Optimal Transport for Domain Adaptation

Nicolas Courty, Rémi Flamary, Devis Tuia, Alain Rakotomamonjy

Optimal Transport for Domain Adaptation

Nicolas Courty, Remi Flamary, Devis Tuia, *Member, IEEE*, Alain Rakotomamonjy

Abstract—Domain adaptation from one data space (or domain) to another is one of the most challenging tasks of modern data analytics. If the adaptation is done correctly, models built on a specific data space become more robust when confronted to data depicting the same semantic concepts (the classes), but observed by another observation system with its own specificities. Among the many strategies proposed to adapt a domain to another, finding a common representation has shown excellent properties: by finding a common representation for both domains, a single classifier can be effective in both and use labelled samples from the source domain to predict the unlabelled samples of the target domain. In this paper, we propose a regularized unsupervised optimal transportation model to perform the alignment of the representations in the source and target domains. We learn a transportation plan matching both PDFs, which constrains labelled samples in the source domain to remain close during transport. This way, we exploit at the same time the few labeled information in the source and the unlabelled distributions observed in both domains. Experiments in toy and challenging real visual adaptation examples show the interest of the method, that consistently outperforms state of the art approaches.

Index Terms—Unsupervised Domain Adaptation, Optimal Transport, Transfer Learning, Visual Adaptation, Classification.

1 Introduction

The multiplication of data sources and acquisition devices allow to consider tremendous quantities of data. In practical applications, the wealth of data available is however often counterbalanced by the lack of annotated information, which is generally used to run classification algorithms aiming at generalizing over new unseen examples. Moreover, classical learning methods are challenged by the plurality of sources, mostly because subtle or pronounced discrepancies are observed in the different data distributions, or *drifts*. In computer vision, this problem is known as the visual adaptation problem, where domain to domain drifts may occur when changing lighting conditions, acquisition devices, or by considering the presence or absence of backgrounds. In speech processing, learning from one speaker and trying to deploy an application targeted to a wide public may also be hindered by the differences in background noise, style, tone or gender of the speaker. Another example is found in remote sensing image analysis, where one would like to reuse the labels defined over one city to classify the land occupation of another city. The drifts observed in the probability density function (PDF) of remote sensing images are due to a variety of factors: different corrections from atmosphere scattering compensation, daylight conditions at the hour of acquisition or even slight changes in the chemical composition of the materials. As a matter of fact, the causes of drift are numerous and specific to the application.

For those reasons, several works study the generalization capabilities of a classifier allowing to transfer knowledge from a labeled source domain to an unlabeled target domain: in our work, we assume that the source and target domains are by essence different, which is usually referred to as the domain adaptation problem [33]. We address the most difficult variant of this problem, where data labels are only available in the source domain. This is the **unsupervised domain adaptation** problem, and we tackle it by assuming that the effects of the drifts can be reduced if data undergo a phase of *adaptation* toward a common representation, where both domains look more alike.

Several theoretical works [1], [32], [19] have emphasized the role played by the divergence of the two domains probability distribution functions, leading to a principled way of solving the domain adaptation problem: moving closer both distributions, while using the label information available in the source domain to learn a classifier. This work follows the same intuition and tries to search for a transformation of the input data that fits **a least effort principle**, *i.e.* an effect that is minimal with respect to a transformation cost or metric. In this sense, the adaptation problem boils down to: *i)* finding a possible transformation of the input data to match the source and target distributions or transportation plan, and then *ii)* learning a new classifier from the transformed source samples. This process is depicted in Figure 1. The question is how to find the best transportation plan possible. In this paper, we propose a solution based on *optimal transport*.

- *NC is with University of Bretagne Sub, IRISA Laboratory E-mail: Nicolas.Courty@irisa.fr*
- *DT is with the Department of Geography, University of Zurich, Switzerland. E-mail: devis.tuia@geo.uzh.ch*
- *RF is with University of Nice Sophia Antipolis, OCA, Lagrange Laboratory, UMR CNRS 7293 , France. E-mail: remi.flamary@unice.fr*

Manuscript received January 2015; revised XXX.

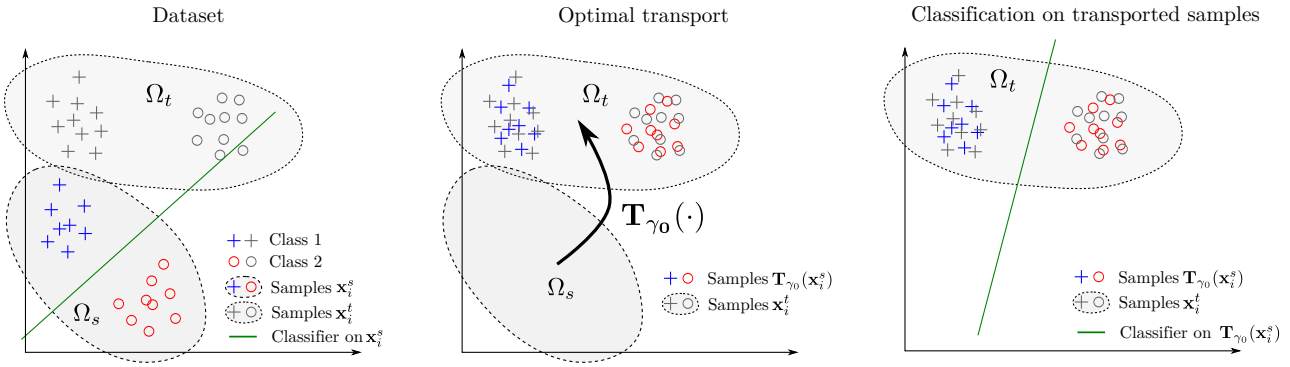


Fig. 1: Illustration of the proposed approach for domain adaptation. (left) dataset for learning, *i.e.* source domain, and testing, *i.e.* target domain. Note that a classifier estimated on the training examples clearly does not fit the target data. (middle) an optimal transportation plan is estimated and used to transport the training samples onto the target domain. Note that this transformation is usually not linear (right) the transported labeled samples are used for estimating a classifier in the target domain.

Optimal Transport (OT) problems have received recently a lot of interests in several fields, and among the potential applications of its theory, one resides in its use for computing distances between probability distributions. Those distances have received several appellations in the literature: Wasserstein, Monge-Kantorovich or Earth Mover distances, and have very strong and important properties: *i)* they can be evaluated when only empirical measures of those distributions are observed and without the estimation of parametrical or semi-parametrical distributions as a pre-process; *ii)* there are no particular constraints on the overlap of the support of the distributions to provide meaningful results, which is clearly not the case with most classical information theoretic divergences, such as the Kullback-Leibler divergence. By directly exploiting this property, we introduce a novel framework for unsupervised domain adaptation which consists in learning an optimal transportation based on empirical observations. In addition, we propose a series of regularizers for the optimal transport learning problem, building either upon the knowledge of class memberships in the source domain or upon Laplacian regularization. Finally, an efficient algorithm has been used for solving the resulting optimization problem.

The reminder of this Section presents the related works, while Section 2 formalizes the problem of unsupervised domain adaptation and the use of optimal transport for its resolution. Algorithms based on a variant of the conditional gradient optimization framework are proposed in Section 3 for regularized optimal transport. The originality of our approach resides in the inclusion of additional regularization terms tailored to fit the domain adaptation constraints. Their pertinence is thoroughly examined in the experimental Section 4, where we demonstrate the efficiency of the new proposed framework in both synthetic and real-world examples.

1.1 Related works

Domain adaptation. Domain adaptation strategies can be roughly divided in two families, depending on whether they assume the presence of few labels in the target domains (semi-supervised DA) or not (unsupervised DA).

In the first family, we find methods searching for projections discriminative in both domains, either by using dot products between the source samples and the transformed target samples [38], [28], [25], or by learning projections, for which labeled samples of the target domain fall on the correct side a large margin classifier trained on the source data [24] or by extracting common features under pairwise constraints [23], [44].

The second family tackles the domain adaptation problem assuming, as in this paper, that no labels are available in the target domain. Besides works dealing with sample reweighing [41], many works have considered finding a common feature representation for the two (or more) domains. This representation, or *latent space*, allows to project samples from all domains in a space where a classifier using only the labeled samples from the source domain generalize well on the target domains [16], [34]. The representation transfer can be performed by matching the means of the domains in the feature space [34], aligning the domains by their correlations [29] or by using pairwise constraints [43]. In most of these works, the common latent space is found via feature extraction, where the resulting projection summarizes the information common to the domains.

Recently, the unsupervised domain adaptation problem has been revisited by considering strategies based on a gradual alignment of a feature representation: in [21], authors start from the hypothesis that domain adaptation can be better approached if comparing gradual distortions and therefore use intermediary projections of both domains along the Grassmannian

geodesic connecting the source and target observed eigenvectors. In [20], [46], authors propose to obtain all sets of transformed intermediary domains by using a geodesic-flow kernel, instead of sampling a fixed number of projections along the geodesic path. While these methods have the advantage of providing easily computable out-of-sample extensions (by projecting unseen samples onto the latent space eigenvectors), the transformation defined remains global and is applied the same way to the whole target domain.

An approach combining the sample reweighing logic with the representation transfer one is found in [45], where authors extend the sample re-weighing in a reproducing kernel Hilbert space by the use of surrogate kernels. By doing so, a linear transformation of the domains is found, but, as for the feature representation approaches above, it stays equal for all transferred samples.

Our proposition strongly differs from those reviewed above, as it defines a local transportation plan for each sample in the source domain. In this sense, the domain adaptation problem can be seen as a graph matching problem [31], [9], [10] for all samples to be transported, where their final coordinates are found by mapping the source samples to target samples while respecting the marginal distribution of the target domain.

