On learning linearly alignable representations for domain adaptation

Anonymous Author(s)
Affiliation
Address
email

Abstract

Optimal transport (OT) is a powerful geometric tool used to compare and align probability measures following the least effort principle. Among many successful applications of OT in machine learning (ML), domain adaptation (DA) – a field of study where the goal is to transfer a classifier from one labelled domain to another similar, yet different unlabelled or scarcely labelled domain – has been historically among the most investigated ones. This success is due to the ability of OT to provide both a meaningful discrepancy measure to assess the similarity of two domains' distributions and a mapping that can project source domain data onto the target one. In this paper, we propose a principally new OT-based approach to DA that uses the closed-form solution of the OT problem given by an affine mapping and learns an embedding space for which this solution is optimal. We show that our approach works in both homogeneous and heterogeneous DA settings and outperforms or is on par with other famous baselines based on both traditional OT and OT in incomparable spaces.

"To design is to devise courses of action aimed at changing existing situations into preferred ones."
Herbert Simon, Nobel Prize winner, 1969.

1 Introduction

Optimal Transportation (OT) theory provides researchers with a large variety of tools to compare and align probability measures that are omnipresent in today's Machine Learning (ML) tasks. When the goal is to find a mapping for two continuous probability measures, one usually seeks to solve the original Monge OT formulation [1], while when one looks for soft-correspondences between the points in the supports of two empirical measures, Kantorovich formulation [2] of OT problem is usually considered. Due to its versatility, OT has recently become popular with its applications, spanning such diverse tasks and areas as unsupervised learning [3, 4], natural language processing [5, 6, 7], generative modelling [8, 9], computer vision [10, 11] and computational biology [12].

Limitations In practice, finding an optimal map or consistently estimating OT costs on real-world high-dimensional and large-scale data is hard, due to the curse of dimensionality of OT on the one hand [13, 14], and its high computational complexity on the other [15]. One popular approach to mitigate the curse of dimensionality is to consider adversarial lower-dimensional projections of the input measures [16, 17, 18] and solve OT on the projected measures. Another example is given by the famous sliced Wasserstein distances [19, 20], which leverage the closed-form solution of the OT problem in 1-dimensional space to calculate the OT cost through averaging over several such projections. These approaches, however, does not allow obtaining the mapping between the distributions, but only the OT cost. Another case of interest is the OT problem between Gaussian

Submitted to 36th Conference on Neural Information Processing Systems (NeurIPS 2022). Do not distribute.

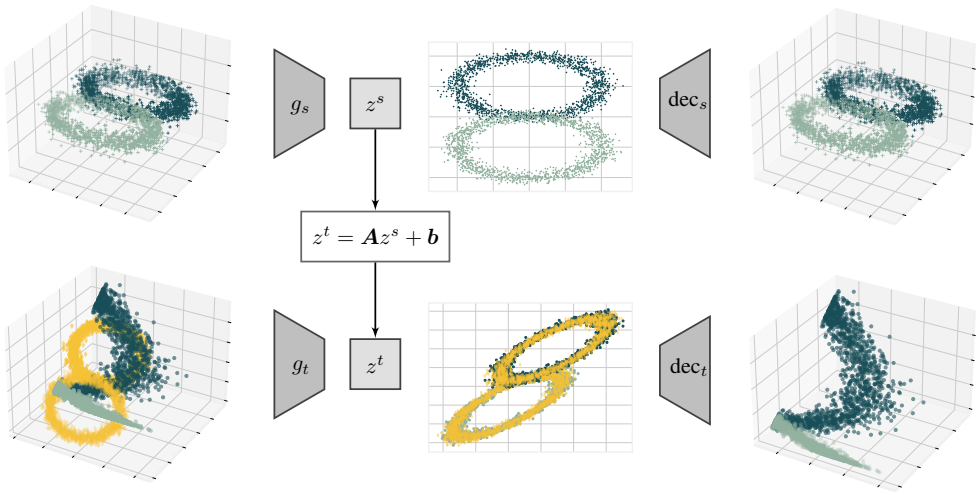

Figure 1: Illustration of the proposed approach for two datasets in $\mathbb{R}^3$. In the original 3D space, the projection obtained via linear Monge mapping (yellow points) between the two 3D datasets fails to align the datasets as the data in the original space neither follows a Gaussian distribution, nor it is linked through an affine transformation. Our approach learns an embedding space where the linear Monge map becomes optimal, while ensuring that the embeddings are discriminative for downstream tasks.

probability measures [21], and random variables linked through an affine transformation [22, 23], for which OT can be calculated in closed-form. However, as real-world data rarely corresponds to such favourable scenarios, this closed-form solution was only used scarcely in practice [24, 11].

Our contributions In this paper, we motivate our main proposal by the following question:

Can representation learning help to find an embedding space where the Monge mapping can be calculated explicitly for two discrete measures?

We answer this question positively and provide a new OT-based algorithm for DA having the following attractive properties:

1. We define a new framework of learning *linearly alignable representations* for DA that can be used to match the two domains' distributions embedded in a space where they become linked through an affine transformation. This is a generalization of the popular invariant representation learning [25] framework where the goal is find an invariant representation for both domains.
2. Once such representations are obtained, we use a closed-form linear Monge mapping that has a very appealing computational complexity and benefits from strong theoretical guarantees for DA. This is contrary to previous works on OT that either use neural networks to parametrize and approximate the Monge map between high-dimensional input distributions [26, 27] or use high-dimensional optimal couplings that do not scale with the increasing sample size [28, 29, 30, 31, 32].
3. Our learning framework covers both the case when the two domains' input spaces are the same (homogeneous DA) or different (heterogeneous DA). This is contrary to previous works on OT in DA that need to consider OT formulations on incomparable spaces to handle the heterogeneous DA setting.

The rest of the paper is organized as follows. Section 2 presents the necessary preliminary knowledge on DA and the use of OT in DA. Section 3 outlines our main contributions and provides a theoretical analysis for DA with linearly alignable representations. In Section 4, we evaluate our proposal on tasks for homogeneous unsupervised and heterogeneous semi-supervised DA where OT methods have previously shown to be efficient. We conclude this paper in Section 5.

2 Preliminary knowledge

Notations In what follows, we will use the following notations. We denote spaces and sets by black-board upper-case letters (e.g. $\mathbb{X}, \mathbb{Y}, \mathbb{R}$), probability measures are denoted by calligraphic upper-case letters (e.g. $\mathcal{S}, \mathcal{T}$), bold upper-case and lower-case Greek letters denote matrices (e.g. $\mathbf{X}, \boldsymbol{\gamma}$) and bold lower-case letters denote vectors (e.g. $\mathbf{x}, \mathbf{b}$). We denote the marginal distribution of $\mathcal{S}$ with respect to $\mathbb{X}$ by $\mathcal{S}_{\mathbb{X}}$ and denote by $\mathcal{P}(\mathbb{X})$ the space of probability measures supported on $\mathbb{X}$ with finite second moments.

Below, we present some background knowledge used in the following sections of this paper.

Domain adaptation Let $\mathbb{X}_S, \mathbb{X}_T$ be two subsets of $\mathbb{R}^d$ and $\mathbb{Y}$ be a discrete set of outputs. Given two datasets

$$\mathbf{S} = \{\mathbf{x}_i^s, y_i^s\}_{i=1}^{n_s} \sim \mathcal{S}(\mathbb{X}_S \times \mathbb{Y}), \text{ and } \mathbf{T} = \{\mathbf{x}_j^t, y_j^t\}_{j=1}^{n_t^l} \sim \mathcal{T}(\mathbb{X}_T \times \mathbb{Y}) \cup \{\mathbf{x}_i^t\}_{i=1}^{n_t^u} \sim \mathcal{T}_{\mathbb{X}},$$

the goal of domain adaptation (DA) [33, 34] is to learn a hypothesis function $h : \mathbb{X}_T \to \mathbb{Y}$ from some hypothesis class $\mathcal{H}$ using the data from $\mathbf{S}$ and $\mathbf{T}$ such that the true target risk $\mathrm{R}_{\mathcal{T}}(h) := \mathbb{E}_{\mathcal{T}}[\ell(h(\mathbf{x}^t), y^t)]$ is as small as possible for some loss function $\ell : \mathbb{Y} \times \mathbb{Y} \to \mathbb{R}$. In what follows, we distinguish between unsupervised DA, ie, $n_t^l = 0$ and, semi-supervised DA, ie, $0 < n_t^l \ll n_t^u$. We also deploy the term **heterogeneous** when considering a setup where $\mathbb{X}_S \neq \mathbb{X}_T$.

The vast majority of algorithms solving DA follow the theoretical foundation laid out in the seminal works on DA theory [35] (surveyed in [36]). This latter can be summarized by the following learning bound:

$$\mathrm{R}_{\mathcal{T}}(h) \leq \mathrm{R}_{\mathcal{S}}(h) + \mathrm{div}(\mathcal{S}_{\mathbb{X}}, \mathcal{T}_{\mathbb{X}}) + \min_{h \in \mathcal{H}}(\mathrm{R}_{\mathcal{T}}(h) + \mathrm{R}_{\mathcal{S}}(h)), \quad \text{for } h \in \mathcal{H}, \tag{1}$$

where $\mathrm{div}(\cdot, \cdot)$ is some divergence or distance on the space of probability measures. Equation (1) suggests the idea of learning an *invariant feature transformation* [25] function $g : \mathbb{X}_S \cup \mathbb{X}_T \to \mathbb{Z}$ such that $\mathrm{div}(\mathcal{S}_{\mathbb{X}}^g, \mathcal{T}_{\mathbb{X}}^g) = 0$ for the distributions $\mathcal{S}_{\mathbb{X}}^g, \mathcal{T}_{\mathbb{X}}^g$ induced by g while ensuring that $\mathrm{R}_{\mathcal{S}}(h \circ g)$ is as small as possible. One should note that, in general, g can also be applied to one of the domains only such that $\mathrm{div}(\mathcal{S}_{\mathbb{X}}^g, \mathcal{T}_{\mathbb{X}}) = 0$. This approach is often referred to as *asymmetric* feature transformation.

