^s, y^s)$, while a set of n_t unlabeled test samples come from a different target distribution $\mathcal{D}_t = P(\mathbf{x}^t, y^t)$, such that it holds $\mathcal{D}_s \neq \mathcal{D}_t$. In particular, the source and the target distributions satisfy the covariate shift property [36] if they have the same labeling function with $P(y^s|\mathbf{x}^s) = P(y^t|\mathbf{x}^t)$, while the marginal distributions differ $P(\mathbf{x}^s) \neq P(\mathbf{x}^t)$. We operate under this hypothesis.

A bound on the target domain error. Theoretical studies on domain adaptation have established the conditions under which a classifier trained on the source data can be expected to perform well on the target data. The following generalization bound on the target error ϵ_t has been demonstrated in [2]:

$$\epsilon_t(h) \leq \epsilon_s(h) + d_{\mathcal{H}}(\mathcal{D}_s, \mathcal{D}_t) + \lambda. \tag{1}$$

Here h indicates the predictor function, while $\mathcal{H}$ is the hypothesis class from which the predictor has been chosen. In words, the bound states that a low target error can be guaranteed if the source error $\epsilon_s(h)$, a measure of the domain distribution divergence $d_{\mathcal{H}}(\mathcal{D}_s, \mathcal{D}_t)$, and the error λ of the ideal joint hypothesis on the two domains are small. The joint error can be written as $\lambda = \epsilon_t(h^*) + \epsilon_s(h^*)$ where $h^* = argmin_{h \in \mathcal{H}}(\epsilon_t(h) + \epsilon_s(h))$. The value of λ is supposed to be low under the the covariate shift assumption.

A subspace measure of domain shift. The low-dimensional intrinsic structure of the source and target domains can be specified by their corresponding orthonormal basis sets, indicated respectively as $S \in \mathbb{R}^{D \times d}$ and $T \in \mathbb{R}^{D \times d}$. These are two full rank matrices, and d is the subspace dimensionality. In [12], a transformation matrix M is introduced to modify the source subspace. The domain shift of the transformed source basis with respect to the target is simply measured by the following function:

$$F(M) = ||SM - T||_F^2, \tag{2}$$

where $|| \cdot ||_F$ is the Frobenius norm. The *Subspace Alignment* (SA) method proposed to minimize this measure, obtaining the optimal

transformation matrix in closed form: $M = S^\top T \in \mathbb{R}^{d \times d}$. The matrix $U = SM = SS^\top T \in \mathbb{R}^{D \times d}$ is finally used to represent the source data. The original domain basis sets can be obtained through different strategies, both unsupervised (PCA) and supervised (PLS, LDA), as extensively studied in [11].

SA has shown promising results for visual cross-domain classification tasks outperforming other subspace adaptive methods. However, on par with its competitors [14,15], it keeps the domain adaptive process (learning M) and the classification process (e.g. learning an SVM model) separated, focusing only on the distribution divergence term $d_{\mathcal{H}}(\mathcal{D}_s, \mathcal{D}_t)$ of the bound in (1).

4. Proposed approach

With the aim of minimizing both the domain divergence and the source error in (1), we propose an algorithm that learns a domain-invariant representation and an optimal cross-domain classification model. For the representation we concentrate on subspace methods and we take inspiration from the SA approach. For the classification we rely on a standard max-margin formulation. The details of our *Joint cross-domain Classification and Subspace Learning* (JCSL) algorithm are described below.

Given a fixed target subspace basis $T \in \mathbb{R}^{D \times d}$ we minimize the following regularized risk functional

$$G(V, \mathbf{w}) = ||\mathbf{w}||_2^2 + \alpha ||V - T||_F^2 + \beta \sum_i^{n_s} \mathcal{L}(\mathbf{x_i^s}, y_i^s, \mathbf{w}, V). \quad (3)$$

Here the regularization terms aim at optimizing separately the linear source classification model $\mathbf{w} \in \mathbb{R}^d$, and the source representation matrix $V \in \mathbb{R}^{D \times d}$, while the loss function $\mathcal{L}$ depends on their combination. For our analysis we choose the hinge loss: $\mathcal{L}(\mathbf{x_i^s}, y_i^s, \mathbf{w}, V) = \max\{0, 1 - \mathbf{x_i^s}^\top V \mathbf{w} y_i^s\}$, but other loss functions can be used for different cross-domain applications. The parameters α and β allows to define a trade-off between the importance of the terms in the objective function. In particular a high α value pushes V towards T giving more importance to the distribution divergence term, while a high β value focuses the attention on the training error term to improve the classification performance in the new space.