Optimal Transport and Machine Learning. The optimal transport (OT) problem has first been introduced by the french mathematician Gaspard Monge in the middle of the 19th century as the way to find a minimal effort solution to the transport of a given mass of dirt into a given hole. The problem reappeared in the middle of the 20th century in the work of Kantorovitch [26], and found recently surprising new developments as a polyvalent tool for several fundamental problems [42]. It was applied in a wide panels of fields, including computational fluid mechanics [5], color transfer between multiple images or morphing in the context of image processing [36], [17], [6], interpolation schemes in computer graphics [7], and economics, via matching and equilibriums problems [11].

Despite the appealing properties and application success stories, the machine learning community has considered optimal transport only recently (see, for instance, computing distances between histograms [14] or label propagation in graphs [40]); the main reason being the high computational cost induced by the computation of the optimal transportation plan. However, new computing strategies have emerged [14], [15], [6] and made possible the application of OT distances in operational settings.

2 Optimal transport and application to domain adaptation

In this section, we present the general unsupervised domain adaptation problem and show how it can be addressed from an optimal transport perspective.

2.1 Problem summary and theoretical motivations

Let $\Omega \in \mathbb{R}^d$ be an input measurable space of dimension d and $\mathcal{C}$ the set of possible labels. $\mathcal{P}(\Omega)$ denotes the set of all probability measures over Ω. The standard learning paradigm assumes the existence of a set of data $\mathbf{X}_s = \{\mathbf{x}_i^s\}_{i=1}^{N_s}$ associated with a set of class label information $\mathbf{Y}_s = \{y_i^s\}_{i=1}^{N_s}$, with $y_i^s \in \mathcal{C}$ (the training set) , and a data set with unknown labels $\mathbf{X}_t = \{\mathbf{x}_i^t\}_{i=1}^{N_t}$ (the testing set). In order to determine the set of labels $\mathbf{Y}_t$ associated with $\mathbf{X}_t$, one usually relies on an empirical estimate of the joint probability distribution $\mathbf{P}(\mathbf{x}, y) \in \mathcal{P}(\Omega \times \mathcal{C})$ from $(\mathbf{X}_s, \mathbf{Y}_s)$, and on the assumption that $\mathbf{X}_s$ and $\mathbf{X}_t$ are drawn from the same distribution $\mathbf{P}(\mathbf{x}) \in \mathcal{P}(\Omega)$.

2.2 Domain adaptation as a transportation problem

In the considered adaptation problem, one assumes the existence of two distinct joint probability distributions $\mathbf{P}_s(\mathbf{x}^s, y)$ and $\mathbf{P}_t(\mathbf{x}^t, y)$ which correspond respectively to two different *source* and *target* domains, noted respectively Ω_s and Ω_t. In the following, μ_s and μ_t are their respective marginal distributions over $\mathbf{X}$. When μ_s and μ_t are only accessible through discrete samples, the corresponding empirical distributions can be written as

$$\mu_s = \sum_{i=1}^{n_s} p_i^s \delta_{\mathbf{x}_i^s}, \quad \mu_t = \sum_{i=1}^{n_t} p_i^t \delta_{\mathbf{x}_i^t} \tag{1}$$

where $\delta_{\mathbf{x}_i}$ is the Dirac function at location $\mathbf{x}_i \in \mathbb{R}^d$. p_i^s and p_i^t are probability masses associated to the i-th sample, and belong to the probability simplex, *i.e.* $\sum_{i=1}^{n_s} p_i^s = \sum_{i=1}^{n_t} p_i^t = 1$. We also denote f_s and f_t the true labeling functions, *i.e.* the Bayes decision functions in each domain.

One of the following two assumptions is generally taken by most of the methods tackling the domain adaptation problem:

- **Class imbalance**: the label distributions are different in the two domains ($\mathbf{P}_s(y) \neq \mathbf{P}_t(y)$), but the conditional distributions of the samples with respect to the labels are the same ($\mathbf{P}_s(\mathbf{x}^s|y) = \mathbf{P}_t(\mathbf{x}^t|y)$);
- **Covariate shift**: the conditional distributions of the labels with respect to the data are the same ($\mathbf{P}_s(y|\mathbf{x}^s) = \mathbf{P}_t(y|\mathbf{x}^t)$, or equivalently $f_s = f_t = f$), but the distributions of the data in the two domains are different ($\mathbf{P}_s(\mathbf{x}^s) \neq \mathbf{P}_t(\mathbf{x}^t)$). In the practical cases they can be different, but this difference needs to be small [1] for the adaptation to work.

Those two views of the problem lead to different reweighting schemes that can be appropriately incorporated in the learning strategies (such as SVM [8]). Yet in real world applications, the drift occurring between the source and the target domains generally implies a change in both marginal and conditional distributions.

In our work, we propose the following modelling: the domain drift is due to an unknown transformation of the input space $\mathbf{T} : \mathbf{\Omega_s} \rightarrow \mathbf{\Omega_t}$. This transformation may have a physical interpretation (e.g. change in the acquisition conditions, sensor drifts, thermal noise, etc.) or be linked with the unknown process that generates the data. In any case, this transformation has no particular reason for being linear. Then we make the following assumption: the transformation preserves the conditional distribution, *i.e.*

$$\mathbf{P}_t(y|\mathbf{x}^s) = \mathbf{P}_s(y|\mathbf{T}(\mathbf{x}^s)).$$

In other words, the label information is preserved by the transformation, and $f_t(\mathbf{x}) = f_s(\mathbf{T}(\mathbf{x}))$. From a probabilistic point of view, $\mathbf{T}$ transforms the probability measure μ in its *image measure*, noted $\mathbf{T}\#\mu$, which is a probability measure over Ω_t, for which

$$\mathbf{T}\#\mu(\mathbf{x}) = \mu(\mathbf{T}^{-1}(\mathbf{x})), \quad \forall \mathbf{x} \in \Omega_t \tag{2}$$

$\mathbf{T}$ is said to be a **transport** from μ_s to μ_t if $\mathbf{T}\#\mu_s = \mu_t$ (as illustrated in Figure 2.a). Under this assumption, $\mathbf{X}_t$ are drawn from $\mathbf{T}\#\mu_s$. This provides a principled way to solve the adaptation problem:

1) Estimate μ_s and μ_t from $\mathbf{X}_s$ and $\mathbf{X}_t$ (Eq. (1))
2) Find a transportation plan $\mathbf{T}$ from μ_s to μ_t
3) Use $\mathbf{T}$ to transport samples $\mathbf{X}_s$, and use the transported samples with a classical approach in machine learning, *i.e.* directly estimating a decision function with a classifier, or use the posterior probability

$$\mathbf{P}_t(\mathbf{x}^t, y) = \mathbf{P}_s(y|\mathbf{T}(\mathbf{x}^s))\mathbf{P}_s(\mathbf{T}(\mathbf{x}^s)).$$

Searching the space of all possible transformations $\mathbf{T}$ is not possible, and other assumptions need to be taken. In order to be sufficiently general, we propose a simple restriction to the class of possible transformation: $\mathbf{T}$ should be chosen such that it is a minimum energy or transformation cost.

Such a cost $C(\mathbf{T})$ can be expressed as:

$$C(\mathbf{T}) = \int_{\Omega_s} c(\mathbf{x}, \mathbf{T}(\mathbf{x}))d\mu(\mathbf{x}), \tag{3}$$

where the cost function $c : \Omega_s \times \Omega_t \rightarrow \mathbb{R}^+$ can be taken as a distance function over the metric space Ω, and can be understood as the energy required to move a probability mass $\mu(\mathbf{x})$ from $\mathbf{x}$ to $\mathbf{T}(\mathbf{x})$. In our domain adaptation setting, this cost has a penalizing function for large displacements that would result in violations of the preservation of conditional distributions.

By adding constraints on those constraints, the problem of estimating $\mathbf{T}$ boils down to an optimal transportation problem. The **optimal transport** $\mathbf{T}_0$ is the solution of the following minimization problem:

$$\mathbf{T}_0 = \underset{\mathbf{T}}{\operatorname{argmin}} \int_{\Omega_s} c(\mathbf{x}, \mathbf{T}(\mathbf{x}))d\mu(\mathbf{x}), \quad \text{s.t. } \mathbf{T}\#\mu_s = \mu_t \tag{4}$$

which is the original Monge transportation problem. Let Π be the set of all probabilistic couplings $\in \mathcal{P}(\Omega_s \times \Omega_t)$ with marginals μ_s and μ_t. The equivalent Kantorovitch formulation of the optimal transport [26] seeks for a coupling $\boldsymbol{\gamma} \in \Pi$ between Ω_s and Ω_t:

$$\boldsymbol{\gamma}_0 = \underset{\boldsymbol{\gamma} \in \Pi}{\operatorname{argmin}} \int_{\Omega_s \times \Omega_t} c(\mathbf{x}^s, \mathbf{x}^t)d\boldsymbol{\gamma}(\mathbf{x}^s, \mathbf{x}^t) \tag{5}$$

In this formulation, $\boldsymbol{\gamma}$ can be understood as a joint probability measure with marginals μ_s and μ_t (depicted in Figure 2.b). $\boldsymbol{\gamma}_0$ is the unique solution to the optimal transport problem [42]. Interestingly, $\boldsymbol{\gamma}_0$ can be directly estimated from $\mathbf{X}_s$ and $\mathbf{X}_t$ as shown in the next Section. By the Kantorovich-Rubinstein theorem, $\boldsymbol{\gamma}_0$ is the transportation map that allows to define the **Wasserstein distance** of order 1 between μ_s and μ_t:

$$\begin{aligned} W_1(\mu_s, \mu_t) &\overset{\text{def}}{=} \inf_{\boldsymbol{\gamma} \in \Pi} \int_{\Omega_s \times \Omega_t} c(\mathbf{x}^s, \mathbf{x}^t)d\boldsymbol{\gamma}(\mathbf{x}^s, \mathbf{x}^t) \\ &= \inf_{\boldsymbol{\gamma} \in \Pi} \{ \mathop{\mathbb{E}}_{\mathbf{x}^s \sim \mu_s, \mathbf{x}^t \sim \mu_t} c(\mathbf{x}^s, \mathbf{x}^t) \} \end{aligned} \tag{6}$$

This distance, also known as the Earth Mover Distance in computer vision community [37], defines a metric over the space of integrable squared probability measures. W_1 takes values in $[0, \text{diam}(\Omega)]$, with $\text{diam}(\Omega)$ being the diameter of Ω, *i.e.*$\text{diam}(\Omega) = \sup_{\mathbf{x}^s, \mathbf{x}^t \in \Omega} c(\mathbf{x}^s, \mathbf{x}^t)$.