As finding a way to minimize $\mathrm{R}_{\mathcal{S}}(h \circ g)$ presents a common well-studied supervised learning problem, the main challenge of solving DA was thus to find a meaningful measure of divergence $\mathrm{div}(\cdot, \cdot)$ and a learning strategy to find the desired g minimizing it. Below, we discuss how optimal transportation (OT) theory has been recently used to achieve this.

Optimal transport OT and its associated metrics have become a popular choice to find g in order to solve both homogeneous [28, 37, 29, 30, 38, 39, 40, 27] and heterogeneous DA [31, 32]. For the former setting, an assymetric feature transformation function $g : \mathbb{X}_S \to \mathbb{X}_T$ can be obtained as a solution to the Monge problem defined for two metric spaces $\mathbb{X}_S$, $\mathbb{X}_T$, and a cost function $c : \mathbb{X}_S \times \mathbb{X}_T \to \mathbb{R}$ as follows:

$$g \in \operatorname*{argmin}_{g : g_{\#}\mathcal{S}_{\mathbb{X}} = \mathcal{T}_{\mathbb{X}}} \mathbb{E}_{\mathbf{x}^s \sim \mathcal{S}_{\mathbb{X}}}[c(\mathbf{x}^s, g(\mathbf{x}^s))]. \tag{2}$$

Here $g_{\#}\mathcal{S}_{\mathbb{X}}$ denotes the push-forward measure, which is equivalent to the law of $g(\mathbf{x}^s)$, for $\mathbf{x}^s \sim \mathcal{S}_{\mathbb{X}}$. Unfortunately, solving (2) is very hard in practice as its constraints are non-convex and the solutions for it may not exist in discrete case when $\mathcal{S}_{\mathbb{X}}$ and $\mathcal{T}_{\mathbb{X}}$ are empirical measures.

A more widely adapted approach is to consider instead the Monge-Kantorovich problem [2] and the Wasserstein distance associated to it. The latter is defined as a value at the solution of the former as follows:

$$W_c(\mathcal{S}_{\mathbb{X}}, \mathcal{T}_{\mathbb{X}}) = \min_{\gamma \in \Pi(\mathcal{S}_{\mathbb{X}}, \mathcal{T}_{\mathbb{X}})} \mathbb{E}_{\gamma} c(\mathbf{x}^s, \mathbf{x}^t), \tag{3}$$

where $\Pi(\mathcal{S}_{\mathbb{X}}, \mathcal{T}_{\mathbb{X}})$ is the space of probability distributions over $\mathbb{X}_S \times \mathbb{X}_T$ with marginals $\mathcal{S}_{\mathbb{X}}$ and $\mathcal{T}_{\mathbb{X}}$. When the squared Euclidean cost function $c(\cdot, \cdot) = ||\cdot - \cdot||_2^2$ is used, we write simply W_2^2. Once γ is obtained, one usually uses the so-called *barycentric mapping*[41] to define g as follows:

$$g : \mathbf{x}^s \to \operatorname*{argmin}_{\mathbf{x}} \mathbb{E}_{\gamma(\cdot|\mathbf{x}^t)} c(\mathbf{x}, \mathbf{x}^t). \tag{4}$$

3 Proposed contributions

3.1 Motivation

Previous OT-based DA methods have several important drawbacks. On one hand, the methods based on deriving the barycentric mapping from the high-dimensional optimal coupling, such as [28, 29, 30, 31], are unsuitable for large-scale applications as shown in [26]. On the other hand, DA methods based on Monge mapping estimation [26, 27] rely on parametrizing the Monge mapping with neural networks that may fail to converge to the true solution [42].

In this section, we present a method that relies on a closed-form solution of the Monge problem in the particular case of random variables linked through an affine transformation. As for real-world data used in DA the relationship between the random variables following source and target distributions is unlikely to be linear, we first present the framework of learning *linearly alignable representations* that generalizes the idea of learning invariant feature transformations for DA [25] to learning feature transformations that are invariant modulo an affine transformation. In practice, we propose to achieve this by embedding the data into a space where the affine transformation between the source and target samples becomes nearly optimal. We now proceed by defining this idea more formally.

3.2 Linearly alignable representations

We propose to use representation learning, and, more particularly, generative modeling, to find a new data representation for which source and target distributions are linearly alignable. Of these, the latter can be formally defined based as follows.

Definition 3.1. *Given two distributions $\mathcal{S}_{\mathbb{X}} \in \mathcal{P}(\mathbb{X}_S)$ and $\mathcal{T}_{\mathbb{X}} \in \mathcal{P}(\mathbb{X}_T)$, the feature transformation functions $g_s : \mathbb{X}_S \to \mathbb{Z}_S$, $g_t : \mathbb{X}_T \to \mathbb{Z}_T$ are called linearly alignable (LA) for $\mathcal{S}_{\mathbb{X}}$ and $\mathcal{T}_{\mathbb{X}}$ if $\exists T : \mathbf{z} \to \mathbf{A}\mathbf{z} + \mathbf{b}$ with an invertible matrix $\mathbf{A}$ and a translation vector $\mathbf{b}$ such that $T_{\#}\mathcal{S}_{\mathbb{X}}^{g_s} = \mathcal{T}_{\mathbb{X}}^{g_t}$.*

One should note that this definition generalizes the invariant feature transformation learning, as the latter is a special case with $\mathbf{A} = \text{Id}, \mathbf{b} = \mathbf{0}$ and $g_s = g_t = g$. This implies, in particular, that the space of solutions to the problem of finding an invariant feature transformation function is included into that of finding linearly alignable feature transformation functions. Consequently, it may be easier to solve the latter problem as it allows for more degrees of freedom.

Given Definition 3.1, learning LA representations thus boils down to identifying two major ingredients: 1) the alignability criterion forcing (g_s, g_t) to provide LA representations for samples drawn from two distributions; 2) the data fidelity term forcing (g_s, g_t) to truthfully reflect the statistical distribution of the input samples in the embedding space. We discuss our choices for both these ingredients below.

Linear Monge mapping When $\mathcal{S}_{\mathbb{X}}$ and $\mathcal{T}_{\mathbb{X}}$ are linked through an affine transformation T with a positive definite matrix $\mathbf{A}$, the OT problem admits a simple solution that can be calculated based on the Gaussian approximations $\mathcal{N}(\mathbf{m}_S, \mathbf{\Sigma}_S)$ and $\mathcal{N}(\mathbf{m}_T, \mathbf{\Sigma}_T)$ of $\mathcal{S}_{\mathbb{X}}$ and $\mathcal{T}_{\mathbb{X}}$ [22, 23]. In particular, we have that for two such distributions, the Wasserstein distance between $\mathcal{S}_{\mathbb{X}}$ and $\mathcal{T}_{\mathbb{X}}$ admits a closed-form expression for the quadratic cost Wasserstein distance:

$$W_2^2(\mathcal{S}_{\mathbb{X}}, \mathcal{T}_{\mathbb{X}}) = ||\mathbf{m}_S - \mathbf{m}_T||_2^2 + \text{tr}(\mathbf{\Sigma}_S) + \text{tr}(\mathbf{\Sigma}_T) - 2\text{tr}(\mathbf{\Sigma}_T^{\frac{1}{2}}\mathbf{\Sigma}_S\mathbf{\Sigma}_T^{\frac{1}{2}})^{\frac{1}{2}}$$

and the optimal transport map T_{aff} of the corresponding Monge problem is given by:

$$\begin{aligned} &T_{\text{aff}}^{[\mathcal{S}_{\mathbb{X}}, \mathcal{T}_{\mathbb{X}}]}(\mathbf{x}) = \mathbf{A}\mathbf{x} + \mathbf{b}, \\ &\mathbf{A} = \mathbf{\Sigma}_T^{\frac{1}{2}}(\mathbf{\Sigma}_T^{\frac{1}{2}}\mathbf{\Sigma}_S\mathbf{\Sigma}_T^{\frac{1}{2}})^{-\frac{1}{2}}\mathbf{\Sigma}_T^{\frac{1}{2}}, \qquad \mathbf{b} = \mathbf{m}_T - \mathbf{A}\mathbf{m}_S. \end{aligned} \tag{5}$$

When dealing with empirical measures $\widehat{\mathcal{S}}_{\mathbb{X}}$ and $\widehat{\mathcal{T}}_{\mathbb{X}}$, $\mathbf{\Sigma}_S$, $\mathbf{\Sigma}_T$, $\mathbf{m}_S$ and $\mathbf{m}_T$ are replaced with their empirical (biased) counterparts defined from available finite samples from the supports of the two distributions. In the sequel, we denote those with a hat as well, ie, $\widehat{\mathbf{A}}$ is defined in terms of the empirical covariance matrices $\widehat{\mathbf{\Sigma}}_S$, $\widehat{\mathbf{\Sigma}}_T$ and empirical means $\widehat{\mathbf{m}}_S$, $\widehat{\mathbf{m}}_T$.