The matrix V has a role analogous to that of U in SA, however in our case it is not necessary to specify a priori the source subspace S which is now optimized together with the alignment transformation matrix M in a single step. Note that, if the source and target data can be considered as belonging to the same domain (no domain shift), our method will automatically provide $V = T$ boiling down to standard learning in the shared subspace. We follow previous literature and propose the use of PCA to define the target subspace T [12,14]. Besides having demonstrated good results empirically, the theoretical meaning of this choice can be identified by writing the mutual information between the target and the source as

$$MI(source; target) = H(target) - KL(source||target). \quad (4)$$

Projecting the target data to the subspace T maximizes the entropy $H(target)$, while our objective function minimizes the domain shift, which is related to the Kullback-Leibler divergence $KL(\cdot||\cdot)$. Hence, we expect to increase the mutual information between source and target.

Minimizing (3) jointly over $(V, \mathbf{w})$ is a non-convex problem and finding a global optimum is generally intractable. However we can apply alternated minimization for V and $\mathbf{w}$ resulting in two interconnected convex problems that can be efficiently solved by stochastic subgradient descent. For this procedure we need the partial derivatives of (3) that can be easily calculated as:

$$\frac{\partial G(V, \mathbf{w})}{\partial V} = 2\alpha(V - T) - \beta \Sigma_{i=1}^{n_s} \Gamma_i$$

$$\frac{\partial G(V, \mathbf{w})}{\partial \mathbf{w}} = 2\mathbf{w} - \beta \Sigma_{i=1}^{n_s} \Theta_i \quad (5)$$

where Γ and Θ are the derivatives of $\mathcal{L}(\mathbf{x_i^s}, y_i^s, \mathbf{w}, V)$ with respect to V and $\mathbf{w}$. When using the hinge loss we get

$$\Gamma_i = \begin{cases} \mathbf{x_i^s}^\top \mathbf{w} y_i^s \\ 0 \end{cases} \quad \Theta_i = \begin{cases} \mathbf{x_i^s}^\top V y_i^s & \text{if } (\mathbf{x_i^s}^\top V \mathbf{w} y_i^s) < 1 \\ 0 & \text{otherwise}. \end{cases}$$

The iterative subgradient descent procedure terminates when the algorithm converges, showing a negligible change of either V or $\mathbf{w}$ between two consecutive iterations. The formulation holds for a binary classifier but can easily be used in its one-vs-all multiclass extension that highly benefits from the choice of the stochastic variant of the optimization process.

At test time, we indicate the classification score of class y for the target sample $\mathbf{x_i^t}$ as $s(\mathbf{x_i^t}, \mathbf{w}_y) = \mathbf{x_i^t}^\top T \mathbf{w}_y$. The multiclass final prediction is then obtained by maximizing over the scores: $y_i^* = argmax_y(s(\mathbf{x_i^t}, \mathbf{w}_y))$. Note that the source representation matrix V is not involved at this stage, and the target subspace basis T appears instead. Differently from the pre-existing unsupervised domain adaptation methods that encode the discriminative information in the representation, JCSL learns directly a domain invariant classification model able to generalize from source to target. The JCSL learning strategy is summarized in Algorithm 1.

Algorithm 1 JCSL.

Input: step size η and batch size γ for stochastic sub-gradient descent
Output: $V^*, \mathbf{w}^*$
1: Initialize $V \leftarrow S, \mathbf{w} \leftarrow 0, k \leftarrow 0$
2: **while** not converged **do**
3:     $k \leftarrow k + 1$
4:     calculate the partial derivatives:

$$\frac{\partial G(V, \mathbf{w})}{\partial V} = 2\alpha(V - T) - \beta \Sigma_{i=1}^{\gamma} \Gamma_i$$

with $\Gamma_i = \begin{cases} \mathbf{x_i^s}^\top \mathbf{w} y_i^s & \text{if } (\mathbf{x_i^s}^\top V \mathbf{w} y_i^s) < 1 \\ 0 & \text{otherwise} \end{cases}$

$$\frac{\partial G(V, \mathbf{w})}{\partial \mathbf{w}} = 2\mathbf{w} - \beta \Sigma_{i=1}^{\gamma} \Theta_i$$

with $\Theta_i = \begin{cases} \mathbf{x_i^s}^\top V y_i^s & \text{if } (\mathbf{x_i^s}^\top V \mathbf{w} y_i^s) < 1 \\ 0 & \text{otherwise} \end{cases}$

5:     Fix V, identify the optimal $\mathbf{w}$:

$$\mathbf{w}_k \leftarrow \mathbf{w}_{k-1} - \eta \left(\frac{\partial F(V, \mathbf{w})}{\partial \mathbf{w}} \right)_{\mathbf{w}_{k-1}}$$

6:     Fix $\mathbf{w}$, identify the optimal V:

$$V_k \leftarrow V_{k-1} - \eta \left(\frac{\partial F(V, \mathbf{w})}{\partial V} \right)_{V_{k-1}}$$

7: **end while**

---

5. Experiments

We validate our approach over several domain adaptation tasks. In the following we first describe our experimental setting (Section 5.1) and then we report on the obtained results (Sections 5.2–5.4). Moreover we present a detailed analysis on the role of the learning parameters and on the domain-shift reduction effect of JCSL (Section 5.2).