2.3 Theoretical justification of the using an optimal transport strategy

A first result in the theory of domain adaptation is due to Ben-David and colleagues [4], [3]. It provides a generalization bound in the target domain for a classifier trained in the source domain. This bound is composed of three terms: i) the classification error in the source domain, ii) a measure of similarity between the classification problems in source and target, and iii) an estimate of the divergence between μ_s and μ_t, expressed as the total variation divergence $d_1(\mu_s, \mu_t)$:

$$d_1(\mu_s, \mu_t) \overset{\text{def}}{=} 2 \sup_{B \subset \mathcal{B}} |\mu_s(B) - \mu_t(B)| \tag{7}$$

where $\mathcal{B}$ is the set of all measurable sets under μ_s and μ_t, and $f : \Omega \rightarrow \mathbb{R}$ satisfies $|f(x)| \leq 1$ in both domains[1]. As such, if $\mathbf{T}$ helps in reducing this divergence, a better performance can be expected for the adaptation. In order to do so, one can notice that there exist relations between the Wasserstein metric and the

1. Whenever Ω is a countable space, this definition leads to the L^1-norm between the two distributions

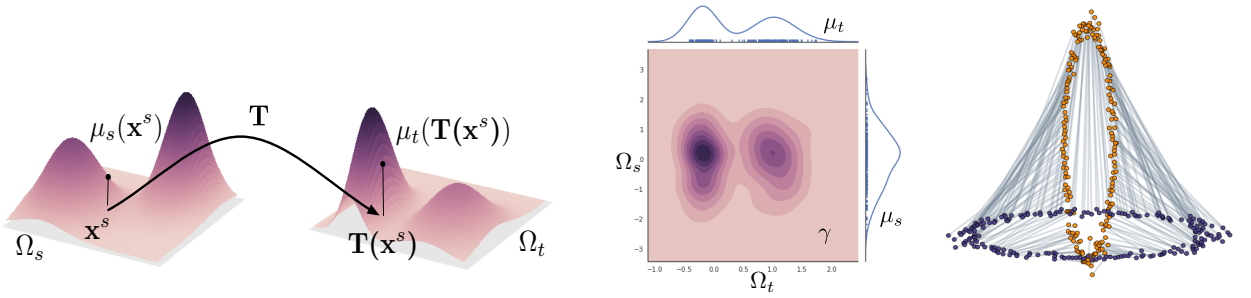


Fig. 2: Illustration of the optimal transport problem. (a) $\mathbf{T}$ is a transportation plan from Ω_s to Ω_t (b) $\boldsymbol{\gamma}$ can be seen as a joint probability distribution with marginals μ_s and μ_t (c) illustration of the transport plan computed between two ellipsoidal distributions. The grey line between two points indicate a non-zero coupling between those points.

total variation. In the general case of the Wasserstein distance, Theorem 6.15 from the monograph of C. Villani [42] shows the possible control of a Wasserstein distance and a **weighted** total variation. In our particular case, one way to understand this relation is to express the total variation in its equivalent coupling characterization, *i.e.*

$$d_1(\mu_s, \mu_t) \stackrel{\text{def}}{=} \inf_J \{ \mathop{\mathbb{E}}_{\mathbf{x}^s \sim \mu_s, \mathbf{x}^t \sim \mu_t} c(\mathbf{x}^s, \mathbf{x}^t) \mathbf{1}_{\mathbf{x}^s \neq \mathbf{x}^t} \}. \quad (8)$$

It is actually a particular case of equation (6) if one takes the cost function $c(\mathbf{x}^s, \mathbf{x}^t) = \mathbf{1}_{\mathbf{x}^s \neq \mathbf{x}^t}$. This allows to derive some relations between the two metrics:

Theorem 2.1: The Wasserstein metric and the total variation distance follow the relation:

$$W_1 \leq \text{diam}(\Omega) d_1. \quad (9)$$

When Ω is a finite set, and if $d_{min} = min_{\mathbf{x}^s \neq \mathbf{x}^t} c(\mathbf{x}^s, \mathbf{x}^t)$, one can define the reciprocal bound:

$$d_1 \leq \frac{1}{d_{min}} W_1 \quad (10)$$

As such, and since $\mathbf{T}$ reduces W_1 to zero by construction, we can expect $d_1(\mu_s, \mathbf{T}\#\mu_s)$ to be also minimised, and thus the adaptation to succeed.

3 Regularized discrete optimal transport

The goal of this Section is to present operational algorithms based on optimal transport for unsupervised domain adaptation. First we express the optimization problem that aims at computing the optimal transport plan based on the (discrete) empirical distributions. We then proceed by adding regularization terms that preserve label information and local proximity of the samples during transport.

3.1 Discrete optimal transport

Usually one does not have a direct access to μ_s or μ_t but rather to collections of samples from those distributions. It is then straightforward to adapt the optimal transport problem to the discrete case. We will denote $\mathcal{B}$ the set of probabilistic couplings between the two empirical distributions, and defined as:

$$\mathcal{B} = \left\{ \boldsymbol{\gamma} \in (\mathbb{R}^+)^{\mathbf{n_s} \times \mathbf{n_t}} \,|\, \boldsymbol{\gamma} \mathbf{1}_{\mathbf{n_t}} = \mu_\mathbf{s}, \boldsymbol{\gamma}^\mathbf{T} \mathbf{1}_{\mathbf{n_s}} = \mu_\mathbf{t} \right\} \quad (11)$$

where $\mathbf{1}_d$ is a d-dimensional vector of ones. The Kantorovitch formulation of the optimal transport [26] reads:

$$\boldsymbol{\gamma}_0 = \mathop{\text{argmin}}_{\boldsymbol{\gamma} \in \mathcal{B}} \langle \boldsymbol{\gamma}, \mathbf{C} \rangle_F \quad (12)$$

where $\langle ., . \rangle_F$ is the Frobenius dot product and $\mathbf{C} \geq 0$ is the cost function matrix, whose term $C(i, j)$ denotes the energy needed to move a probability mass from $\mathbf{x}_i^s$ to $\mathbf{x}_j^t$. In our setting, this cost was chosen as the Euclidian distance between the two locations, i.e. $C(i, j) = ||\mathbf{x}_i^s - \mathbf{x}_j^t||^2$, but other types of metric could be considered, such as Riemannian distances over a manifold [42].

Note that when $n_s = n_t = n$ and $\forall i, j \;\; p_i^s = p_j^t = 1/n$, $\boldsymbol{\gamma}_0$ is simply a permutation matrix. the optimal transport problem boils down to an optimal assignment problem. In the general case, it can be shown that $\boldsymbol{\gamma}_0$ is a sparse matrix with at most $n_s + n_t - 1$ non zero entries, that is the rank of the constraint matrix.

This problem can be solved by linear programming, with combinatorial algorithms such as the simplex methods and its network variants (transport simplex, network simplex, etc.). Yet, the computational complexity was shown to be $\mathcal{O}(n^2)$ in practical situations [7] for the network simplex (while being $\mathcal{O}(n^3)$ in theory), which dampens the utility of the method when handling big data. However, the regularization scheme recently proposed by Cuturi [14] allows a very fast transport computation. We will present this regularized scheme in Section 3.2.

Once the transport $\boldsymbol{\gamma}_0$ has been computed, the source samples must be transported in the target domain using their transportation plan. One can seek the interpolation of the two distributions by following the geodesics of the Wasserstein metric [42] (parameterized by $t \in [0,1]$). This allows to define a new distribution $\hat{\mu}$ such that:

$$\hat{\mu} = \underset{\mu}{\operatorname{argmin}} (1-t) W_1(\mu_s, \mu)^2 + t W_1(\mu_t, \mu)^2. \quad (13)$$

One can show that this distribution is [42]:

$$\hat{\mu} = \sum_{i,j} \boldsymbol{\gamma}_0(i,j) \delta_{(1-t)\mathbf{x}_i^s + t\mathbf{x}_j^t}. \quad (14)$$

In our approach, we suggest to compute directly the image of the source samples as the result of this transport, *i.e.* for $t = 1$. Yet, it is undesirable to split the source samples into several fraction elements as required by the transport. Instead, we propose to compute images of the source samples as barycenters of the target samples. The corresponding barycentric coordinates are readily found in $\boldsymbol{\gamma}_0$. Let $\mathbf{T}_{\boldsymbol{\gamma}_0} : \mathbb{R}^d \rightarrow \mathbb{R}^d$ be the mapping induced by the optimal transport coupling. This map transforms the source elements $\mathbf{X}_s$ in a target domain dependent version, $\hat{\mathbf{X}}_s$. The mapped samples can therefore be expressed as:

$$\hat{\mathbf{X}}_s = \mathbf{T}_{\boldsymbol{\gamma}_0}(\mathbf{X}_s) = \operatorname{diag}((\boldsymbol{\gamma}_0 \mathbf{1}_{n_t})^{-1}) \boldsymbol{\gamma}_0 \mathbf{X}_t. \quad (15)$$

We note that the reverse mapping from the target to the source domain can also be easily computed from $\boldsymbol{\gamma}_0^T$. As a matter of fact, one can show [15] that this transformation is a first order approximation of the true n_s Wasserstein barycenters of the target distributions. Finally, when the marginal μ_s and μ_t are uniform, one can easily derive the interpolation as a linear expression:

$$\hat{\mathbf{X}}_s = n_s \boldsymbol{\gamma}_0 \mathbf{X}_t \quad \text{and} \quad \hat{\mathbf{X}}_t = n_t \boldsymbol{\gamma}_0^\top \mathbf{X}_s \quad (16)$$

for the source and target samples.