Based on this, we propose to define the alignability for two distributions $\mathcal{S}_{\mathbb{X}}$ and $\mathcal{T}_{\mathbb{X}}$ as the Wasserstein distance between the push-forward of $\mathcal{S}_{\mathbb{X}}$ with $T^{[\mathcal{S}_{\mathbb{X}}, \mathcal{T}_{\mathbb{X}}]}$ and $\mathcal{T}_{\mathbb{X}}$, ie,

$$\mathcal{L}_{\text{LA}}(\mathcal{S}_{\mathbb{X}}, \mathcal{T}_{\mathbb{X}}) := W_2^2(T_{\text{aff}}^{[\mathcal{S}_{\mathbb{X}}, \mathcal{T}_{\mathbb{X}}]}{}_{\#}\mathcal{S}_{\mathbb{X}}, \mathcal{T}_{\mathbb{X}}),$$

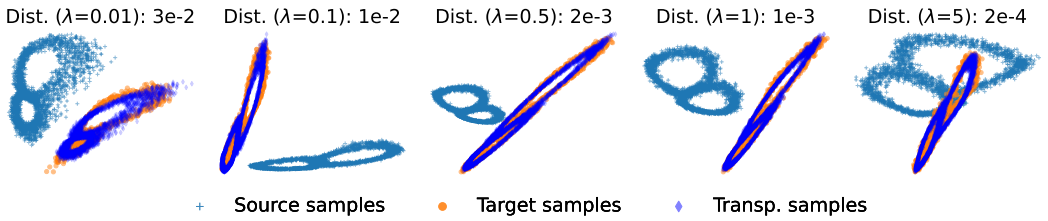

Figure 2: Illustration of the embeddings (light blue and orange points) and linear Monge mapping projection (blue points) obtained by our approach for different values of λ for $g_s, g_t : \mathbb{R}^{20} \to \mathbb{R}^2$. We can see that the linear Monge mapping becomes more and more optimal in the embedding space, as confirmed by the Wasserstein distance after the projection that reduces when λ increases.

where T_{aff} is defined as in (5). The intuition behind this is that when this distance is close to 0, the linear Monge mapping T_{aff} becomes the optimal mapping between the two distributions implying that they become linearly alignable with T_{aff}.

Data fidelity To preserve the information contained in the samples drawn from $\mathcal{S}_{\mathbb{X}}$ and $\mathcal{T}_{\mathbb{X}}$ when making them linearly alignable, we propose to model $g_s : \mathbb{X}_S \to \mathbb{Z}_S$ and $g_t : \mathbb{X}_T \to \mathbb{Z}_T$ as encoders of two different auto-encoders with the same size of the embedding space k, ie, $\mathbb{Z}_S, \mathbb{Z}_T \subseteq \mathbb{R}^k$. More formally, we have the following reconstruction term:

$$\mathcal{L}_{\text{Rec.}}(\mathcal{S}_{\mathbb{X}}, \mathcal{T}_{\mathbb{X}}) := \mathbb{E}_{\mathbf{x}^s \sim \mathcal{S}_{\mathbb{X}}} ||\mathbf{x}^s - (g_s \circ \text{dec}_s)\mathbf{x}^s||_2^2 + \mathbb{E}_{\mathbf{x}^t \sim \mathcal{T}_{\mathbb{X}}} ||\mathbf{x}^t - (g_t \circ \text{dec}_t)\mathbf{x}^t||_2^2, \tag{6}$$

where the decoders $\text{dec}_s : \mathbb{Z}_S \to \mathbb{X}_S, \text{dec}_t : \mathbb{Z}_T \to \mathbb{X}_T$ seek to reconstruct the learned embeddings by mapping them back into the original space.

Using two separate auto-encoders with two feature transformation functions brings two benefits. First, it allows to deal with the heterogeneous DA by initializing the input layers of the used auto-encoders with different widths; second, it adds more expressiveness allowing to learn richer individual representations for samples from two different domains and to adjust the complexity of the used architecture depending on the quality of the input data accordingly.

Optimization problem Putting all the ingredients together, we propose to optimize the following objective function:

$$\min_{g_s, g_t, \text{dec}_S, \text{dec}_T} \mathcal{L}_{\text{Rec.}}(\mathcal{S}_{\mathbb{X}}, \mathcal{T}_{\mathbb{X}}) + \lambda \mathcal{L}_{\text{LA}}(\mathcal{S}_{\mathbb{X}}^{g_s}, \mathcal{T}_{\mathbb{X}}^{g_t}), \tag{7}$$

where λ is a hyper-parameter controlling the degree to which the linear alignability is promoted as illustrated in Figure 2. In a nutshell, (7) seeks to embed the data from two distributions supported on potentially different metric spaces into two representation spaces for which there exists an affine map – given by the linear Monge map – that aligns them. This idea is illustrated in Figure 1.

Complexity analysis As noted in [22], the sample complexity of linear Monge mapping estimation is dimension-free and addresses the curse of dimensionality of solving the original OT problem. Given two samples of the same size n from $\mathbb{R}^d$, the latter is known to have a sample complexity of $\mathcal{O}(n^{-\frac{1}{d}})$, while the former is $\mathcal{O}(n^{-\frac{1}{2}})$(Theorem 1, [22]). Similarly, the computational complexity of calculating the linear Monge map is $\mathcal{O}(nd^2 + d^3)$ which is particularly attractive for large-scale applications due to its linearity in n. The dependence on dimensionality is alleviated by the fact that we estimate it in the embedding space of dimensionality $k \ll d$.

Lifting to the input space Minimizing (7) allows to obtain new low-dimensional embeddings of the input measures for which the linear Monge mapping is optimal. One may wonder, however, whether it is possible to lift the obtained mapping back to the original space. This question was studied in [43] where the authors showed how a Monge mapping that is optimal on a subspace can be used to define an optimal mapping, or a coupling, in the original space as well. In the particular case of our work that uses closed-form Monge mapping, [43] shows that it can be used to define an optimal coupling in a closed-form based on the subspace optimal solution. Unfortunately, g_s and g_t are not subspace projectors in our case, meaning that identifying whether the linear Monge mapping is optimal on the input measures is much harder. We leave this idea for future investigation.

3.3 Theoretical guarantees for domain adaptation

The simplicity of the proposed approach, and the closed-form expression of the Monge mapping in the embedding space, allow us to rely on theoretical guarantees for the performance of a classifier transferred from $\mathcal{S}_{\mathbb{X}}^{g_s}$ to $\mathcal{T}_{\mathbb{X}}^{g_t}$ via $T_{\text{aff}}[\mathcal{S}_{\mathbb{X}}^{g_s}, \mathcal{T}_{\mathbb{X}}^{g_t}]$. Before introducing these guarantees, we recall the definition of the Lipschitz function used in the statements.

Definition 3.2. *A function $h : \mathbb{X} \to \mathbb{Y}$ is called M-Lipschitz if $||h(\mathbf{x}) - h(\mathbf{x}')|| \leq M||\mathbf{x} - \mathbf{x}'||$ for all $\mathbf{x}, \mathbf{x}' \in \mathbb{X}$.*

We now present our main theoretical results for the DA task and postpone all the proofs of this paper to the Supplementary materials.

Theorem 3.1. *(Best-case bound) Let $h \in \mathcal{H}$ be M_h-Lpschitz and the loss function ℓ be M_ℓ-Lipschitz in its second argument. Then, if there exists a mapping m such that $m_{\#}\mathcal{S}^{g_s} = \mathcal{T}^{g_t}, m(\mathbf{z}^s, y^s) = m(T_{aff}^{[\mathcal{S}_{\mathbb{X}}^{g_s}, \mathcal{T}_{\mathbb{X}}^{g_t}]}(\mathbf{z}^s), y^t)$ and linearly alignable feature transformation functions g_s and g_t for $\mathcal{S}_{\mathbb{X}}$ and $\mathcal{T}_{\mathbb{X}}$, we have that*

$$R_{\mathcal{T}^{g_t}}\left(h \circ (T_{aff}^{[\mathcal{S}_{\mathbb{X}}^{g_s}, \mathcal{T}_{\mathbb{X}}^{g_t}]})^{-1}\right) \leq R_{\mathcal{S}^{g_s}}(h) + M_h M_\ell ||\hat{\mathbf{A}}^{-1}|| O\left(\max(n_s, n_t)^{-\frac{1}{2}}\right), \tag{8}$$

As mentioned in Section 2, previous works on DA theory introduced the learning bounds on the target error following the general shape of (1). For instance, in [44, 37] the obtained bounds corresponded exactly to (1) with $\text{div}(\mathcal{S}_{\mathbb{X}}, \mathcal{T}_{\mathbb{X}}) = W_{||\cdot||_1}(\mathcal{S}_{\mathbb{X}}, \mathcal{T}_{\mathbb{X}})$ while in [29] a similar bound was obtained with $W_{||\cdot||_1}(\mathcal{S}, \mathcal{T})$ where $\mathcal{T}$ was defined with pseudo-labels. In the case of linear Monge mapping, however, the learning bound on the target error becomes much simpler and does not involve any additional terms under the introduced assumptions. Furthermore, it can be improved using [22, Theorem 2] where under some additional assumptions, one can show that the true target error of the hypothesis calculated from the available source data, ie, $h^* \in \underset{h \in \mathcal{H}}{\text{argmin}}\, \widehat{\text{R}}_{\mathcal{S}^{g_s}}(h)$ converges to the optimal target classifier $h_t^* = \underset{h \in \mathcal{H}}{\text{argmin}}\, \text{R}_{\mathcal{T}^{g_t}}(h)$, even despite the absence of labelled data in the target domain. This remarkable result thus motivates our framework of learning linearly alignable representations as it provably transposes the problem of DA to a much more favourable setting.

To complete this section, we also present a more general learning bound close in spirit to that given in (1). For this result, we do not assume the existence of a mapping m that allows to remove the ideal joint error term $\min_h(\text{R}_{\mathcal{T}^{g_s}}(h) + \text{R}_{\mathcal{S}^{g_s}}(h))$, and do not assume that our feature transformation functions are linearly alignable. We only assume that the linear Monge mapping is used to align the two distributions in the embedding space.