5.1. Datasets, baselines and implementation details

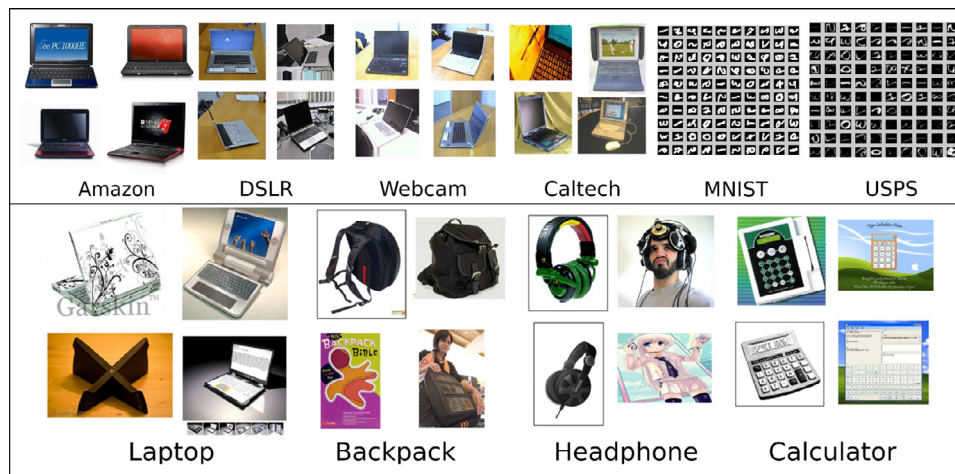We choose three image datasets (see Fig. 1) and a wifi signal dataset.

Fig. 1. Top line: examples from Office+Caltech dataset and the MNIST+USPS dataset. Bottom line: weakly labeled images from Bing dataset.

Office + Caltech [14]. This dataset was created by combining the Office dataset [34] with Caltech256 [18] and it contains images of 10 object classes over four domains: Amazon, Dslr, Webcam and Caltech. Amazon consists of images from online merchants' catalogues, while Dslr and Webcam domains are composed by respectively high and low resolution images. Finally, Caltech corresponds to a subset of the original Caltech256. We use the features provided by Gong et al. [14]: SURF descriptors quantized into histograms of 800 bag-of-visual words and standardized by z-score normalization. All the 12 possible source-target domain pairs are considered. We use the data splits provided by Hoffman et al. [20].

MNIST [27] **+ USPS** [22]. This dataset combines two existing image collections of digits presenting different gray scale data distributions. Specifically they share 10 classes of digits. We randomly selected 1800 images from USPS and 2000 images from MNIST. By following [28] we uniformly re-scale all images to size 16×16 and we use the L2-normalized gray-scale pixel values as feature vectors. Both domains are alternatively used as source and target.

Bing + Caltech [3]. In this dataset, weakly annotated images from the Bing search engine define the source domain while images of Caltech256 are used as target. We run experiments varying the number of categories (5, 10, 15, 20, 25 and 30) and the number of source examples per category (5 and 10) using the same train/test split adopted in [3]. As typically done for this dataset, Classemes features are used as image representation [3].

Wifi [42]. This dataset was used in the 2007 IEEE ICDM contest for domain adaptation. The goal is to estimate the location of mobile devices based on the received signal strength (RSS) values from different access points. The domains correspond to two different time periods during which the collected RSS values present different distributions. The dataset contains 621 labeled examples collected during time period A (source) and 3128 unlabeled examples collected during time period B (target). The location recognition performance is generally evaluated by measuring the average error distance between the predicted and the correct space position of the mobile devices. We slightly modify the task to define a classification rather than a regression problem. We consider 247 locations and we evaluate the classification performance between the sets A and B with and without domain adaptation. We repeat the experiments both testing over all the target data and considering 10 random target splits, each with 400 random samples.

We benchmark JCSL[2] against the following subspace-based domain adaptation methods:

TCA, SSTCA: Transfer Component Analysis and its semi-supervised extension [32]. We implemented TCA and SSTCA by following the original paper description. For SSTCA we turned off the locality preserving option to have a fair comparison with all the other considered methods, none of which exploits local geometry[3].

TSL: Transfer Subspace Learning [37]. We used the code made publicly available by the authors[4] which implements TLS by adding a Bregman-divergence based regularization to the Fisher's Linear Discriminant Analysis (FLDA).

GFK $_{\text{(LDA−PCA)}}$, **SA** $_{\text{(LDA−PCA)}}$: for both the Geodesic Flow Kernel [14] and the Subspace Alignment [12] methods we slightly modified the original implementation provided by the authors[5] to integrate the available discriminative information in the source domain. As preliminary evaluation we compared the results of GFK and SA when the basis of the source subspace were obtained with PLS and LDA. Although performing similarly on average, PLS showed less stability than LDA with large changes in the outcome for small variations of the subspace dimensionality d. This can be explained by considering the difficulty of finding the best d that jointly maximizes the source data/label coherence and minimizes the source/target shift. Thus, for our experiments we rely on the more stable LDA for the source which fixes d equal $K − 1$. On the other hand, the target subspace is always obtained by applying PCA and selecting the first $K − 1$ eigenvectors.