3.2 Regularized optimal transport

Regularization is a classical approach used to prevent overfitting when only few samples are available, or even in presence of outliers. While it is always possible to enforce *a posteriori* a given regularity in the transport result, a more theoretically convincing solution is to regularize the transport by considering and additional penalty term in the problem formulation of Eq.(12).

More specifically, in [14] it is proposed to regularize the expression of the transport by the entropy of the probabilistic coupling. This information-theoretic regularized version of the transport $\boldsymbol{\gamma}_0^\lambda$ is the solution of the following minimization problem:

$$\boldsymbol{\gamma}_0^\lambda = \underset{\boldsymbol{\gamma} \in \mathcal{B}}{\operatorname{argmin}} \langle \boldsymbol{\gamma}, \mathbf{C} \rangle_F + \lambda \Omega_s(\boldsymbol{\gamma}), \quad (17)$$

where $\Omega_s(\boldsymbol{\gamma}) = \sum_{i,j} \boldsymbol{\gamma}(i,j) \log \boldsymbol{\gamma}(i,j)$ computes the negentropy of $\boldsymbol{\gamma}$. The intuition behind this form of regularization is the following: since most of the elements of $\boldsymbol{\gamma}_0$ should be zero with high probability, one can look for a smoother version of the transport whiches relaxes the sparsity of $\boldsymbol{\gamma}$ by increasing its entropy. As a result, and contrarily to the previous approach, more couplings with non-null weights are allowed, leading to a denser coupling between the distributions. An appealing result of this formulation is the possibility to derive a computationally very efficient algorithm, which uses the scaling matrix approach of Sinkhorn-Knopp [27]. In addition $\Omega_s(\cdot)$ can be seen as a Kullback-Leibler divergence $KL(\boldsymbol{\gamma} \| \boldsymbol{\gamma}_u)$ between the joint probability $\boldsymbol{\gamma}$ and a uniform joint probability $\boldsymbol{\gamma}_u$ such that $\gamma_u(i,j) = \frac{1}{n_s n_t}$. In this case we get $\Omega_s(\boldsymbol{\gamma}) = \log n_s n_t + \sum_{i,j} \boldsymbol{\gamma}(i,j) \log \boldsymbol{\gamma}(i,j)$ where the first term is a constant *w.r.t.* $\boldsymbol{\gamma}$.

As the parameter λ weighting the entropy-based regularization increases, the corresponding sparsity of $\boldsymbol{\gamma}_0^\lambda$ decreases and source points tend to dispatch their probability masses toward more target points. As a side effect of strong entropy-based regularization, the corresponding transport tends to gather every transported source points toward the center of mass of the target points.

3.3 Regularizing the transport with class labels

Optimal transport, as presented in the Section above, aims at minimizing a transport cost related to a metric between distributions. It does not include any information about the particular nature of the elements of the source domain (*e.g.* the fact that those samples belong to different classes). However, this information is generally available, as labeled samples are used in the classification step following the adaptation. Our proposition is to take advantage of label information and to penalize couplings that match together samples with different labels.

To this end, we propose to add a new term to the regularized optimal transport leading to the following optimization problem

$$\min_{\boldsymbol{\gamma} \in \mathcal{B}} \quad \langle \boldsymbol{\gamma}, \mathbf{C} \rangle_F + \lambda \Omega_s(\boldsymbol{\gamma}) + \eta \Omega_c(\boldsymbol{\gamma}), \quad (18)$$

where $\eta \geq 0$ and $\Omega_c(\cdot)$ is a class-based regularization term.

Specifically, in this work we propose and study two choices for this regularizer $\Omega_c(\cdot)$. The first one is based on group sparsity and promotes transportation, where a target sample has to be represented in $\boldsymbol{\gamma}$ by samples from the same class. The second is based on graph Laplacian regularization and promotes a locally smooth and class-regular structure in the transported samples.

3.3.1 Regularization with group-lasso

As introduced above, this first solution for label regularization is based on regularization inducing group-sparsity. We want to integrate the available label

information in the optimal transport. In our case, this information is available for all the source samples. The main idea is that, even if we do not know the class of the target distribution, we still can promote group sparsity in the columns of $\boldsymbol{\gamma}$ such that a given target point (i.e., one column of $\boldsymbol{\gamma}$) will be transported in the convex hull of only one class, since the sparsity support would consist in lines of $\boldsymbol{\gamma}$ belonging to the same class. Note that this approach has been proposed in our preliminary work [13]. We proposed to use the $\ell_p - \ell_1$ regularization with $p < 1$ in [13] mainly for algorithmic reasons. Indeed, when applying a Majoration-Minimization on the $\ell_p - \ell_1$, the problem can be casted as problem (17) and can be solved using the efficient Sinkhorn-Knopp solver at each iteration. But this regularization term is non-convex and the proposed algorithm only converges to a local stationary point.

In this work, we propose to maintain the convexity of the cost function and use the convex group lasso regularization $\ell_1 - \ell_2$ instead. This regularizer is defined as

$$\Omega_c(\boldsymbol{\gamma}) = \sum_j \sum_c ||\boldsymbol{\gamma}(\mathcal{I}_c, j)||_2, \tag{19}$$

where $\mathcal{I}_c$ contains the indices of the lines related to samples of the class c in the source domain, $\boldsymbol{\gamma}(\mathcal{I}_c, j)$ is a vector containing coefficients of the jth column of $\boldsymbol{\gamma}$ associated to class c and $||\cdot||_2$ denotes the ℓ_2 norm. Note that among other benefits, the convexity of the cost function allows to use a generic optimization scheme (presented in section 3.4) that can be used in practice for all the regularization schemes proposed in this paper.

3.3.2 Laplacian regularization

The second regularization we investigate in this work is the graph Laplacian regularization. This regularization preserves the structure of the data – approximated by a graph – during the transport [17], [12]. Note that the graph regularization is defined by symmetric similarity matrices $\mathbf{S}_s$ and $\mathbf{S}_t$ computed between the source and target samples in their original configurations. Indeed, $\mathbf{S}_s$ conveys an interesting mean to encode the class labels in the regularization by enforcing an absence of connection between elements of different classes. To this end, we can set the connections of $\mathbf{S}_s$ to 0 if the corresponding samples are from different classes.

When regularizing *w.r.t.* the source samples only, the regularization term becomes

$$\Omega_c(\boldsymbol{\gamma}) = \frac{1}{N_s^2} \sum_{i,j} S_s(i,j) \|\hat{\mathbf{x}}_i^s - \hat{\mathbf{x}}_j^s\|^2 \tag{20}$$

where $S_s(i,j) \geq 0$ are the coefficients of matrix $\mathbf{S}_s \in \mathbb{R}^{N_s \times N_s}$ that encodes the similarity between the source samples of the same class. Note that the sample $\hat{\mathbf{x}}_i^s$ used in this regularization term is the transported source sample $\mathbf{x}_i^s$ that linearly depends on the transportation matrix $\boldsymbol{\gamma}$ (Eq. (15)). Note that a similar Laplacian regularization term that controls the sample displacements has been proposed in [17] to adapt image histograms through the transportation of the pixels from one image to another. In [17], authors proposed a slightly different regularization term than (20) and no hard class enforcement was proposed.

When the marginals are uniform, one can express the regularization with

$$\Omega_c(\boldsymbol{\gamma}) = \text{Tr}(\mathbf{X}_t^\top \boldsymbol{\gamma}^\top \mathbf{L}_s \boldsymbol{\gamma} \mathbf{X}_t) \tag{21}$$

where $\mathbf{L}_s = \text{diag}(\mathbf{S}_s \mathbf{1}) - \mathbf{S}_s$ is the Laplacian of the graph $\mathbf{S}_s$. The regularizer is therefore quadratic *w.r.t.* $\boldsymbol{\gamma}$. Finally, the regularization term (20) is only applied on the transported source samples, thus leading to a non-symmetric regularization. When a structure exists also in the target samples (defined by matrix $\mathbf{S}_t$), a symmetric Laplacian regularization of the form

$$\Omega_c(\boldsymbol{\gamma}) = (1-\alpha)\text{Tr}(\mathbf{X}_t^\top \boldsymbol{\gamma}^\top \mathbf{L}_s \boldsymbol{\gamma} \mathbf{X}_t) + \alpha \text{Tr}(\mathbf{X}_s^\top \boldsymbol{\gamma} \mathbf{L}_t \boldsymbol{\gamma}^\top \mathbf{X}_s) \tag{22}$$

can be used instead. In the equation above $\mathbf{L}_t = \text{diag}(\mathbf{S}_t \mathbf{1}) - \mathbf{S}_t$ is the Laplacian of the graph in the target domain and $0 \leq \alpha \leq 1$ is a regularization parameter that weights the importance of each regularization term. Note that, since the target domain does not have labeled pixels, the $\mathbf{S}_t$ similarity matrix cannot be modified according to the class structure, as it is done for the source matrix $\mathbf{S}_s$.