Theorem 3.2. *(Worst case bound) Let $h \in \mathcal{H}$ be M_h-Lipschitz. Denote by $T[\mathcal{S}_{\mathbb{X}}^{g_s}] := T_{aff}^{[\mathcal{S}_{\mathbb{X}}^{g_s}, \mathcal{T}_{\mathbb{X}}^{g_t}]}{}_{\#}\mathcal{S}_{\mathbb{X}}^{g_s}$ and let $f_S : \mathbb{Z}_S \to \mathbb{Y}$ and $f_T : \mathbb{Z}_T \to \mathbb{Y}$ be the true labelling function associated to $T[\mathcal{S}_{\mathbb{X}}^{g_s}]$ and $\mathcal{T}_{\mathbb{X}}^{g_t}$, respectively. Then, for two arbitrary feature transformation functions g_s and g_t, we have that*

$$R_{\mathcal{T}_{\mathbb{X}}^{g_t}}(h, f_t) \leq R_{T[\mathcal{S}_{\mathbb{X}}^{g_s}]}(h, f_s) + 2\sqrt{2}M_h tr(\Sigma_{\mathcal{T}_{\mathbb{X}}^{g_t}})^{\frac{1}{2}} + \min_{h \in \mathcal{H}} R_{\mathcal{T}_{\mathbb{X}}^{g_t}}(h, f_t) + R_{T[\mathcal{S}_{\mathbb{X}}^{g_s}]}(h, f_s). \tag{9}$$

This result is the worst-case scenario for our proposed framework as it bounds the Wasserstein distance between $T[\mathcal{S}_{\mathbb{X}}^{g_s}]$ and $\mathcal{T}_{\mathbb{X}}^{g_t}$ by its largest possible value given by $\text{tr}(\Sigma_{\mathcal{T}_{\mathbb{X}}^{g_t}})^{\frac{1}{2}}$. As in practice our learning algorithm solves a non-convex optimization problem and can, in principle, converge to approximatively linearly alignable feature transformations g_s and g_t, this result suggests controlling the variance of the target embedded features to avoid having a target latent space $\mathbb{Z}_T$ that is too spread along all k directions in the embedding space.

3.4 Related works

Our work is situated at the cross-roads of computational OT and transfer learning. Below, we review the related approaches and point out their differences with respect to our proposal.

Monge mapping estimation Estimating the OT map from finite samples drawn from two probability distributions is a very active research topic nowadays. The vast majority of such methods

(see Table 1 in [42] and references therein) parametrize the Monge mapping, or the potential that defines it following Brenier theorem [45], using either a traditional or an input convex neural network [46]. Our work is principally different from this line of research in two main aspects. First, these contributions use the high expressive power of NNs and ICNNs to solve the hard problem of finding a mapping between two continuous high-dimensional measures. Our work instead uses the power of representation learning to find a new space where the problem of mapping two distributions becomes easy. As such, neural OT methods and our proposal solve different problems and cannot be used interchangeably. Finally, [47] approximates the barycentric mapping from (4) using a linear mapping either in the original or in the similarity-induced space. Contrary to it, we use a closed-form expression of the true Monge mapping that is optimal in the embedding space and scales better as it never explicitly calculates the high-dimensional coupling used by the latter method.

Subspace learning for OT Our approach is related to OT methods that use a projection of the data to a low-dimensional subspace [19, 48, 43, 49] to accelerate OT computation. In [19] (and follow-up works [50, 51]), the authors propose sliced Wasserstein distance computed as an average of the Wasserstein distances over one-dimensional projections of the high-dimensional distributions where the Wasserstein distance can be calculated in closed-form. Sliced Wasserstein distances are commonly used as a way to compute the approximate OT cost faster, for instance in generative modelling [20], yet they do not provide a meaningful mapping between the considered distributions. In [48], the authors embed the data into a new space where the Euclidean distance between the embedded samples corresponds to the Wasserstein distance between the empirical measures supported on these samples in the original space. The purpose of their method is thus different as it aims primarily to accelerate the OT computation. [43] is much closer in spirit to what we propose: their idea is to extend the Monge map that is optimal on the low-dimensional subspace to be optimal on the full space. This idea is further extended to Gromov-Wasserstein distance in [49]. Our approach learns a new representation, rather than finding a subspace of the original space, for which the optimal Monge map is easy to compute and does not seek to lift it to the original space.

OT in DA We now briefly discuss other OT-based DA works here. [28] is the seminal work that proposed to use OT in DA. The authors solve (3) with entropic and class-based regularizations and then use (4) to project source data to the target domain. This method was further extended to the alignment of joint probability distributions in [29] and its deep version [38]. Another line of work on OT in DA is concerned with target shift [30] and generalized target shift [40, 27] where $\mathcal{S} \neq \mathcal{T}$ due to $\mathcal{S}_{\mathbb{Y}} \neq \mathcal{T}_{\mathbb{Y}}$ for target shift and $\mathcal{S}(\mathbb{X}|y) \neq \mathcal{T}(\mathbb{X}|y)$ in addition to it for generalized target shift. Several methods also follow the invariant feature transformation framework such as [37, 39]. Finally, [31, 32] tackle the heterogeneous DA setup using Gromov-Wasserstein [52] and Co-Optimal transport problem problems in [32]. Our work is different from all these methods as it relies on closed-form Monge mapping and allows to unify both heterogeneous and homogeneous DA setups in one approach. Additionally, its simplicity also allows us to benefit from stronger theoretical guarantees in the embedding space that are unavailable for other existing methods. For a general survey on DA, we refer to [34, 53, 54].

4 Experimental evaluations

In this section, we evaluate our method, termed **LaOT** (Linearly Alignable Optimal Transport) against other OT-based methods for commonly considered unsupervised homogeneous (UDA) and semi-supervised heterogeneous DA (HDA) tasks. For both evaluations we use Office/Caltech10 dataset [55] that consists of 4 different domains, namely: Amazon (A) (958 images), Caltech (C) (1123 images), Webcam (W) (295 images) and DSLR (D) (157 images) from 10 overlapping classes. Given a pair "Source $\rightarrow$ Target", for both settings the final goal is to learn a classifier using only the available labelled data in the source domain to further evaluate it in the target domain. We now present in more detail the evaluation setup for each of the two settings considered below.

Implementation details We use fully connected NNs with 1 hidden layer for $g_s, g_t, \text{dec}_s, \text{dec}_t$ with ReLU activation function. In all experiments, the size of the hidden layer is fixed to half of the size of the input layer. The classifier used for UDA is a fully connected NN with softmax function applied to the output. For HDA, none of the considered baselines learns a classifier simultaneously to solving the OT problem so that in this case we minimize (7) without any additional terms. The optimization

is carried out using Adam optimizer [56] in PyTorch [57] with gradient normalization and default initialization of the weights. We also use POT library [58] to minimize W_2^2. The code, as well as the visualizations of the learned embeddings and several ablation studies are provided as part of the Supplementary material.

Model selection As suggested in [30], we use reverse validation [59] with 3NN classifier for our method in order to choose the best hyperparameters that include the size of the embedding space $k \in [64, 128, 256]$, regularization strength $\alpha \in [0.1, 0.05, 0.01]$, batch size $\in [32, 64, 128]$ and learning rate $\in [5e-5, 1e-4, 5e-4]$. We perform 10 runs of 10 epochs for each set of hyperparameters and pick the model having the lowest variance of the reverse validation score. We also report the best model chosen by reverse validation, i.e. without using target labels unavailable during learning, over the runs. This latter metric is common for deep DA methods [37] as the considered datasets are rather small and may lead to a model converging to bad local minima.

4.1 Homogeneous unsupervised DA

Setup For this evaluation, we constitute 12 pairs of adaptation tasks for the 4 domains available and use the weights of the fully connected 6th layer of the DECAF convolutional neural network [60] pre-trained on ImageNet as their features. This leads to an adaptation problem between sparse 4096 dimensional vectors. Following [29], we use cross-validated SVC classifier with linear kernel [61] for all methods. We compare our proposal against famous OT-based approaches used in DA, namely: entropy-regularized (OT-IT) and class-wise regularized OT (OT-MM) (both from [62]) that adds a group-lasso penalty on the coupling matrix that doesn't allow source points of different classes to be transported to the same target point. Finally, we also add Joint Distribution Optimal Transportation (JDOT) [30] method to our comparison that uses OT to align joint probability distributions and learns a classifier for pseudo-labelled target data simultaneously. All these baselines are evaluated against the source classifier directly applied in the target domain (Base). Additionally, and to show that our method compares favourably to deep DA methods, we follow the evaluation protocol of [37] and compare the best achieved performance (using target labels) of our method against three other deep-based baselines, namely: domain adversarial neural networks (DANN) [63], Deep Correlation Alignment (CORAL) [64] and Wasserstein-guided Representation learning (WGRL) [37].

Tasks	Base	OT-IT	OT-MM	JDOT	LaOT
A→C	84.77	85.93	**87.36**	85.22	<u>86.02</u> (84.93±0.77)
A→D	86.62	77.71	79.62	87.90	**92.36** (**88.85**±2.55)
A→W	79.32	74.24	85.08	84.75	**96.95** (**92.33**±2.83)
C→A	<u>92.07</u>	89.98	**92.59**	91.54	**92.59** (90.73±1.01)
C→D	84.08	78.34	76.43	89.81	**93.63** (<u>89.87</u>±1.55)
C→W	76.27	80.34	78.98	<u>88.81</u>	**93.90** (88.07±2.27)
D→A	83.19	**90.50**	**90.50**	88.10	<u>89.87</u> (86.96±0.6)
D→C	77.03	**85.57**	83.35	<u>84.33</u>	79.52 (76.5±0.87)
D→W	<u>96.27</u>	**96.61**	**96.61**	**96.61**	95.93 (94.07±1.07)
W→A	79.44	89.56	90.50	<u>90.71</u>	**93.42** (90.16±0.74)
W→C	71.77	**84.06**	82.99	82.64	<u>83.26</u> (75.57±1.52)
W→D	96.18	**99.36**	**99.36**	<u>98.09</u>	97.45 (95.92±2.24)
p-value	<0.05	0.2	0.33	0.62	–

Table 1: Classification results for UDA task. Bold and underlined scores present the best and the second best results. Baseline results reported from [29].

Results The obtained results are presented in Tables 1-5. From the comparison with both shallow OT-based methods and deep DA methods, we can see that LaOT is statistically on par with them according to Wilcoxon signed-rank test calculated with respect to the best model. This performance is achieved despite the simplicity of our method, that similarly to CORAL and OT-IT, doesn't rely on adversarial training [63, 37], on structural constraints on the coupling matrix (OT-MM) or pseudo-labeling and joint distribution adaptation (JDOT).