As further baselines we also consider the source classifier learned with no adaptation (**NA**) in the original feature space and in the target subspace. The last one is obtained by applying PCA on the target domain and using the eigenvectors as basis to represent both the source and the target data (**PCA**$_T$).

For all the methods the final classifier is a linear SVM with the C parameter tuned by two-fold cross-validation on the source over the range {0.001, 0.01, 0.1, 1.0, 10}. Our JCSL has three main parameters (α, β, d) that are also chosen by two-fold cross validation on the source. We remark that the target data are not annotated, thus tuning the parameters on the source is the only feasible option. We searched for α, β in the same range indicated before for C. The parameter d was tuned in {10, 20, . . . , 100} both for JCSL and for the baselines **PCA**$_T$, **TSL**, **TCA** and **SSTCA**.

We implemented the stochastic sub-gradient descent using a step size of $\eta = 0.1$ and a batch size of $\gamma = 10$. The alternating optimization converges for less than 100 iterations and we can obtain the results for any of the source-target domain pairs of the Office+Caltech

---

[2] We implemented our algorithm in MATLAB. The code is submitted with the paper as supplementary material.

[3] Following the original paper notation we fixed $\mu = 0.1$, $\lambda = 0$, $\gamma = 0.5$.

[4] http://www.cs.utexas.edu/~ssi/TrFLDA.tar.gz

[5] http://www-scf.usc.edu/~boqinggo/domain_adaptation/GFK_v1.zip,
http://homes.esat.kuleuven.be/~bfernand/DA_SA/downloadit.php?fn=DA_SA.zip

Table 1
Recognition rate (%) results over the Office+Caltech and MNIST+USPS datasets.

| DA problem	NA	PCA$_T$	SA$_{(LDA-PCA)}$	GFK$_{(LDA-PCA)}$	TCA	SSTCA	TSL	JCSL
A → C	38.1 ± 2.6	41.1 ± 1.7	43.4 ± 3.2	43.2 ± 3.7	**43.5 ± 3.2**	38.8 ± 2.4	40.4 ± 0.9	42.6 ± 0.9
A → D	32.9 ± 2.8	37.5 ± 1.4	**44.7 ± 2.6**	43.7 ± 2.8	38.8 ± 2.1	34.1 ± 6.9	40.8 ± 1.7	42.5 ± 3.2
A → W	36.8 ± 2.9	39.1 ± 2.8	40.3 ± 2.9	41.3 ± 2.0	41.0 ± 1.0	34.1 ± 3.8	41.1 ± 2.3	**47.6 ± 2.1**
C → A	39.5 ± 1.0	40.6 ± 2.8	39.3 ± 1.8	39.9 ± 1.5	42.5 ± 1.2	39.1 ± 3.5	43.0 ± 4.2	**44.3 ± 1.2**
C → D	38.8 ± 2.4	40.9 ± 1.3	44.0 ± 2.0	42.0 ± 2.5	42.1 ± 2.5	38.3 ± 4.1	40.4 ± 3.4	**46.5 ± 1.5**
C → W	37.8 ± 1.4	35.9 ± 3.2	37.3 ± 3.3	41.8 ± 3.8	39.4 ± 2.5	31.6 ± 4.9	40.9 ± 2.4	**46.5 ± 2.0**
D → A	24.4 ± 1.7	30.6 ± 2.7	35.7 ± 2.3	31.0 ± 2.7	30.4 ± 2.1	38.0 ± 2.6	39.6 ± 1.2	**41.3 ± 0.9**
D → C	30.5 ± 2.1	37.9 ± 1.3	**41.5 ± 1.4**	40.9 ± 2.8	36.7 ± 2.6	32.9 ± 1.8	33.4 ± 2.1	35.1 ± 0.9
D → W	60.9 ± 3.1	67.9 ± 2.8	58.6 ± 2.1	60.5 ± 3.8	64.5 ± 3.2	**76.2 ± 3.0**	73.7 ± 1.5	74.2 ± 3.6
W → A	29.7 ± 1.7	34.1 ± 1.7	34.7 ± 0.9	33.1 ± 1.2	34.6 ± 1.4	35.1 ± 2.6	38.0 ± 1.6	**43.1 ± 1.0**
W → C	34.1 ± 1.6	38.1 ± 1.3	36.7 ± 1.2	37.7 ± 1.8	**39.6 ± 2.2**	29.7 ± 2.5	30.4 ± 1.3	36.1 ± 2.0
W → D	71.1 ± 2.6	74.6 ± 0.9	69.5 ± 2.4	75.3 ± 2.6	**77.3 ± 2.7**	69.9 ± 3.4	66.9 ± 1.6	66.2 ± 2.9
AVG.	39.6	43.2	43.8	44.2	44.2	41.4	44.0	**47.2**
MNIST → USPS	45.4	45.1	**48.6**	34.6	40.8	40.6	43.5	46.7
USPS → MNIST	33.3	33.4	22.2	22.6	27.4	22.2	34.1	**35.5**
AVG.	39.4	39.2	35.4	28.6	34.1	31.4	38.8	**41.1**
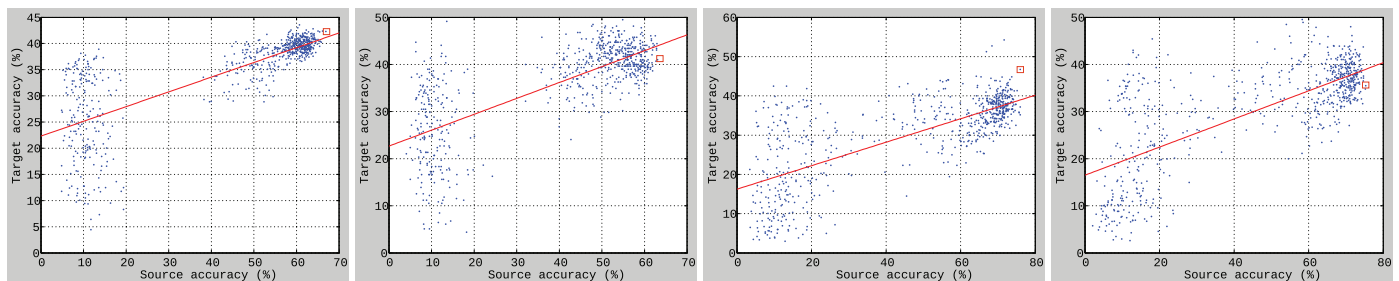
Fig. 2. Target accuracy vs source accuracy over domain adaptation problem (Amazon → Caltech), (Amazon → Webcam), (MNIST → USPS) and (USPS → MNIST) obtained by using JCSL and changing the parameters α, β and d. In all the cases the top right point cluster shows the high correlation between the source and target accuracy. By comparing the x- and y- axis values of this cluster it is also evident the source-to-target performance drop with respect to the source-to-source result in each experiment. The red square indicates the result selected by our method for the considered split. The red line is obtained by least-square fitting and makes it evident the trend in the results shared by all the source-target pairs.