3.4 Conditional Gradient Splitting for solving the regularized OT

In this section we propose an efficient algorithm for solving the optimization problem (18), when any of the group-lasso or Laplacian regularization is in play. But, at first, let us briefly characterize the existence of a solution to the problem.

For continuous regularizers as those we are considering in this work, the objective function is continuous. Since the doubly stochastic constraint set is a convex, closed and bounded, hence compact, subset of $\mathbb{R}^d$, the objective function reaches its mininum on $\mathcal{B}$. If, as for the Laplacian regularization, the regularizer is strictly convex, that minimum is unique.

Now, regarding algorithms for retrieving the optimal transport problem (18), we can note that for the Laplacian regularization, Ferradans et al. [17] proposed to use a conditional gradient (CG) algorithm. This approach is appealing and could be extended to our problem. Indeed, a CG approach has several nice properties that are of high interest. It is an iterative scheme that guarantees any iterate to belong to $\mathcal{B}$, which means that any of those iterates is a transport plan. In addition, at each iteration, for finding a feasible search direction, it looks for a minimizer of a linearized approximation of the objective function.

Hence, at each iteration it solves a Linear Program (LP) that is easier to handle than the original regularized optimal transport problem. Nevertheless, and despite the availability of efficient LP solvers such as CPLEX or MOSEK, the dimensionality of the problem that can be solved is limited, since it involves a LP problem of size $n_s \times n_t$.

In practice, we would like to have a more efficient algorithm that can handle a larger number of samples. Since we know how to solve efficiently problem (17) thanks to the Sinkhorn-Knopp algorithm proposed by Cuturi [14], we propose an algorithmic solution exploiting this efficient solver. To this end, we introduce a novel algorithm dubbed as conditional gradient splitting (CGS). The framework of the CGS algorithm addresses the general case of constrained minimization of composite functions as defined by

$$\min_{\boldsymbol{\gamma} \in \mathcal{B}} \quad f(\boldsymbol{\gamma}) + g(\boldsymbol{\gamma}), \tag{23}$$

where both $f(\cdot)$ and $g(\cdot)$ are convex and differentiable functions and $\mathcal{B}$ denotes any convex and compact subset of $\mathbb{R}^n$. Because the full analysis of this algorithm goes beyond the scope of this paper, we refer the interested reader to [18] which details its convergence properties as well as different case studies. In what follows, we briefly describe the CGS algorithm, along with its application to problem (18).

The CGS is illustrated in Algorithm 1. Its main difference with a classical conditional gradient algorithm relies in the way the search direction is built. The main idea of CGS is to linearize only a part of the composite objective function, for instance $f(\cdot)$ in Equation (23), instead of the full objective. This idea is justified by the hypothesis that solving the resulting nonlinear optimization problem can be done with an efficient solver. The other steps of the algorithm are exactly the same as those used for conditional gradient. We have shown in [18] that any limit point of the iterates $\{\boldsymbol{\gamma}^k\}$ is a minimizer of problem (23).

More specifically, for problem (18) one can set

$$f(\boldsymbol{\gamma}) = \langle \boldsymbol{\gamma}, \mathbf{C} \rangle_F + \eta \Omega_c(\boldsymbol{\gamma}) \quad \text{and} \quad g(\boldsymbol{\gamma}) = \lambda \Omega_s(\boldsymbol{\gamma})$$

Supposing now that $\Omega_c(\boldsymbol{\gamma})$ is differentiable, the step 3 of Algorithm 1 boils down to be

$$\boldsymbol{\gamma}^\star = \underset{\boldsymbol{\gamma} \in \mathcal{B}}{\operatorname{argmin}} \quad \left\langle \boldsymbol{\gamma}, \mathbf{C} + \eta \nabla \Omega_c(\boldsymbol{\gamma}^k) \right\rangle_F + \lambda \Omega_s(\boldsymbol{\gamma})$$

Interestingly, this problem is an entropy-regularized optimal transport problem similar to the one in Equation (17) and as such it can be efficiently solved using the Sinkhorn-Knopp scaling matrix approach. This shows that our CGS algorithm allows us to leverage on that efficient algorithm for proposing another efficient algorithm that solves the class-label regularization-based optimal transport problem.

Note that for our optimal transport problem, the Laplacian regularization is differentiable, whereas the group-lasso one is not when there exists a class c and an index j for which $\boldsymbol{\gamma}(\mathcal{I}_c, j)$ is a vector of 0. For dealing with this issue, one can note that if $\boldsymbol{\gamma}^k$ is so that $\boldsymbol{\gamma}^k(\mathcal{I}_c, j) \neq 0 \ \forall c, \forall j$ then the same property holds for $\boldsymbol{\gamma}^{k+1}$. This is due to the exponentiation occurring in the Sinkhorn-Knopp algorithm used for the entropy-based regularizer optimal transport problem. This means that if we initialize $\boldsymbol{\gamma}^0$ so that $\boldsymbol{\gamma}^0(\mathcal{I}_c, j) \neq 0$, then $\Omega_c(\boldsymbol{\gamma}^k)$ is always differentiable at any iteration k. It also means that $\boldsymbol{\gamma}$ becomes group-sparse only when k tends toward ∞. Hence, our CGS algorithm can also be applied to the group-lasso regularization despite its non-differentiability in 0.

Algorithm 1 Conditional gradient splitting (CGS)

1: Initialize $k = 0$ and $\boldsymbol{\gamma}^0 \in \mathcal{P}$
2: **repeat**
3: With $\mathbf{G} \in \nabla f(\boldsymbol{\gamma}^k)$, solve

$$\boldsymbol{\gamma}^\star = \underset{\boldsymbol{\gamma} \in \mathcal{B}}{\operatorname{argmin}} \quad \langle \boldsymbol{\gamma}, \mathbf{G} \rangle_F + g(\boldsymbol{\gamma})$$

4: Find the optimal step with $\Delta\boldsymbol{\gamma} = \boldsymbol{\gamma}^* - \boldsymbol{\gamma}^k$

$$\alpha^k = \underset{0 \leq \alpha \leq 1}{\operatorname{argmin}} f(\boldsymbol{\gamma}^k + \alpha \Delta\boldsymbol{\gamma}) + g(\boldsymbol{\gamma}^k + \alpha \Delta\boldsymbol{\gamma})$$

5: $\boldsymbol{\gamma}^{k+1} \leftarrow \boldsymbol{\gamma}^k + \alpha^k \Delta\boldsymbol{\gamma}$, set $k \leftarrow k + 1$
6: **until** Convergence

4 Numerical experiments

In this Section, we study the behavior of four different versions of optimal transport applied to the DA problem. In the rest of the section, **OT-exact** is the original transport problem (12), **OT-IT** the Information theoretic regularisation (17), and our two novel classes regularizers are **OT-GL** (Eq. (19)) and **OT-Laplace** (Eq. (20)), corresponding to the Group-Lasso and Laplacian regularizations, respectively.

4.1 Two moons: simulated problem with controllable complexity

In the first experiment, we consider the same toy example as considered in [19]. The simulated dataset consists of two domains: for the source, the standard two entangled moons data, where each moon is associated to a specific class (See Fig. 3(a)). The target domain is built by applying a rotation to the two moons, which allows to consider an adaptation problem with an increasing difficulty as a function of the rotation angle. This example is notably interesting because the corresponding transformation is clearly nonlinear, and because the input dimensionality is small, 2, which leads to poor performances when applying methods based on subspace alignment (*e.g.* [20], [30]).

We follow the same experimental protocol as in [19]. This allows for a direct comparison with the state-of-the-art results presented therein.

	10°	20°	30°	40°	50°	70°	90°
SVM (no adapt.)	0	0.104	0.24	0.312	0.4	0.764	0.828
DASVM [8]	0	0	0.259	0.284	0.334	0.747	0.82
PBDA [19]	0	0.094	0.103	0.225	0.412	0.626	0.687
OT-exact	0	0.028	0.065	0.109	0.206	0.394	0.507
OT-IT	0	0.007	0.054	0.102	0.221	0.398	0.508
OT-GL	0	0	0	0.013	0.196	0.378	0.508
OT-Laplace	0	0	0.004	0.062	0.201	0.402	0.524

TABLE 1: Mean error rate over 10 realizations of a 2-classes classification problem. The target domain is given by a rotation of the source domain of angle given in the first row.

The source domain is composed of two moons of 150 samples each. The target domain is also sampled from these two shapes, with the same number of individuals. Then, the generalization power of our method is tested over a set of 1000 samples that follow the same distribution as the target domain. The experiments are conducted 10 times, and we consider the mean classification error as a comparison criterium. As classifier, we used a SVM with a Gaussian kernel, whose parameters were set by 5-fold cross-validation. We compare the adaptation results with two state-of-the-art methods: the DA-SVM approach [8] and the more recent PBDA [19], which has proved to provide competitive results over this dataset. Results are reported in Table 1.

Our first observation is that all the methods based on optimal transport behave better than the state-of-the-art methods, in particular for low rotation angles, where results indicate that the geometrical structure is better preserved through the adaptation by optimal transport. Also, for large angle (*e.g.* $90°$), the final score is also significantly better than other state-of-the-art method, but falls down to a 0.5 error rate, which is natural since in this configuration a transformation of $-90°$, with an inversion of labels, would have led to similar empirical distributions. This clearly shows the capacity of our method to handle large domain transformations. Adding the class-label information into the regularization also clearly helps for the mid-range angle values, where as expected the adaptation shows nearly optimal results up to angles $< 40°$. This suggests that our optimal transport strategies are able to handle important non-linear transformations of low dimensional spaces. For the strongest deformation ($> 70°$ rotation), no clear winner among the OT methods can be found. We think that, regardless of the amount and type of regularization chosen, the classification of test samples becomes too much tributary of the training samples, which mostly come from the denser part of μ_s and that, as a consequence, the less dense parts of this PDF are less satisfactorily transported. This behavior can be seen in Figure 3d.