	DANN	CORAL	WGRL	LaOT
Mean	87.67±6.78	90.76±4.39	<u>92.74</u>±3.52	**93.82**±5.55
p-value	0.09	0.2	0.33	–

Table 2: Average best accuracy for UDA against deep-based DA methods. Complete results are presented in the Supplementary materials.

4.2 Heterogeneous semi-supervised DA

In this experiment, we evaluate LaOT on the same dataset but with source and target feature representations given by activations from GoogleNet [65] and Decaf [60] neural network architectures.

Tasks	Base	SGW	COOT_{LP}	COOT	LaOT
A→A	83.04±3.07	89.75±4.8	**92.89**±0.32	89.74±0.01	<u>91.86</u> (91±0.91)
A→C	69.98±2.88	79.80±5.82	**86.76**±1.28	<u>83.76</u>±2.02	81.12 (80.07±1.77)
A→W	80.49±3.96	93.76±2.06	**96.61**(±1.34)	94.44±2.23	<u>95.59</u> (92.92±1.47)
C→A	83.09±2.94	78.37±5.08	67.28±1.02	89.66±1.23	<u>89.35</u> (88.51±1.4)
C→C	68.46±3.13	81.31±5.09	67.28±1.19	81.95±1.79	**82.72** (79.82±1.78)
C→W	81.66±4.62	90.81±3.36	69.39±2.01	90.92±1.85	**91.53** (88.34±2.34)
W→A	84.59±3.4	82.63±11.12	72.33±1.19	84.75±1.57	**91.34** (88.92±2.44)
W→C	67.60±4.63	75.25±6.13	63.51±0.78	77.3±3.7	**81.75** (76.08±3.11)
W→W	82.83±3.42	94.00±1.13	77.49±2.6	**95.42**±1.39	<u>94.24</u> (93.28±2.65)
p-value	<1e-2	<0.05	0.05	0.73	–

Table 3: Classification results for semi-supervised HDA task. Bold and underlined scores present the best and the second best results.

In OT context, aligning two such heterogeneous datasets is alleviated by using OT in incomparable spaces: first such contribution relies on the Gromov-Wasserstein distance [31] (SGW), while a more recent method improving upon this latter used its generalization termed Co-Optimal Transport [32] (COOT). We follow the protocol of [32] where only the domains A, C and W were considered. To help guiding adaptation in this case, previous works commonly consider the semi-supervised setting with a handful of labelled examples in the target domain. In this evaluation, we set the number of such examples to 3 per class, ie, $n_t^l = 30$. For all baselines, we use the hyper-parameters suggested by authors in the respective papers. As our method aligns datasets using a Monge mapping and not the coupling matrix used in [32] to perform label propagation [66], we present the results of SGW and our method with 3NN classifier, and use label propagation results for COOT only.

Results From Table 3, we can see that our method is statistically better than SGW and COOT with label propagation and is on par with COOT followed by 3NN classifier. As in the homogeneous setting, our method uses a very simple closed-form expression in the embedding space, contrary to simultaneous sample and feature alignment of COOT and pair-wise matrices' alignment with conditional distribution matching of SGW. This further supports our claim about the fact that representation learning can help to alleviate the intrinsic complexity of aligning high-dimensional probability measures by finding embeddings making the OT problem easier to solve.

5 Discussions and future work

In this paper, we proposed a novel contribution at the crossroads of computational OT and transfer learning. On one hand, we proposed a learning framework that embeds the data from two distributions to a new representation space where we can explicitly calculate the Monge mapping between their induced distributions. On the other hand, we showed how this learning framework, termed learning linearly alignable representations, can be used in both homogeneous and heterogeneous domain adaptation with strong theoretical guarantees and high competitive performance. Our work is a first contribution that aims at exploiting one of the simplest solutions to the Monge mapping estimation problem in general k-dimensional spaces. In this work we concentrated on only one application of our general approach, mainly to showcase how its simplicity can bring both theoretical and empirical advantages in transfer learning. Our proposal, however, can be used in many other ML problems where Monge mapping is already used such as in, for instance, GANs, were the use of sliced Wasserstein distance is known to reduce significantly the computational burden related to their training.

Limitations Our work exploits previously overlooked linear Monge mapping to perform both UDA and HDA. Just as with the invariant feature transformation setting, our method is also subject to impossibility theorems [25] stating that DA can fail even when the source and target distributions are perfectly aligned and the source error is minimized. In addition to this, our method does not benefit from "subspace detours" guarantees that can justify their optimality in the original space as mentioned in Section 3.

References

[1] Gaspard Monge. Mémoire sur la théorie des déblais et des remblais. *Histoire de l'Académie Royale des Sciences*, pages 666–704, 1781.

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

[3] Charlotte Laclau, Ievgen Redko, Basarab Matei, Younès Bennani, and Vincent Brault. Co-clustering through optimal transport. In *ICML*, pages 1955–1964, 2017.

[4] Antoine Rolet, Marco Cuturi, and Gabriel Peyré. Fast dictionary learning with a smoothed wasserstein loss. In *AISTATS*, pages 630–638, 2016.

[5] David Alvarez-Melis and Tommi S. Jaakkola. Gromov-Wasserstein Alignment of Word Embedding Spaces. In *EMNLP*, pages 1881–1890, 2018.

[6] Matt Kusner, Yu Sun, Nicholas Kolkin, and Kilian Weinberger. From word embeddings to document distances. In *Proceedings of the 32nd International Conference on Machine Learning*, pages 957–966, 2015.

[7] Sidak Pal Singh, Andreas Hug, Aymeric Dieuleveut, and Martin Jaggi. Context mover's distance and barycenters: Optimal transport of contexts for building representations. In *Proceedings of the Twenty Third International Conference on Artificial Intelligence and Statistics*, pages 3437–3449, 2020.

[8] Martin Arjovsky, Soumith Chintala, and Léon Bottou. Wasserstein generative adversarial networks. In *ICML*, pages 214–223, 2017.

[9] Charlotte Bunne, David Alvarez-Melis, Andreas Krause, and Stefanie Jegelka. Learning Generative Models across Incomparable Spaces. In *ICML*, pages 851–861, 2019.

[10] Nicholas I. Kolkin, Jason Salavon, and Gregory Shakhnarovich. Style transfer by relaxed optimal transport and self-similarity. In *CVPR*, pages 10051–10060, 2019.

[11] Youssef Mroueh. Wasserstein style transfer. In *AISTATS*, pages 842–852, 2020.

[12] Pinar Demetci, Rebecca Santorella, Björn Sandstede, William Stafford Noble, and Ritambhara Singh. Gromov-wasserstein optimal transport to align single-cell multi-omics data. *bioRxiv*, 2020.

[13] Nicolas Fournier and Arnaud Guillin. On the rate of convergence in wasserstein distance of the empirical measure, 2013.

[14] Jonathan Weed and Francis Bach. Sharp asymptotic and finite-sample rates of convergence of empirical measures in wasserstein distance, 2017.

[15] Gabriel Peyré and Marco Cuturi. Computational optimal transport. *Foundations and Trends® in Machine Learning*, 11:355–607, 2019.

[16] François-Pierre Paty and Marco Cuturi. Subspace robust wasserstein distances. In *ICML*, pages 5072–5081, 2019.

[17] Sofien Dhouib, Ievgen Redko, Tanguy Kerdoncuff, Rémi Emonet, and Marc Sebban. A swiss army knife for minimax optimal transport. In *ICML*, pages 2504–2513, 2020.

[18] Mokhtar Z. Alaya, Maxime Bérar, Gilles Gasso, and Alain Rakotomamonjy. Theoretical guarantees for bridging metric measure embedding and optimal transport. *Neurocomputing*, 468:416–430, 2022.

[19] Nicolas Bonneel, Julien Rabin, Gabriel Peyré, and Hanspeter Pfister. Sliced and radon wasserstein barycenters of measures. *J. Math. Imaging Vis.*, 51(1):22–45, 2015.

[20] Ishan Deshpande, Ziyu Zhang, and Alexander G. Schwing. Generative modeling using the sliced wasserstein distance. In *CVPR*, pages 3483–3491, 2018.

[21] D. C. Dowson and B. V. Landau. The fréchet distance between multivariate normal distributions. *Journal of Multivariate Analysis*, 12(3):450–455, 1982.

[22] Rémi Flamary, Karim Lounici, and André Ferrari. Concentration bounds for linear monge mapping estimation and optimal transport domain adaptation. *CoRR*, abs/1905.10155, 2019.

[23] Anton Mallasto, Karol Arndt, Markus Heinonen, Samuel Kaski, and Ville Kyrki. Affine transport for sim-to-real domain adaptation. *CoRR*, abs/2105.11739, 2021.

[24] François Pitié and Anil C. Kokaram. The linear monge-kantorovitch linear colour mapping for example-based colour transfer. In *IEEE European Conference on Visual Media Production*, 2007.

[25] Han Zhao, Remi Tachet Des Combes, Kun Zhang, and Geoffrey Gordon. On learning invariant representations for domain adaptation. In *ICML*, pages 7523–7532, 2019.

[26] Vivien. Seguy, Bharath B. Damodaran, Remi Flamary, Nicolas Courty, Antoine Rolet, and Mathieu Blondel. Large-scale optimal transport and mapping estimation. In *International Conference on Learning Representations (ICLR)*, 2018.

[27] Matthieu Kirchmeyer, Alain Rakotomamonjy, Emmanuel de Bezenac, and patrick gallinari. Mapping conditional distributions for domain adaptation under generalized target shift. In *International Conference on Learning Representations*, 2022.

[28] N. Courty, R. Flamary, and D. Tuia. Domain adaptation with regularized optimal transport. In *ECML PKDD*, pages 1–16, 2014.

[29] Nicolas Courty, Rémi Flamary, Amaury Habrard, and Alain Rakotomamonjy. Joint distribution optimal transportation for domain adaptation. In *NIPS*, pages 3730–3739, 2017.

[30] Ievgen Redko, Nicolas Courty, Rémi Flamary, and Devis Tuia. Optimal transport for multi-source domain adaptation under target shift. In *AISTATS*, pages 849–858, 2019.