(excluding feature extraction) in 2 min using a modern desktop computer (2.8 GHz cpu, 4Gb of ram, 1 core). With respect to the considered competing methods, the training phase of JCSL is slower (e.g. 60 times slower with respect to SA and GFK), but we remark that JCSL provides an optimized cross-domain classifier besides reducing the data distribution shift. The test phase runtime is comparable for all the considered approaches. In practical applications domain adaptation models are usually learned offline, thus the training time is a minor issue.

5.2. Results - Office+Caltech and MNIST+USPS

The obtained results over the Office+Caltech and MNIST+USPS datasets are presented in Table 1. Overall JCSL outperforms the considered baselines in 7 source-target pairs out of 14 and shows the best average results over the two datasets. Thus, we can state that, minimizing a trade-off between source-target similarity and the source classification error pays off compared to only reducing the cross-domain representation divergence. Still SA shows an advantage with respect to JCSL in a few of the considered cases most probably because it can exploit the discriminative LDA subspace. With respect to JCSL, TCA and SSTCA seem to work particularly well when the domain shift is small (e.g. Amazon → Caltech, Dslr → Webcam). Interestingly JCSL is the only method that consistently outperforms NA over MNIST+USPS.

Parameter analysis. To better understand the performance of JCSL we analyze how the target accuracy varies with respect to the source accuracy while changing the learning parameters α, β and d. The plots in Fig. 2 consider four domain adaptation problems, namely (Amazon → Caltech), (Amazon → Webcam), (MNIST → USPS) and (USPS → MNIST)[6]. All of them present two main clusters. On the left, when the source accuracy is low, the target accuracy is uniformly distributed. This behavior mostly appears when β is very small and α has a high value: this indicates that minimizing only $||V - T||_F^2$ does not guarantee stable results on the target task. On the other hand, in the second cluster the source accuracy is highly correlated with the target accuracy. On average, for the points in this region, both the domain divergence term and the misclassification loss obtain low values. The final JCSL result with the optimal (V^*, $\mathbf{w}^*$) appears always in this area and the dimensionality of the subspace d seems to have only a moderate influence on the final results. The red line reported on the plots is obtained by least-square fitting over the source and target accuracies and presents an analogous trend for all the considered source-target pairs. This is an indication that when domains are adaptable (negligible λ in (1)) our method is able to find a good source representation as well as a classifier that generalizes to the target domain.