4.2 Visual adaptation datasets

We now evaluate our method on three challenging real world vision adaptation tasks, which have attracted a lot of interest in recent vision literature [35]. They all aim at visual recognition tasks of different categories of objects: digits, faces and various objects. We start by presenting the datasets, then the experimental protocol, and finish by providing and discussing the obtained results, as well as the computational cost of the OT-based methods.

4.2.1 Datasets

Three types of image recognition problems are considered: digits, faces and miscellaneous objects recognition. This choice of datasets was already featured in [30], and we follow the same preparation procedure. A summary of the properties of each domain considered in the three problems is provided in Table 2. An illustration of some examples of the different domains for a particular class is shown in Figure 4.

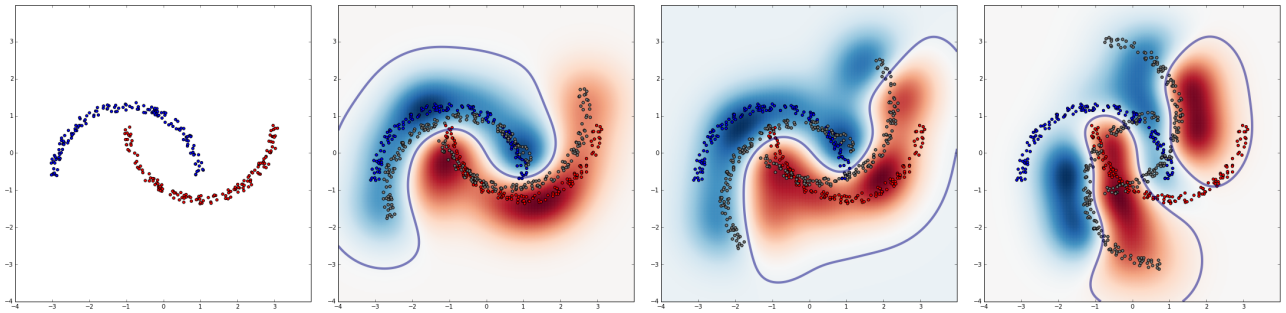
Digit recognition. As source and target domains, we use the two digits datasets USPS and MNIST, that share 10 classes of digits (single digits $0-9$). We randomly sampled $1,800$ and $2,000$ images from each original dataset. The MNIST images are resized to the same resolution as that of USPS (16×16). The grey levels of all images are then normalized to obtain a final common feature space for both domains.

Face recognition. In the face recognition experiment, we take the PIE ("Pose, Illumination, Expression") dataset, which contains 32×32 images of 68 individuals taken under various pose, illumination and expressions conditions. The 5 experimental domains are constructed by selecting 5 distinct poses: PIE05 (C05, left pose), PIE07 (C07, upward pose), PIE09 (C09, downward pose), PIE27 (C27, frontal pose) and PIE29 (C29, right pose). This allows to define 20 different adaptation problems with increasing difficulty (the most challenging being the adaptation from right to left poses). Let us note that each domain has a strong variability for each class due to illumination and expression variations.

Object recognition. We used the Caltech-Office dataset, which has been extensively studied in several papers [38], [21], [20], [46], [35]. The dataset contains images coming from four different domains: *Amazon* (online merchant), the *Caltech-256* image collection [22], *Webcam* (images taken from a webcam) and *DSLR* (images taken from a high resolution digital SLR camera). A feature extraction method is used to preprocess those images; it operates by computing SURF descriptors [38], then used to transform each image into a 800 bins histogram. The histograms are then subsequently normalized and reduced to standard scores. The variability of the different domains come from several factors: presence/absence of background, lightning conditions, noise, etc.

4.2.2 Experimental setup

Following [20], the classification is conducted using a 1-Nearest Neighbor (1NN) classifier, which has the



(a) source domain (b) rotation=20° (c) rotation=40° (d) rotation=90°

Fig. 3: Illustration of the classification decision boundary produced by **OT-Laplace** over the two moons example for increasing rotation angles. The source domain is represented as coloured points. The target domain is depicted as points in grey (best viewed with colors).

Problem	Domains	Dataset	# Samples	# Features	# Classes	Abbr.
Digits	USPS	USPS	1800	256	10	U
	MNIST	MNIST	2000	256	10	M
Faces	PIE05	PIE	3332	1024	68	P1
	PIE07	PIE	1629	1024	68	P2
	PIE09	PIE	1632	1024	68	P3
	PIE27	PIE	3329	1024	68	P4
	PIE29	PIE	1632	1024	68	P5
Objects	Calltech	Calltech	1123	800	10	C
	Amazon	Office	958	800	10	A
	Webcam	Office	295	800	10	W
	DSLR	Office	157	800	10	D

TABLE 2: Summary of the domains used in the visual adaptation experiment

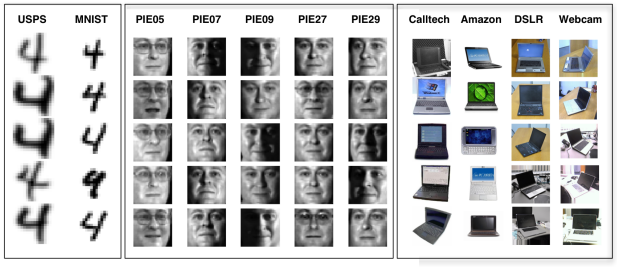


Fig. 4: Illustration of the datasets used in the visual adaptation experiment. 5 random samples from one class are given for all the considered domains.

advantage of being parameter free. In all experiments, 1NN is trained with the adapted source data, and evaluated over the target data to provide a classification accuracy score. We compare our optimal transport solutions to the following baseline methods that are particularly well adapted for image classification:

- **1NN** is the original classifier without adaptation and constitutes a baseline for all experiments;
- **PCA**, which consists in applying a projection on the first principal components of the joint source/target distribution (estimated from a concatenation of source and target samples);
- **GFK**, Geodesic Flow Kernel [20];
- **TCA**, Transfer Component Analysis [34];
- **TSL**, Transfer Subspace Learning [39];
- **JDA**, Joint Distribution Adaptation [30].

In the unsupervised version of the DA we consider, no target labels are available. As a consequence, it is impossible to consider a cross-validation step for the hyperparameters of the different methods. However, and in order to compare the methods at their best, we selected the best possible parameters for each method by crossvalidation as proposed in [30]. In the case of subspace learning methods (**PCA**,**TCA**,**GFK**, **TSL** and **JDA**), we search for reduced k-dimensional spaces with $k \in \{10, 20, \dots, 200\}$. For **JDA**, we follow the guidelines of [30], since they follow the same experimental protocol. For the transfer learning approach that have a regularization parameter λ, the best value was searched in $\{0.001, 0.01, 0.1, 1, 10, 100, 1000\}$. The λ and η parameters of our different regularizers (Eq. (18)), are validated using the same search in-

terval. In the case of the Laplacian regularization (**OT-Laplace**), $\mathbf{S}_t$ is a binary matrix which encodes a nearest neighbors graph with a 8-connectivity. For the source domain, since the labels are available, $\mathbf{S}_s$ is filtered such that connections between elements of different classes are pruned. Finally, we set the α value Eq. (22) to 0.5 to both the shape of the source and target.

4.2.3 Results

Results of the experiment are reported in Table 3 where the best performing method for each domain adaptation problem is highlighted in bold. First, we observe that, on average, all the OT-based domain adaptation methods perform better than the baseline methods. Even **OT-Exact**, which is the worst performing OT-based strategy, is outperformed only slightly by the best baseline method (**JDA**). The best performing strategies are clearly **OT-GL** and **OT-Laplace** with a slight advantage for **OT-GL**. Interestingly, among the baseline methods, those focusing on a joint projection of the data (**TCA**, **TSL** and **JDA**) seem to perform better than the Geodesic Flow Kernel (**GFK**). When considering each adaptation dataset on average, one can see that the OT-based strategies lead only to a small increment in performance on the digit recognition problem, while on the faces and object recognition the gain of the OT-based strategies is much more important and provide the best results in 18/20 and 11/12 DA problems, respectively.

Finally, since we have a large number of experiments, we can compare the best performing method, **OT-GL**, to all the other approaches using a Wilcoxon signrank test. This test evaluates the probability that the difference in performance between two methods has a 0 median with high probability. A small value of the p-value means that the median is different from 0. With an α risk of 1%, **OT-GL** has been tested as statistically better than all the other approaches except **OT-Laplace**.

4.2.4 Computational performances

Let us first show that the CGS algorithm is more efficient than a classical conditional gradient method as the one used in [17] . We illustrate this in Figure 5, showing the convergence (top panel) and the corresponding computational times (bottom panel). We take as an example the case of computing the **OT-GL** transport plan of digits 1 and 2 in USPS to those in MNIST. For this example, we have allowed a maximum of 50 iterations. Regarding convergence of the cost function along the iterations, we can clearly see that, while CGS reaches nearly-optimal objective value in around 10 iterations, the CG approach is still far from convergence after 50 iterations. In addition, the *per-iteration* cost is significantly higher for the conditional gradient algorithm. For this example, we save an order of magnitude of running time yet CG has not converged.