[31] Yuguang Yan, Wen Li, Hanrui Wu, Huaqing Min, Mingkui Tan, and Qingyao Wu. Semi-supervised optimal transport for heterogeneous domain adaptation. In *IJCAI*, pages 2969–2975, 2018.

[32] Ivegen Redko, Titouan Vayer, Rémi Flamary, and Nicolas Courty. Co-optimal transport. In *Neural Information Processing Systems (NeurIPS)*, 2020.

[33] S. J. Pan and Q. Yang. A Survey on Transfer Learning. *IEEE Transactions on Knowledge and Data Engineering*, 22(10):1345–1359, 2010.

[34] Karl Weiss, Taghi M. Khoshgoftaar, and DingDing Wang. A survey of transfer learning. *Journal of Big Data*, 3(1):9, 2016.

[35] Shai Ben-David, John Blitzer, Koby Crammer, Alex Kulesza, Fernando Pereira, and Jennifer Vaughan. A theory of learning from different domains. *Machine Learning*, 79:151–175, 2010.

[36] Ievgen Redko, Emilie Morvant, Amaury Habrard, Marc Sebban, and Younès Bennani. *Advances in Domain Adaptation Theory*. Elsevier, 2019.

[37] Jian Shen, Yanru Qu, Weinan Zhang, and Yong Yu. Wasserstein distance guided representation learning for domain adaptation. In Sheila A. McIlraith and Kilian Q. Weinberger, editors, *AAAI-18*, pages 4058–4065, 2018.

[38] Bharath B. Damodaran, Benjamin Kellenberger, Rémi Flamary, Devis Tuia, and Nicolas Courty. Deepjdot: Deep joint distribution optimal transport for unsupervised domain adaptation. In *European Conference in Computer Visions (ECCV)*, 2018.

[39] Renjun Xu, Pelen Liu, Liyan Wang, Chao Chen, and Jindong Wang. Reliable weighted optimal transport for unsupervised domain adaptation. In *CVPR*, pages 4393–4402, 2020.

[40] Alain Rakotomamonjy, Rémi Flamary, Gilles Gasso, Mokhtar Z Alaya, Maxime Berar, and Nicolas Courty. Optimal transport for conditional domain matching and label shift. *Machine Learning*, 2021.

[41] Sira Ferradans, Nicolas Papadakis, Julien Rabin, Gabriel Peyré, and Jean-François Aujol. Regularized discrete optimal transport. In Arjan Kuijper, Kristian Bredies, Thomas Pock, and Horst Bischof, editors, *Scale Space and Variational Methods in Computer Vision*, pages 428–439, 2013.

[42] Alexander Korotin, Lingxiao Li, Aude Genevay, Justin Solomon, Alexander Filippov, and Evgeny Burnaev. Do neural optimal transport solvers work? A continuous wasserstein-2 benchmark. 2021.

[43] Boris Muzellec and Marco Cuturi. Subspace detours: Building transport plans that are optimal on subspace projections. In *NeurIPS*, pages 6914–6925, 2019.

[44] Ievgen Redko, Amaury Habrard, and Marc Sebban. Theoretical analysis of domain adaptation with optimal transport. In *Proceedings of the European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases (ECML/PKDD)*, pages 737–753, 2017.

[45] Yann Brenier. Polar factorization and monotone rearrangement of vector-valued functions. (4):375–417, 1991.

[46] Brandon Amos, Lei Xu, and J. Zico Kolter. Input convex neural networks. In *ICML*, page 146–155, 2017.

[47] Michaël Perrot, Nicolas Courty, Rémi Flamary, and Amaury Habrard. Mapping estimation for discrete optimal transport. In *NIPS*, pages 4197–4205, 2016.

[48] Nicolas Courty, Rémi Flamary, and Mélanie Ducoffe. Learning wasserstein embeddings. In *ICLR*, 2018.

[49] Clément Bonet, Titouan Vayer, Nicolas Courty, François Septier, and Lucas Drumetz. Subspace detours meet gromov-wasserstein. *Algorithms*, 14(12):366, 2021.

[50] Soheil Kolouri, Kimia Nadjahi, Umut Simsekli, Roland Badeau, and Gustavo K Rohde. Generalized sliced wasserstein distances. *arXiv preprint arXiv:1902.00434*, 2019.

[51] Ishan Deshpande, Yuan-Ting Hu, Ruoyu Sun, Ayis Pyrros, Nasir Siddiqui, Sanmi Koyejo, Zhizhen Zhao, David Forsyth, and Alexander G Schwing. Max-sliced wasserstein distance and its use for gans. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10648–10656, 2019.

[52] Facundo Memoli. Gromov wasserstein distances and the metric approach to object matching. *Foundations of Computational Mathematics*, pages 1–71, 2011.

[53] Garrett Wilson and Diane J. Cook. A Survey of Unsupervised Deep Domain Adaptation. *arXiv:1812.02849 [cs, stat]*, 2019.

[54] Lei Zhang. Transfer Adaptation Learning: A Decade Survey. *arXiv:1903.04687 [cs]*, 2019.

[55] K. Saenko, B. Kulis, M. Fritz, and T. Darrell. Adapting visual category models to new domains. In *ECCV*, LNCS, pages 213–226, 2010.

[56] Diederik P. Kingma and Max Welling. Auto-Encoding Variational Bayes. In *ICLR*, 2014.

[57] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems 32*, pages 8024–8035. 2019.

[58] Rémi Flamary, Nicolas Courty, Alexandre Gramfort, Mokhtar Z. Alaya, Aurélie Boisbunon, Stanislas Chambon, Laetitia Chapel, Adrien Corenflos, Kilian Fatras, Nemo Fournier, Léo Gautheron, Nathalie T.H. Gayraud, Hicham Janati, Alain Rakotomamonjy, Ievgen Redko, Antoine Rolet, Antony Schutz, Vivien Seguy, Danica J. Sutherland, Romain Tavenard, Alexander Tong, and Titouan Vayer. Pot: Python optimal transport. *Journal of Machine Learning Research*, 22(78):1–8, 2021.

[59] Erheng Zhong, Wei Fan, Qiang Yang, Olivier Verscheure, and Jiangtao Ren. Cross validation framework to choose amongst models and datasets for transfer learning. In *Machine Learning and Knowledge Discovery in Databases*, pages 547–562, 2010.

[60] J. Donahue, Y. Jia, O. Vinyals, J. Hoffman, N. Zhang, E. Tzeng, and T. Darrell. Decaf: A deep convolutional activation feature for generic visual recognition. In *ICML*, 2014.

[61] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

[62] Nicolas Courty, Rémi Flamary, Devis Tuia, and Alain Rakotomamonjy. Optimal transport for domain adaptation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39(9):1853–1865, 2017.

[63] Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, Mario Marchand, and Victor Lempitsky. Domain-adversarial training of neural networks. *J. Mach. Learn. Res.*, 17(1):2096–2030, 2016.

[64] Baochen Sun and Kate Saenko. Deep coral: Correlation alignment for deep domain adaptation. In *ECCV Workshops*, 2016.

[65] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. In *CVPR*, pages 1–9, 2015.

[66] I. Redko, N. Courty, R. Flamary, and D. Tuia. Optimal transport for multi-source domain adaptation under target shift. In *International Conference on Artificial Intelligence and Statistics (AISTAT)*, 2019.

[67] Laurens van der Maaten and Geoffrey Hinton. Visualizing data using t-SNE. *Journal of Machine Learning Research*, 9:2579–2605, 2008.

Checklist

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes]
   (b) Did you describe the limitations of your work? [Yes] See Section 5
   (c) Did you discuss any potential negative societal impacts of your work? [N/A] We do not see any overreaching impacts of our work.
   (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes]
2. If you are including theoretical results...
   (a) Did you state the full set of assumptions of all theoretical results? [Yes]
   (b) Did you include complete proofs of all theoretical results? [Yes] We provide proof sketches and postpone full proofs to the Supplementary materials.
3. If you ran experiments...
   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [Yes] We provide all the details and provide the code for reproducibility.

(b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [Yes] Partly in text, partly in the Supplementary materials.

(c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [Yes] For our methods we report the results averaged over random seeds. We do not report them for Table 1 as the original results didn't contain this information.

(d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [No]

4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...

(a) If your work uses existing assets, did you cite the creators? [Yes] See Section 4.

(b) Did you mention the license of the assets? [N/A]

(c) Did you include any new assets either in the supplemental material or as a URL? [Yes] We provide our code.

(d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [N/A] All data is publicly available.

(e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [N/A]

5. If you used crowdsourcing or conducted research with human subjects...

(a) Did you include the full text of instructions given to participants and screenshots, if applicable? [N/A]

(b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A]

(c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A]

A Appendix

A.1 Proofs of theorems

Full proof of Theorem 3.1

Proof. From the definition of linearly alignable feature transformation functions, we deduce that $\exists T$ such that $T_{\#}\mathcal{S}_{\mathbb{X}}^{g_s} = \mathcal{T}_{\mathbb{X}}^{g_t}$. Given the assumption about the existence of mapping m, we have that for any $h \in \mathcal{H}$, $\mathrm{R}_{\mathcal{S}^{g_s}}(h) = \mathrm{R}_{\mathcal{T}^{g_t}}\left(h \circ T^{-1}\right)$. We then use [22, Proposition 1] to obtain the desired result by replacing the original source and target distributions with their embedded counterparts. □