Measuring the domain shift. For the same domain pairs considered above we also evaluate empirically the $\mathcal{H}\Delta\mathcal{H}$ divergence measure defined in [2]. This is obtained by learning a linear SVM that discriminates between the source and target instances, respectively pseudo-labeled with +1 and −1. We separated each domain into two halves and use them for training and test when learning a linear SVM model. A high final accuracy indicates high domain divergence. We perform this analysis by comparing the domain shift before and after the application of SA and JCSL, according to their standard settings. SA presents a single step and learns one subspace representation U. JCSL exploits a one-vs-all procedure learning as many V_y as the number

---

[6] Analogous results are obtained for all the remaining source-target pairs.
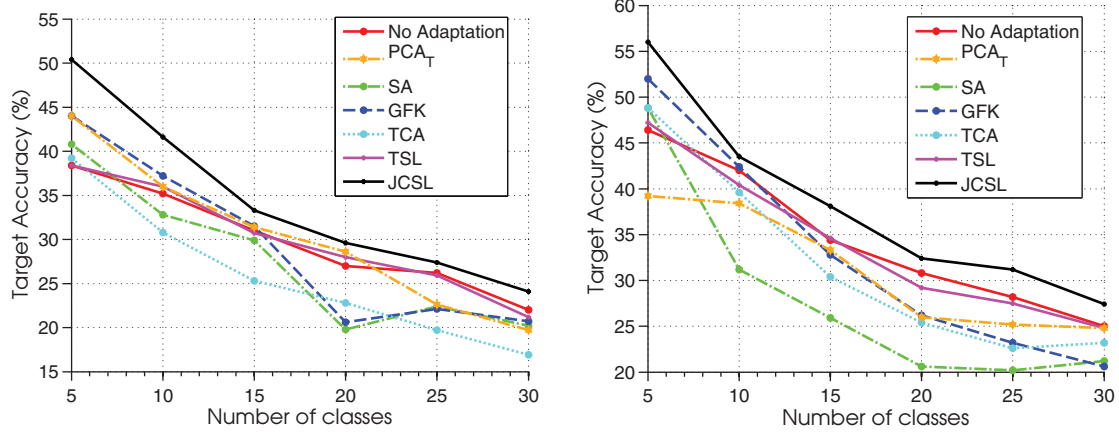
Fig. 3. Experimental results on Bing+Caltech obtained when using 5 (left) and 10 (right) samples per class in the source. SSTCA has shown similar or worse results than TCA, so we did not include it in this evaluation to avoid further clutter in the plot.

Table 2
$\mathcal{H}\Delta\mathcal{H}$ analysis. Lower values indicate lower cross-domain distribution discrepancy.

Space	$A \rightarrow C$	$A \rightarrow W$	MNIST $\rightarrow$ USPS	USPS $\rightarrow$ MNIST
Original features	74.82	90.18	100.00	100.00
$SA_{(LDA-PCA)}$	65.96	56.56	55.78	55.74
JCSL	65.76	54.97	57.03	53.28

Table 3
Classification accuracy obtained over wifi localization dataset [42]. The *full*-row contains the results over the whole target set. In the *split*-row we present the results obtained over 10 splits of the target, each containing 400 randomly extracted samples.

	NA	$SA_{(LDA-PCA)}$	$GFK_{(LDA-PCA)}$	TCA	SSTCA	JCSL
Full	16.6	17.3	17.3	19.0	18.5	20.2
Splits	16.9 ± 2.1	17.6 ± 2.2	17.4 ± 2.1	19.2 ± 2.1	18.0 ± 2.2	20.5 ± 2.4

of classes: each step involves all the data (no per class sample selection). The final domain shift for JCSL is the average over the obtained separate shift values. The results in Table 2 indicate that SA and JCSL produce comparable results in terms of domain-shift reduction, suggesting that the main advantage of JCSL comes from the learned classifier.

5.3. Results - Bing+Caltech

Due to the way in which it was defined, Bing+Caltech can be considered as a much more challenging testbed for unsupervised domain adaptation compared to the other used datasets (see also Fig. 1). At the same time it also corresponds to one of the most realistic scenarios where domain adaptation is needed: we have access to only a limited number of noisy labeled source images obtained from the web and we want to use them to classify over a curated collection of object images. For this problem exploiting at the best all the available information is crucial. Specifically, since the source is not fully reliable, coding its discriminative information in the representation (e.g. through LDA or PLS) may be misleading. On the other hand, using the subspace of the non-noisy target data to guide the learning process can be much more beneficial.

As shown in Fig. 3, JCSL is the only method that consistently improves over the non-adaptive approach independently from the number of considered classes. TSL is always equivalent to NA, while the other subspace methods, although initially helpful for problems with few classes, lose their advantage over NA when the number of classes increases. This behavior is almost equivalent when using both 5 and 10 source samples per class.

5.4. Results - wifi localization

To demonstrate the generality of the proposed algorithm, we evaluate JCSL also on non-visual data. Since the wifi vector dimensionality (100) is lower than the number of classes (247), we do not exploit LDA here but we simply apply PCA to define the subspace dimensionality for both the source and target domains. The results on the

wifi-localization task are reported in Table 3 and show that domain adaptation is clearly beneficial.