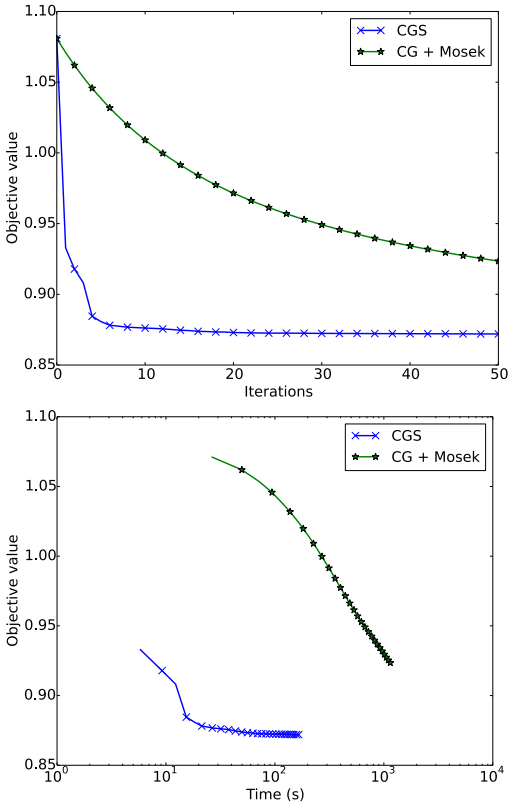


Fig. 5: Example of the evolution of the objective value along (top) the iterations and (bottom) the running times for the conditional gradient splitting and the conditional gradient algorithms.

We now study the computational performances of the different optimal transport strategies in the visual object recognition tasks considered above. We use Python implementations of the different OT methods. The test were run on a simple Macbook pro station, with a 2.4 Ghz processor. The original **OT-Exact** solution to the optimal transport problem is computed with the MOSEK [2] linear programming solver[2], whereas the other strategies follow our own implementation based on the Sinkhorn-Knopp method. For the regularized optimal transport **OT-GL** and **OT-Laplace** we used the conditional gradient splitting algorithm presented in Section 3.4 (the source code will be made available upon the acceptance of the article).

We report in Table 4 the computational time needed by the OT methods using the same configuration leading to the results in Table 3. As expected, **OT-IT** is the less computationally intensive of the methods. The solution of the exact optimal transport (**OT-exact**) is longer to compute by a factor 4. Also, as expected,

2. other publicly available solvers were considered, but it turned out this particular one was an order of magnitude faster than the others

TABLE 3: Overall recognition accuracies in % obtained over all possible pairs of domains. Maximum values for each pair is indicated in bold font.

	Methods									
	Standard Learning		Baseline methods				Optimal Transport Strategies			
	1NN	**PCA**	**GFK**	**TCA**	**TSL**	**JDA**	**OT-exact**	**OT-IT**	**OT-GL**	**OT-Laplace**
U→M	44.70	44.95	46.45	51.05	53.75	59.65	45.25	53.30	**61.00**	58.30
M→U	65.94	66.22	67.22	56.28	66.06	67.28	45.33	69.28	**70.28**	69.39
P1→P2	26.09	24.80	26.15	40.76	44.08	58.81	61.26	67.71	**71.76**	70.17
P1→P3	26.59	25.18	27.27	41.79	47.49	54.23	59.99	69.55	69.73	**69.85**
P1→P4	30.67	29.26	31.15	59.63	62.78	84.50	81.80	84.32	**84.68**	84.44
P1→P5	16.67	16.30	17.59	29.35	36.15	49.75	45.89	55.88	**56.19**	55.88
P2→P1	24.49	24.22	25.24	41.81	46.28	57.62	61.25	63.66	64.92	**65.46**
P2→P3	46.63	45.53	47.37	51.47	57.60	62.93	67.71	72.06	72.06	**72.12**
P2→P4	54.07	53.35	54.25	64.73	71.43	75.82	75.46	76.84	**77.89**	77.86
P2→P5	26.53	25.43	27.08	33.70	35.66	39.89	50.80	56.13	56.99	**57.72**
P3→P1	21.37	20.95	21.82	34.69	36.94	50.96	58.25	60.56	60.98	**61.46**
P3→P2	41.01	40.45	43.16	47.70	47.02	57.95	67.71	71.21	71.64	**72.99**
P3→P4	46.53	46.14	46.41	56.23	59.45	68.45	70.77	71.73	**72.00**	71.73
P3→P5	26.23	25.31	26.78	33.15	36.34	39.95	53.12	57.41	57.78	**58.64**
P4→P1	32.95	31.96	34.24	55.64	63.66	80.58	81.81	86.31	86.58	**86.97**
P4→P2	62.68	60.96	62.92	67.83	72.68	82.63	78.51	85.76	87.72	**87.97**
P4→P3	73.22	72.18	73.35	75.86	83.52	**87.25**	73.47	81.37	83.52	81.99
P4→P5	37.19	35.11	37.38	40.26	44.79	54.66	54.78	64.95	**69.36**	65.81
P5→P1	18.49	18.85	20.35	26.98	33.28	46.46	46.94	47.21	47.72	**47.84**
P5→P2	24.19	23.39	24.62	29.90	34.13	42.05	51.01	53.84	55.37	**55.49**
P5→P3	28.31	27.21	28.49	29.90	36.58	53.31	53.12	57.11	58.03	**58.33**
P5→P4	31.24	30.34	31.33	33.64	38.75	**57.01**	54.49	55.06	55.36	55.54
C→A	23.70	36.95	41.02	38.20	44.47	44.78	31.63	39.14	**48.02**	43.63
C→W	25.76	32.54	40.68	38.64	34.24	41.69	30.51	35.59	**44.75**	37.29
C→D	25.48	38.22	38.85	41.40	43.31	**45.22**	25.48	43.31	**45.22**	43.31
A→C	26.00	34.73	**40.25**	37.76	37.58	39.36	31.52	34.64	38.56	36.15
A→W	29.83	35.59	38.98	37.63	33.90	37.97	28.81	34.92	**41.36**	37.97
A→D	25.48	27.39	36.31	33.12	26.11	39.49	29.30	36.94	**44.59**	38.22
W→C	19.86	26.36	30.72	29.30	29.83	31.17	25.91	32.59	**37.22**	33.57
W→A	22.96	31.00	29.75	30.06	30.27	32.78	24.74	39.98	39.98	**40.29**
W→D	59.24	77.07	80.89	87.26	87.26	89.17	81.53	90.45	**92.99**	91.72
D→C	26.27	29.65	30.28	31.70	28.50	31.52	26.54	31.79	**33.84**	32.15
D→A	28.50	32.05	32.05	32.15	27.56	33.09	28.29	32.36	**36.74**	34.03
D→W	63.39	75.59	75.59	86.10	85.42	89.49	70.17	87.80	**90.51**	88.47
mean	34.77	37.22	39.30	44.87	47.55	55.51	52.15	58.85	61.33	60.08
p-value	3.7e-07	3.7e-07	4.4e-07	3.7e-07	5.4e-07	9.0e-06	3.7e-07	7.9e-07	-	1.6e-02

the two regularized versions **OT-GL** and **OT-Laplace** are the most demanding methods. We recall here that the maximum number of inner loop of the CGS approach was set to 10, meaning that each of those methods made 10 calls to the Sinkhorn-Knopp solver used by **OT-IT**. However, the added computational cost is mostly due to the line search procedure (line 4 in Algorithm 1), which involves several computations of the cost function. We explain the difference between **OT-GL** and **OT-Laplace** by the difference of computation time needed by this procedure. All in all, one can notice that even for large problems (case P1→P4 for instance, involving 3332×3329 variables), the computation time is not prohibitive and remains tractable.

5 Conclusion

In this paper, we proposed a general framework based on optimal transport to solve the unsupervised domain adaptation problem. Optimal transport plans are first estimated to find a matching transformation between the PDFs of the source and the target domains. Then, they can be used to provide a local transformation of the labeled source samples (each sample undergoes its own transport). We proposed two regularization schemes to encode class-structure in the source domain during the estimation of the transportation plan, thus enforcing the intuition that samples of the same class must undergo similar deformations. We also proposed to use a modified version of the conditional gradient algorithm, the conditional gradient splitting, that can benefit from the efficient Sinkhorn matrix scaling approach. With such a strategy, the method can scale up to process real-world datasets. Finally, we applied the proposed methods on both synthetic and real world datasets. Results show that the optimal transportation domain adaptation schemes always outperform the competing state-of-the-art methods. Moreover the optimal transport methods exploiting class-regularization obtained the best results.

We believe that the framework presented in this paper will lead to a paradigm shift for the domain

TABLE 4: Computational time (seconds) for the best set of regularization parameters

	OT-exact	OT-IT	OT-GL	OT-Laplace
U→M	86.0	4.6	92.5	55.6
M→U	85.0	2.3	75.4	20.5
P1→P2	131.0	30.8	432.7	333.2
P1→P3	133.9	27.9	456.6	296.2
P1→P4	319.1	85.4	1303.2	985.3
P1→P5	132.5	21.3	276.5	153.4
P2→P1	130.9	38.3	666.8	413.1
P2→P3	65.1	14.3	418.9	182.8
P2→P4	143.7	67.8	1054.0	730.4
P2→P5	67.8	11.7	270.1	120.0
P3→P1	135.7	36.4	519.0	389.0
P3→P2	67.6	14.3	297.8	156.6
P3→P4	140.8	79.5	1115.3	895.8
P3→P5	62.2	9.8	248.3	108.8
P4→P1	326.1	91.6	1142.6	935.8
P4→P2	145.3	23.8	557.9	451.7
P4→P3	148.0	24.5	844.1	916.5
P4→P5	137.3	27.4	406.5	290.7
P5→P1	134.6	22.9	540.4	272.6
P5→P2	66.7	11.1	262.5	123.0
P5→P3	65.3	12.6	269.6	127.5
P5→P4	134.6	30.2	492.7	329.0
C→A	26.9	1.2	22.5	17.8
C→W	6.8	0.3	6.6	7.4
C→D	3.4	0.2	3.5	2.4
A→C	26.5	1.1	29.4	23.8
A→W	5.6	0.3	6.0	6.0
A→D	2.9	0.2	3.2	4.1
W→C	6.8	0.3	16.2	7.6
W→A	5.7	0.3	4.1	7.4
W→D	0.8	0.1	2.2	1.1
D→C	3.6	0.2	14.7	5.9
D→A	3.0	0.2	12.5	5.1
D→W	0.8	0.1	3.8	1.1
mean	86.8	20.4	349.1	246.4

adaptation problem. Estimating a transport is much more general than finding a common subspace but it comes with the problem of finding a proper regularization. The proposed class-based or Laplacian regularizations show very good performances, but we believe that other types of regularization should be investigated. Indeed, as the transportation should correspond to the physical process that leads to the distribution shift, one may want the transport to enforce physical constraints, which can be included with dedicated regularization terms. Also, we plan to extend our optimal transport framework to the multi-domain adaptation problem, where the problem of matching several distributions can be cast as a multi-marginal optimal transport problem.