Full proof of Theorem 3.2

Proof. Let $h^* \in \operatorname*{argmin}_h \mathrm{R}_{\mathcal{T}_{\mathbb{X}}^{g_t}}(h, f_t) + \mathrm{R}_{T[\mathcal{S}_{\mathbb{X}}^{g_s}]}(h, f_s)$. Then, we have that:

$$
\begin{aligned}
\mathrm{R}_{\mathcal{T}_{\mathbb{X}}^{g_t}}(h, f_t) &\leq \mathrm{R}_{\mathcal{T}_{\mathbb{X}}^{g_t}}(h, h^*) + \mathrm{R}_{\mathcal{T}_{\mathbb{X}}^{g_t}}(h^*, f_t) \\
&\leq \mathrm{R}_{\mathcal{T}_{\mathbb{X}}^{g_t}}(h, h^*) + \mathrm{R}_{\mathcal{T}_{\mathbb{X}}^{g_t}}(h^*, f_t) + \mathrm{R}_{T[\mathcal{S}_{\mathbb{X}}^{g_s}]}(h, h^*) - \mathrm{R}_{T[\mathcal{S}_{\mathbb{X}}^{g_s}]}(h, h^*) \\
&\leq \mathrm{R}_{\mathcal{T}_{\mathbb{X}}^{g_t}}(h^*, f_t) + \mathrm{R}_{T[\mathcal{S}_{\mathbb{X}}^{g_s}]}(h, h^*) + 2M_h W_1(T[\mathcal{S}_{\mathbb{X}}^{g_s}], \mathcal{T}_{\mathbb{X}}^{g_t}) \\
&\leq \mathrm{R}_{\mathcal{T}_{\mathbb{X}}^{g_t}}(h^*, f_t) + \mathrm{R}_{T[\mathcal{S}_{\mathbb{X}}^{g_s}]}(h, f_s) + \mathrm{R}_{T[\mathcal{S}_{\mathbb{X}}^{g_s}]}(h^*, f_s) + 2M_h W_1(T[\mathcal{S}_{\mathbb{X}}^{g_s}], \mathcal{T}_{\mathbb{X}}^{g_t}) \\
&= \mathrm{R}_{\mathcal{T}_{\mathbb{X}}^{g_t}}(h^*, f_t) + 2M_h W_1(T[\mathcal{S}_{\mathbb{X}}^{g_s}], \mathcal{T}_{\mathbb{X}}^{g_t}) + \min_{h \in \mathcal{H}} \mathrm{R}_{\mathcal{T}_{\mathbb{X}}^{g_t}}(h, f_t) + \mathrm{R}_{T[\mathcal{S}_{\mathbb{X}}^{g_s}]}(h, f_s) \\
&\leq \mathrm{R}_{\mathcal{T}_{\mathbb{X}}^{g_t}}(h^*, f_t) + 2M_h W_2(T[\mathcal{S}_{\mathbb{X}}^{g_s}], \mathcal{T}_{\mathbb{X}}^{g_t}) + \min_{h \in \mathcal{H}} \mathrm{R}_{\mathcal{T}_{\mathbb{X}}^{g_t}}(h, f_t) + \mathrm{R}_{T[\mathcal{S}_{\mathbb{X}}^{g_s}]}(h, f_s) \\
&\leq \mathrm{R}_{T[\mathcal{S}_{\mathbb{X}}^{g_s}]}(h, f_s) + 2\sqrt{2} M_h \mathrm{tr}(\Sigma_{\mathcal{T}_{\mathbb{X}}^{g_t}})^{\frac{1}{2}} + \min_{h \in \mathcal{H}} \mathrm{R}_{\mathcal{T}_{\mathbb{X}}^{g_t}}(h, f_t) + \mathrm{R}_{T[\mathcal{S}_{\mathbb{X}}^{g_s}]}(h, f_s).
\end{aligned}
$$

The proof follows the common reasoning used to obtain DA learning bounds with the Wasserstein distance [44, 37]. Line 3 is obtained using Lemma 1 from [37], Line 5 is due to the Jensen inequality implying for all $0 < p < q$, that $W_p \leq W_q$. It is then completed by an upper-bound on the Wasserstein distance between $T[\mathcal{S}_{\mathbb{X}}^{g_s}]$ and $\mathcal{T}_{\mathbb{X}}^{g_t}$ that was bounded in [23] by $\mathrm{tr}(\Sigma_{\mathcal{T}_{\mathbb{X}}^{g_t}})^{\frac{1}{2}}$. □

A.2 Comparison of LaoT with linear Monge mapping on raw data

In Table 4, we present an abalation study showing how promoting linear alignability affects the performance on DA task compared to applying linear Monge mapping on raw data directly (OT-Gauss). We can see that apart from two DA tasks, OT-Gauss method is always far below LaOT and even of the base classifier.

Tasks	**Base**	**OT-Gauss**	**LaOT**
A→C	<u>84.77</u>	83.35	**86.02** (**84.93**±0.77)
A→D	<u>86.62</u>	83.44	**92.36** (**88.85**±2.55)
A→W	79.32	<u>81.36</u>	**96.95** (**92.33**±2.83)
C→A	<u>92.07</u>	89.56	**92.59** (90.73±1.01)
C→D	<u>84.08</u>	82.17	**93.63** (**89.87**±1.55)
C→W	76.27	<u>81.69</u>	**93.90** (**88.07**±2.27)
D→A	<u>83.19</u>	82.67	**89.87** (**86.96**±0.6)
D→C	77.03	<u>78.45</u>	**79.52** (76.5±0.87)
D→W	<u>96.27</u>	**97.63**	95.93 (94.07±1.07)
W→A	79.44	<u>84.13</u>	**93.42** (**90.16**±0.74)
W→C	71.77	<u>76.22</u>	**83.26** (75.57±1.52)
W→D	96.18	**1**	<u>97.45</u> (95.92±2.24)

Table 4: Classification results for UDA task comparing LaOT and linear Monge mapping on the raw data (OT-Gauss). Bold and underlined scores present the best and the second best results. Baseline results reported from [29].

A.3 Full comparison with deep UDA methods

Below, we provide full results for all pairs of Office/Caltech dataset corresponding to the average results in Table . We can see that our method remains efficient even when compared to stronger baselines given by adversarial DA methods.

Tasks	**DANN**	**DeepCORAL**	**WGRL**	**LaOT**
A→C	**87.80**	86.18	86.99	<u>87.62</u>
A→D	82.46	91.23	<u>93.68</u>	**98.09**
A→W	77.81	<u>90.53</u>	89.47	**99.32**
C→A	93.27	93.01	**93.54**	<u>93.53</u>
C→D	91.23	89.47	<u>94.74</u>	**96.18**
C→W	89.47	<u>92.63</u>	91.58	**97.97**
D→A	84.70	85.75	<u>91.69</u>	**92.07**
D→C	82.11	<u>85.37</u>	**90.24**	83.17
D→W	**98.95**	97.89	97.89	<u>98.64</u>
W→A	82.98	88.39	<u>93.67</u>	**94.47**
W→C	81.30	<u>88.62</u>	**89.43**	84.77
W→D	**100**	**100**	**100**	**100**

Table 5: Best accuracy results for UDA against deep-based DA methods. Baseline results are reported from [37].

A.4 Illustration of the trade-off between data fidelity and linear alignability

In Figure 3, we present the results obtained by best performing LaOT models when varying the λ parameter in $[0, 0.01, 0.05, 0.1, 0.5, 1]$. The value of $\lambda = 0$ correspond to the case when only data fidelity loss is minimized and no alignment is forced between the two embeddings. As can be seen from this result, this leads to a drastic loss in terms of accuracy, while other values of λ lead to approximately the same results.

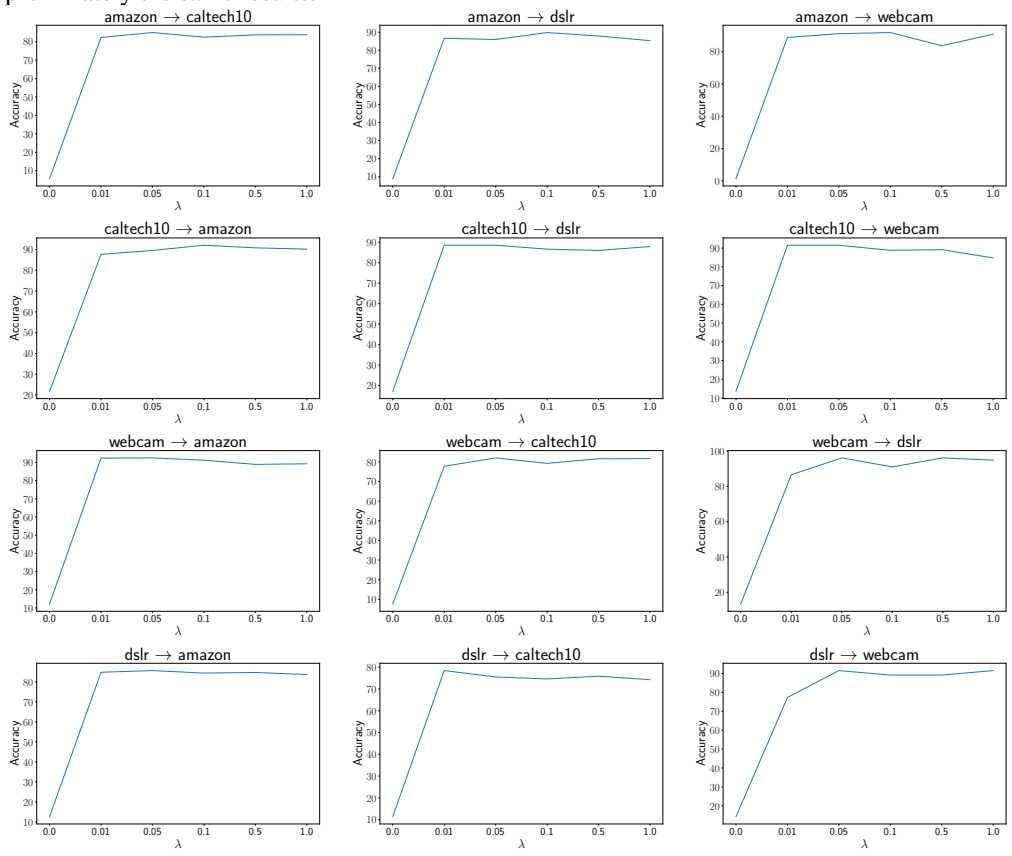

Figure 3: Trade-off between linear alignability loss and data fidelity loss for optimal LaOT models achieving highest UDA performance.