TCA and SSTCA are the state of the art linear methods on the wifi dataset and they confirm their value even in the considered classification setting by outperforming SA and GFK. Still JCSL presents the best results. The obtained classification accuracy confirms the value of our method over the other subspace-based techniques.

6. Conclusions

Motivated by the theoretical results of Ben-David et al. [2], in this paper we proposed to integrate the learning process of the source prediction function with the optimization of the invariant subspace for unsupervised domain adaptation. Specifically, JCSL learns a representation that minimizes the divergence between the source subspace and the target subspace, while optimizing the classification model. Extensive experimental results have shown that, by taking advantage of the described principled combination and without the need of passing through the evaluation of the data distributions, JCSL outperform other subspace domain adaptation methods that focus only on the representation part.

Recently several works have demonstrated that Convolutional Neural Network classifiers are robust to domain shift [7,31]. Reasoning at high level we can identify the cause of such a robustness on the same idea at the basis of JCSL: deep architectures learn jointly a discriminative representation *and* the prediction function. The highly non-linear transformation of the original data coded into the CNN activation values can also be used as input data descriptors for JCSL with the aim of obtaining a combined effect. As future work we plan to evaluate principled ways to find automatically the best subspace dimensionality *d* using low-rank optimization methods.

Supplementary material

Supplementary material associated with this article can be found, in the online version, at 10.1016/j.patrec.2015.07.009.

References

[1] M. Baktashmotlagh, M.T. Harandi, B.C. Lovell, M. Salzmann, Unsupervised domain adaptation by domain invariant projection, in: Proceedings of the IEEE International Conference on Computer Vision (ICCV), 2013.

[2] S. Ben-David, J. Blitzer, K. Crammer, F. Pereira, Analysis of representations for domain adaptation, in: Proceedings of the Conference on Neural Information Processing Systems (NIPS), 2007.

[3] A. Bergamo, L. Torresani, Exploiting weakly-labeled web images to improve object classification: a domain adaptation approach, in: Proceedings of the Conference on Neural Information Processing Systems (NIPS), 2010.

[4] J. Blitzer, D. Foster, S. Kakade, Domain adaptation with coupled subspaces, in: International Conference on Artificial Intelligence and Statistics (AISTATS), 2011.

[5] J. Blitzer, R. McDonald, F. Pereira, Domain adaptation with structural correspondence learning, in: Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP), 2006.

[6] H. Daumé III, D. Marcu, Domain adaptation for statistical classifiers, J. Artif. Int. Res. 26 (2006) 101–126.

[7] J. Donahue, Y. Jia, O. Vinyals, J. Hoffman, N. Zhang, E. Tzeng, T. Darrell, Decaf: A deep convolutional activation feature for generic visual recognition, in: Proceedings of the International Conference in Machine Learning (ICML), 2014.

[8] L. Duan, I.W. Tsang, D. Xu, Domain transfer multiple kernel learning, IEEE Trans. Pattern Anal. Mach. Intell. 34 (2012) 465–479.

[9] L. Duan, I.W. Tsang, D. Xu, T.-S. Chua, Domain adaptation from multiple sources via auxiliary classifiers, in: Proceedings of the International Conference in Machine Learning (ICML), 2009.

[10] L. Duan, I.W. Tsang, D. Xu, S.J. Maybank, Domain transfer svm for video concept detection, in: Proceedings of the Conference on Computer Vision and Pattern Recognition (CVPR), 2009.

[11] B. Fernando, A. Habrard, M. Sebban, T. Tuytelaars, Subspace alignment for domain adaptation, CoRR abs/1409.5241 (2014). http://arxiv.org/abs/1409.5241.

[12] B. Fernando, A. Habrard, M. Sebban, T. Tuytelaars, et al., Unsupervised visual domain adaptation using subspace alignment, in: Proceedings of the International Conference on Computer Vision (ICCV), 2013.

[13] B. Gong, K. Grauman, F. Sha, Connecting the dots with landmarks: Discriminatively learning domain-invariant features for unsupervised domain adaptation, in: Proceedings of the International Conference in Machine Learning (ICML), 2013.

[14] B. Gong, Y. Shi, F. Sha, K. Grauman, Geodesic flow kernel for unsupervised domain adaptation, in: Proceedings of the Computer Vision and Pattern Recognition (CVPR), 2012.

[15] R. Gopalan, R. Li, R. Chellappa, Domain adaptation for object recognition: an unsupervised approach, in: Proceedings of the International Conference on Computer Vision (ICCV), 2011.

[16] A. Gretton, K. Borgwardt, M. Rasch, B. Schlkopf, A. Smola, A kernel method for the two sample problem, in: Proceedings of the Conference on Neural Information Processing Systems (NIPS), 2007.

[17] A. Gretton, K.M. Borgwardt, M.J. Rasch, B. Schölkopf, A. Smola, A kernel two-sample test, J. Mach. Learn. Res. 13 (2012) 723–773.