ACKNOWLEDGMENTS

REFERENCES

[1] "Impossibility theorems for domain adaptation." in *AISTATS*, 2010, pp. 129–136.

[2] E. Andersen and K. Andersen, "The mosek interior point optimizer for linear programming: An implementation of the homogeneous algorithm," in *High Performance Optimization*, ser. Applied Optimization. Springer US, 2000, vol. 33, pp. 197–232.

[3] S. Ben-David, J. Blitzer, K. Crammer, A. Kulesza, F. Pereira, and J. Vaughan, "A theory of learning from different domains," *Machine Learning*, vol. 79, no. 1-2, pp. 151–175, May 2010.

[4] S. Ben-david, J. Blitzer, K. Crammer, and O. Pereira, "Analysis of representations for domain adaptation," in *NIPS*. MIT Press, 2006.

[5] J.-D. Benamou and Y. Brenier, "A computational fluid mechanics solution to the monge-kantorovich mass transfer problem," *Numerische Mathematik*, vol. 84, no. 3, pp. 375–393, 2000. [Online]. Available: http://dx.doi.org/10.1007/s002110050002

[6] N. Bonneel, J. Rabin, G. Peyr'e, and H. Pfister, "Sliced and radon wasserstein barycenters of measures," *Journal of Mathematical Imaging and Vision*, no. to appear, to appear 2014.

[7] N. Bonneel, M. van de Panne, S. Paris, and W. Heidrich, "Displacement interpolation using lagrangian mass transport," *ACM Transaction on Graphics*, vol. 30, no. 6, pp. 158:1–158:12, Dec. 2011.

[8] L. Bruzzone and M. Marconcini, "Domain adaptation problems: A dasvm classification technique and a circular validation strategy," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 32, no. 5, pp. 770–787, May 2010.

[9] T. S. Caetano, T. Caelli, D. Schuurmans, and D. Barone, "Grapihcal models and point pattern matching," *IEEE Trans. Pattern Anal. Mach, Intell.*, vol. 28, no. 10, pp. 1646–1663, 2006.

[10] T. S. Caetano, J. J. McAuley, L. Cheng, Q. V. Le, and A. J. Smola, "Learning graph matching," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 31, no. 6, pp. 1048–1058, 2009.

[11] G. Carlier, A. Oberman, and E. Oudet, "Numerical methods for matching for teams and Wasserstein barycenters," Tech. Rep., 2014. [Online]. Available: http://hal.archives-ouvertes.fr/hal-00987292

[12] M. Carreira-Perpinan and W. Wang, "LASS: A simple assign-

ment model with laplacian smoothing," in *AAAI Conference on Artificial Intelligence*, 2014.

[13] N. Courty, R. Flamary, and D. Tuia, "Domain adaptation with regularized optimal transport," in *European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases (ECML PKDD)*, 2014.

[14] M. Cuturi, "Sinkhorn distances: Lightspeed computation of optimal transportation," in *NIPS*, 2013, pp. 2292–2300.

[15] M. Cuturi and A. Doucet, "Fast computation of wasserstein barycenters," in *ICML*, Sep. 2014.

[16] H. Daumé III, "Frustratingly easy domain adaptation," in *Ann. Meeting of the Assoc. Computational Linguistics*, 2007.

[17] S. Ferradans, N. Papadakis, J. Rabin, G. Peyré, and J.-F. Aujol, "Regularized discrete optimal transport," in *Scale Space and Variational Methods in Computer Vision, SSVM*, 2013, pp. 428–439.

[18] R. Flamary, N. Courty, and A. Rakotomamonjy, "Conditional gradient splitting," in *International Conference on Machine Learning (submitted)*, 2015.

[19] P. Germain, A. Habrard, F. Laviolette, and E. Morvant, "A PAC-Bayesian Approach for Domain Adaptation with Specialization to Linear Classifiers," in *ICML*, Atlanta, USA, 2013, pp. 738–746.

[20] B. Gong, Y. Shi, F. Sha, and K. Grauman, "Geodesic flow kernel for unsupervised domain adaptation." in *CVPR*. IEEE, 2012, pp. 2066–2073.

[21] R. Gopalan, R. Li, and R. Chellappa, "Domain adaptation for object recognition: An unsupervised approach," in *ICCV*. IEEE, 2011, pp. 999–1006.

[22] G. Griffin, A. Holub, and P. Perona, "Caltech-256 Object Category Dataset," California Institute of Technology, Tech. Rep. CNS-TR-2007-001, 2007.

[23] J. Ham, D. Lee, and L. Saul, "Semisupervised alignment of manifolds," in *10th International Workshop on Artificial Intelligence and Statistics*, R. G. Cowell and Z. Ghahramani, Eds., 2005, pp. 120–127.

[24] J. Hoffman, E. Rodner, J. Donahue, K. Saenko, and T. Darrell, "Efficient learning of domain invariant image representations," in *ICLR*, 2013.

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

[26] L. Kantorovich, "On the translocation of masses," *C.R. (Doklady) Acad. Sci. URSS (N.S.)*, vol. 37, pp. 199–201, 1942.

[27] P. Knight, "The sinkhorn-knopp algorithm: Convergence and applications," *SIAM J. Matrix Anal. Appl.*, vol. 30, no. 1, pp. 261–275, Mar. 2008.

[28] B. Kulis, K. Saenko, and T. Darrell, "What you saw is not what you get: domain adaptation using asymmetric kernel transforms," in *Proc. Computer Vision and Pattern Recognition (CVPR)*, Colorado Springs, CO, 2011.

[29] A. Kumar, H. Daumé III, and D. Jacobs, "Generalized multiview analysis: A discriminative latent space," in *CVPR*, 2012.

[30] M. Long, J. Wang, G. Ding, J. Sun, and P. Yu, "Transfer feature learning with joint distribution adaptation," in *ICCV*, Dec 2013, pp. 2200–2207.

[31] B. Luo and R. Hancock, "Structural graph matching using the em algorithm and singular value decomposition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 23, no. 10, pp. 1120–1136, 2001.

[32] Y. Mansour, M. Mohri, and A. Rostamizadeh, "Domain adaptation: Learning bounds and algorithms," in *COLT'09*, 2009, pp. 19–30.

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

[34] ——, "Domain adaptation via transfer component analysis," *IEEE Trans. Neural Networks*, vol. 22, pp. 199–210, 2011.

[35] V. M. Patel, R. Gopalan, R. Li, and R. Chellappa, "Visual domain adaptation: an overview of recent advances," *IEEE Trans. Pattern Anal. Mach. Intel.*, submitted. [Online]. Available: http://www.umiacs.umd.edu/~pvishalm/Journal_pub/SPM_DA_v7_embeded.pdf

[36] J. Rabin, G. Peyre, J. Delon, and M. Bernot, "Wasserstein barycenter and its application to texture mixing," in *Scale Space and Variational Methods in Computer Vision*, ser. Lecture Notes in Computer Science, 2012, vol. 6667, pp. 435–446.

[37] Y. Rubner, C. Tomasi, and L. Guibas, "A metric for distributions with applications to image databases," in *ICCV*, Jan 1998, pp. 59–66.

[38] K. Saenko, B. Kulis, M. Fritz, and T. Darrell, "Adapting visual category models to new domains," in *ECCV*, ser. LNCS, 2010, pp. 213–226.

[39] S. Si, D. Tao, and B. Geng, "Bregman divergence-based regularization for transfer subspace learning," *IEEE Trans. Knowledge Data Eng.*, vol. 22, no. 7, pp. 929–942, July 2010.

[40] J. Solomon, R. Rustamov, G. Leonidas, and A. Butscher, "Wasserstein propagation for semi-supervised learning," in *Proceedings of The 31st International Conference on Machine Learning*, 2014, pp. 306–314.

[41] M. Sugiyama, S. Nakajima, H. Kashima, P. Buenau, and M. Kawanabe, "Direct importance estimation with model selection and its application to covariate shift adaptation," in *NIPS*, 2008.

[42] C. Villani, *Optimal transport: old and new*, ser. Grundlehren der mathematischen Wissenschaften. Springer, 2009.

[43] C. Wang and S. Mahadevan, "Manifold alignment without correspondence," in *International Joint Conference on Artificial Intelligence*, Pasadena, CA, 2009.

[44] ——, "Heterogeneous domain adaptation using manifold alignment," in *IJCAI*. AAAI Press, 2011, pp. 1541–1546.

[45] K. Zhang, V. W. Zheng, Q. Wang, J. T. Kwok, Q. Yang, and I. Marsic, "Covariate shift in Hilbert space: A solution via surrogate kernels," in *ICML*, 2013.

[46] J. Zheng, M.-Y. Liu, R. Chellappa, and P. Phillips, "A grassmann manifold-based domain adaptation approach," in *ICPR*, Nov 2012, pp. 2095–2099.