A.5 Illustration of learned embeddings

In Figure 4, we provide plots of embeddings obtained using tSNE [67] learned for UDA task with LaOT. We can see that LaOT does not explicitly align two domains but has an extra degree of flexibility allowing it to learn potentially richer representations.

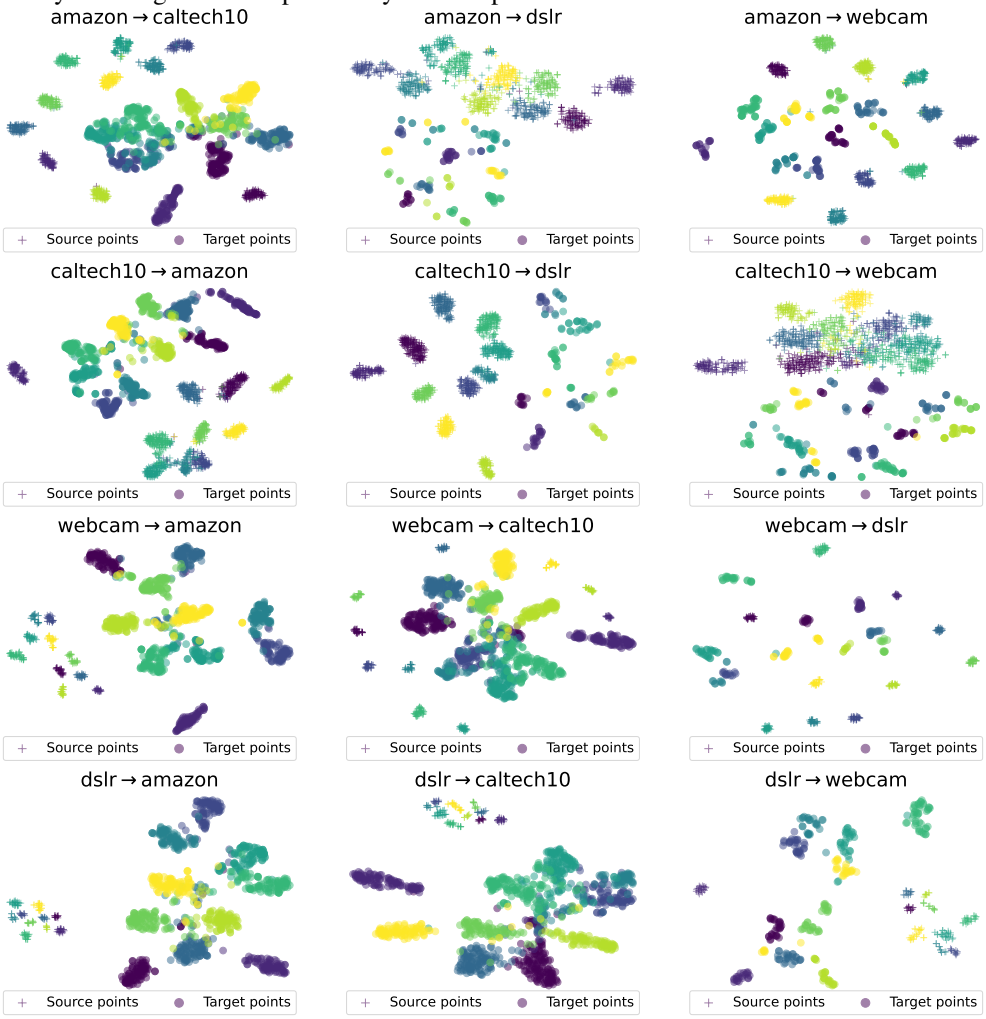

Figure 4: Visualizations of embeddings for different UDA tasks.

A.6 Illustration of learning dynamics

In Figure 5 we provide illustration for learning dynamics of our method on UDA tasks. From this, we can see that the accuracy of the linear classifier increases when the distance after the projection with the linear Monge map in the embedding space decreases. This is in line with what we expect from the minimization of our objective function.

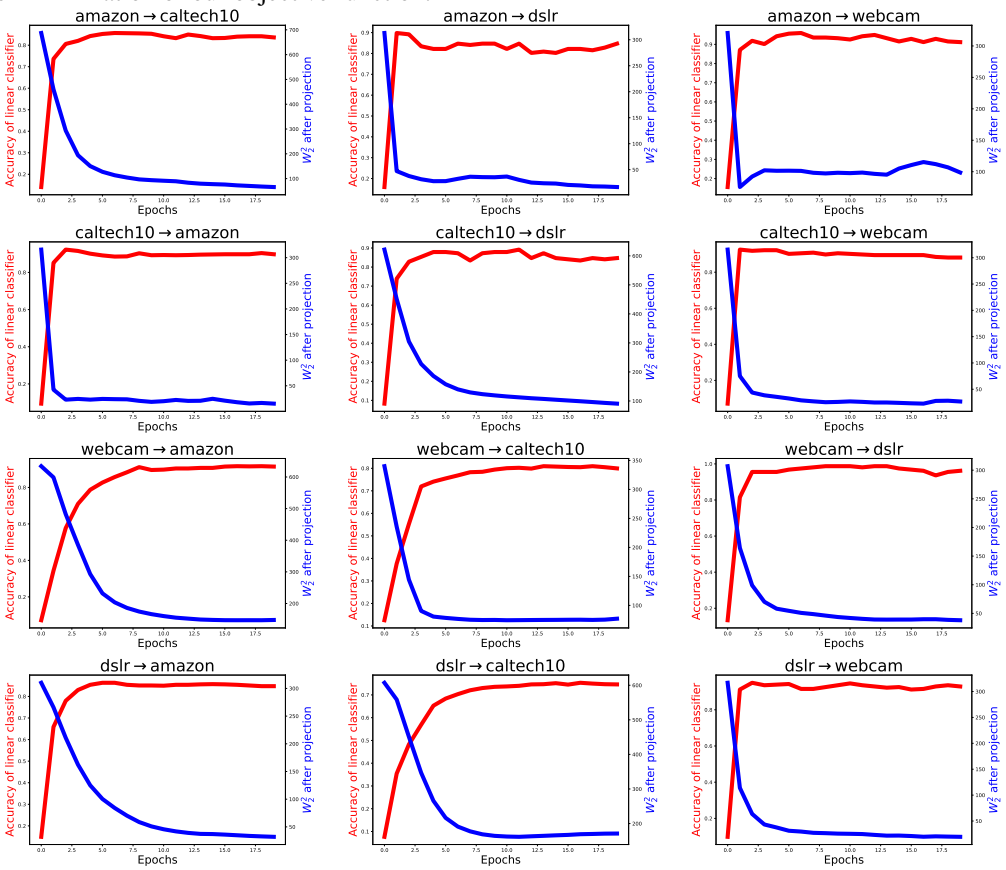

Figure 5: Learning dynamics of our method on UDA tasks.

A.7 Comparison with invariant feature transformation learning

Finally, we compare our approach against invariant feature transformation learning where the source and target data are explicitly forced to be close in the embedding space. For this, we simply set $T_{\text{aff}}(\mathbf{x}) = \mathbf{A}\mathbf{x} + \mathbf{b}$ in (7) and optimize it as before. For the sake of clarity, we take the task D→W to illustrate both the learned embeddings and the learning dynamics of LaOT and the invariant feature transformation approach.

These results are presented in Figure 6. From this plot, we distinctly see that LaOT allows for the embeddings to maintain their own topology for each individual domain as seen on the left, yet they are well aligned after the projection with the linear Monge mapping as seen on the right. Invariant feature transformation learning forces the embeddings to be close to each other in the embedding space but achieves a less precise alignment of the data in the embedding space. In this particular case, both achieve good performance, yet LaOT manages to do it in fewer epochs due to the additional flexibility that it has that does not require it to perfectly align the two domains.

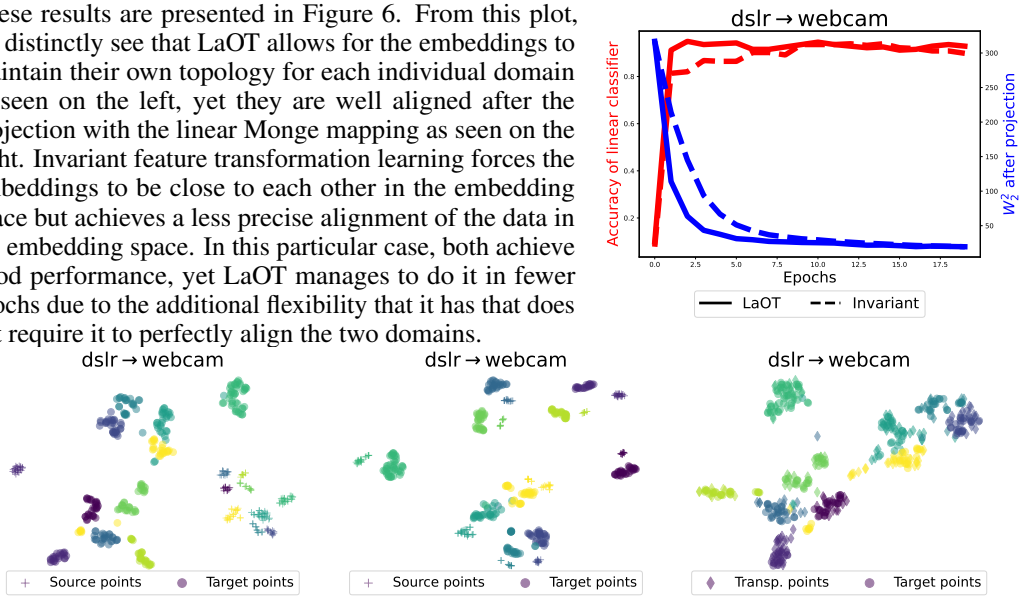

Figure 6: Comparison with invariant feature transformation learning. **(left)** embeddings learned with LaOT; **(middle)** embeddings learned with invariant feature transformation; **(right)** source and target data after the projection with linear Monge mapping in the embedding space. **Upper right**: learning dynamics comparing the two models.