[18] G. Griffin, A. Holub, P. Perona, Caltech-256 object category dataset, Technical Report, California Institute of Technology, 2007.

[19] J. Hoffman, B. Kulis, T. Darrell, K. Saenko, Discovering latent domains for multi-source domain adaptation, in: Proceedings of the European Conference on Computer Vision (ECCV), 2012.

[20] J. Hoffman, E. Rodner, J. Donahue, K. Saenko, T. Darrell, Efficient learning of domain-invariant image representations, in: Proceedings of the International Conference on Learning Representations (ICLR), 2013.

[21] J. Hoffman, E. Tzeng, J. Donahue, Y. Jia, K. Saenko, T. Darrell, One-shot adaptation of supervised deep convolutional models, arXiv preprint arXiv:1312.6204 0 (2013) 1–8.

[22] J. Hull, A database for handwritten text recognition research, IEEE Trans. Pattern Anal. Mach. Intell. 16 (1994) 550–554.

[23] H.D. III, Frustratingly easy domain adaptation, in: Proceedings of the Association for Computational Linguistics (ACL), 2007.

[24] I.-H. Jhuo, D. Liu, D.T. Lee, S.-F. Chang, Robust visual domain adaptation with low-rank reconstruction, in: Proceedings of the Computer Vision and Pattern Recognition (CVPR), 2012, pp. 2168–2175.

[25] B. Kulis, K. Saenko, T. Darrell, What you saw is not what you get: Domain adaptation using asymmetric kernel transforms, in: Proceedings of the Computer Vision and Pattern Recognition (CVPR), 2011.

[26] A. Kumar, A. Saha, H. Daume, Co-regularization based semi-supervised domain adaptation, in: Proceedings of the Neural Information Processing Systems (NIPS), 2010, pp. 478–486.

[27] Y. LeCun, L. Bottou, Y. Bengio, P. Haffner, Gradient-based learning applied to document recognition, in: S. Haykin, B. Kosko (Eds.), Intelligent Signal Processing, IEEE Press, 2001, pp. 306–351.

[28] M. Long, J. Wang, G. Ding, J. Sun, P. Yu, Transfer feature learning with joint distribution adaptation, in: Proceedings of the International Conference on Computer Vision (ICCV), 2013.

[29] Y. Mansour, M. Mohri, A. Rostamizadeh, Domain adaptation: learning bounds and algorithms, in: Proceedings of the Conference on Learning Theory (COLT), 2009.

[30] J. Ni, Q. Qiu, R. Chellappa, Subspace interpolation via dictionary learning for unsupervised domain adaptation, in: Proceedings of the Computer Vision and Pattern Recognition (CVPR), 2013.

[31] M. Oquab, L. Bottou, I. Laptev, J. Sivic, Learning and transferring mid-level image representations using convolutional neural networks, in: Proceedings of the Computer Vision and Pattern Recognition (CVPR), 2014.

[32] S.J. Pan, I.W. Tsang, J.T. Kwok, Q. Yang, Domain adaptation via transfer component analysis, IEEE Trans. Neural Netw. 22 (2011) 199–210.

[33] V.M. Patel, R. Gopalan, R. Li, R. Chellappa, Visual domain adaptation: a survey of recent advances, IEEE Signal Process. Mag. (2014).

[34] K. Saenko, B. Kulis, M. Fritz, T. Darrell, Adapting visual category models to new domains, in: Proceedings of the European Conference on Computer Vison (ECCV), 2010.

[35] S. Shekhar, V.M. Patel, H.V. Nguyen, R. Chellappa, Generalized domain-adaptive dictionaries, in: Proceedings of the Computer Vision and Pattern Recognition (CVPR), 2013.

[36] H. Shimodaira, Improving predictive inference under covariate shift by weighting the log-likelihood function, J. Stat. Plan. Inference 90 (2000) 227–244.

[37] S. Si, D. Tao, B. Geng, Bregman divergence-based regularization for transfer subspace learning, IEEE Trans. Knowl. Data Eng. 22 (2010) 929–942.

[38] Q. Sun, R. Chattopadhyay, S. Panchanathan, J. Ye, A two-stage weighting framework for multi-source domain adaptation, in: Proceedings of the Neural Information Processing Systems (NIPS), 2011.

[39] A. Vedaldi, A. Zisserman, Efficient additive kernels via explicit feature maps, IEEE Trans. Pattern Anal. Mach. Intell. 34 (3) (2012) 480–492.

[40] C. Wang, S. Mahadevan, Heterogeneous domain adaptation using manifold alignment, in: Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI), 2011.

[41] J. Yang, R. Yan, A.G. Hauptmann, Cross-domain video concept detection using adaptive svms, in: Proceedings of the International Conference on Multimedia (ACM-MULTIMEDIA), 2007, pp. 188–197.

[42] Q. Yang, S.J. Pan, V.W. Zheng, Estimating location using wi-fi, IEEE Intell. Syst. 23 (2008) 8–13.