Diffusion Transport Alignment

Andrés F. Duque[1], Guy Wolf[2], and Kevin R. Moon[1]

[1] Utah State University, Logan UT, USA
[2] kevin.moon@usu.edu.com
[3] Université de Montréal; Mila - Quebec AI Institute, Montréal, Canada
guy.wolf@umontreal.ca

Abstract. The integration of multimodal data presents a challenge in cases where the study of a given phenomena by different instruments or conditions generates distinct but related domains. Many existing data integration methods assume a known one-to-one correspondence between domains of the entire dataset, which may be unrealistic. Furthermore, existing manifold alignment methods are not suited for cases where the data contains domain-specific regions, i.e., there is not a counterpart for a certain portion of the data in the other domain. We propose Diffusion Transport Alignment (DTA), a semi-supervised manifold alignment method that exploits prior knowledge of between only a few points to align the domains. After building a diffusion process, DTA finds a transportation plan between data measured from two heterogeneous domains with different feature spaces, which by assumption, share a similar geometrical structure coming from the same underlying data generating process. DTA can also compute a partial alignment in a data-driven fashion, resulting in accurate alignments when some data are measured in only one domain. We empirically demonstrate that DTA outperforms other methods in aligning multiview data in this semi-supervised setting. We also show that the alignment obtained by DTA can improve the performance of machine learning tasks, such as domain adaptation, inter-domain feature mapping, and exploratory data analysis, while outperforming competing methods.

Keywords: Manifold alignment · Semi-supervised learning · Manifold learning

1 Introduction

In many data science applications, data may be collected from different measurement instruments, conditions, or protocols of the same underlying system. Examples include single cell RNA sequence and ATAC sequence measurements of the same group of cells [30], text documents translated into different languages [24], brain images from multiple neuroimaging techniques [33], and images of a scene captured from different views [17]. In such settings, researchers are often interested in integrating data from the different domains to enhance our understanding of the system as well as the relationships between the different

domains. Integrating the data may also lead to improved downstream analysis, such as classification, if there is domain-specific information about the task.

Multi-view data integration is usually performed assuming knowledge of one-to-one correspondences, i.e., the data comes in a paired fashion between domains. One of the simplest methods for this setting is Canonical Correlation Analysis (CCA), a linear approach that finds a projection that maximizes the correlation between the two domains [31]. Kernel CCA extends this to nonlinear projections via the kernel trick [5, 13]. Alternating diffusion [18] and integrated diffusion [19] are nonlinear alignment methods based on the robust manifold learning algorithm Diffusion Maps [8]. For an overview of other approaches see [14, 21].

A popular way to integrate distinct domains is manifold alignment. First introduced in the seminal works [15] and [16], this family of methods seeks to find projections of the multiple domains into a common latent space where inter-domain relationships can be captured. Manifold alignment can be performed in various scenarios, depending on how much information is provided about the correspondence between different domains. The edge case, usually referred to as *unsupervised manifold alignment*, arises in the absence of any relationship known a priori between the domains as in [3, 4, 11, 12, 29, 35]. Some of the data integration approaches described previously, such as CCA, may be viewed as belonging to the opposite edge case of *supervised manifold alignment*.

In contrast, other problems can be categorized as *semi-supervised manifold alignment*, where some degree of correspondence between domains is assumed to be known. In some cases, a one-to-one correspondence is known for only a few of the data points. This is the case in [16], which uses the Laplacian eigenmaps loss function in both domains while penalizing mismatches of known correspondences in the embedding. In [34], the authors first learn a latent representation for each domain using a variation of Laplacian eigenmaps [2]. Then, they use Procrustes analysis in the common embedding space to find a transformation that aligns the matching observations, which subsequently is applied to the rest of the data. Similarly, the approach proposed in [20] finds a low dimensional embedding generated by diffusion maps [8] and then performs an affine transformation to align the known correspondences. More recently, a generative adversarial network called manifold alignment GAN (MAGAN) was introduced in [1]. MAGAN is based on a similar architecture as cycleGAN [38], which learns functions that map from one domain to another. However, the authors of MAGAN showed that cycleGAN and similar approaches tend to superimpose rather than align the data manifolds, resulting in incorrect alignments between distinct groups. To mitigate this issue, MAGAN incorporates a correspondence loss between the known correspondences enforcing a consistent alignment.

Alternatively, the correspondence information may be available at the feature level. MAGAN can be applied to this case with a correspondence loss imposed on the shared features. Other approaches use class labels in both domains as the correspondence knowledge, as in [36] where the labels act as anchors points for the alignment. This was further expanded to a kernelized version in [32].

In this work we focus on the semi-supervised problem where we assume a known one-to-one correspondence between domains is available for a few of the data points. Our method, called Diffusion Transport Alignment (DTA), starts by building a diffusion process [8] that connects measurements in different domains via the known correspondences. In this fashion, DTA transforms both domains to a shared feature space, allowing us to extract inter-domain distances. Finally, DTA solves a partial optimal transport problem to determine a coupling between data samples from one domain and their counterparts in the other domain.. The obtained coupling can be further used to improve the performance of downstream analysis. For instance, one may be interested in learning a mapping between both domains, but the known correspondences are insufficient to successfully train a regression model. Another use-case is to perform unsupervised multi-domain analysis with methods as in [22] or [18], which require one-to-one correspondences between all points in all domains. DTA is also useful for domain adaptation, where a model is trained on a source domain and then applied to a target domain.

In summary, our contributions are as follows: 1) We develop a manifold alignment method, DTA, that outperforms current methods in recovering inter-domain relationships. 2) DTA can perform a data-driven partial alignment when a subset of the data is domain-specific, preventing spurious couplings between domains. 3) We demonstrate how DTA can leverage limited correspondence knowledge to improve the performance in other tasks, such as regression and domain adaptation.

2 Diffusion Transport Alignment

Consider a multi-domain data collection of a data generating process where two different views in potentially different feature spaces $\Phi_1 \in \mathrm{R}^{n \times q}$ and $\Phi_2 \in \mathrm{R}^{m \times p}$ are measured, containing observations $\{x_i\}_{i=1}^n$, and $\{y_i\}_{i=1}^m$, respectively. We wish to learn a correspondence between both domains in a semi-supervised setting, where one-to-one correspondence is known for a set of observations denoted by $\mathcal{C}$. That is, for each $c \in \mathcal{C}$ we have access to its features in both domains.

As a motivating example, consider a classification problem where both domains contain labeled data points for some shared classes. The two domains may contain distinct information that is relevant for classification. An example of this is in single cell data with both RNA-sequencing and ATAC-sequencing measurements. In this case, training on the aligned data will lead to improved performance compared with training on the domains separately. As another example, researchers may be interested in the relationships between variables measured in separate domains. Aligning the domains enables a larger dataset to obtain more accurate estimates of relationship measures such as the correlation coefficient or mutual information.

The fundamental idea of DTA consists of learning a diffusion process in each particular domain, and then leverage the known correspondences as anchor points to find a common feature representation. Ultimately, this allows us to extract an inter-domain distance measure, providing a dissimilarity among the observations in both domains. The diffusion operators over each domain, denoted as P_{Φ_1} and

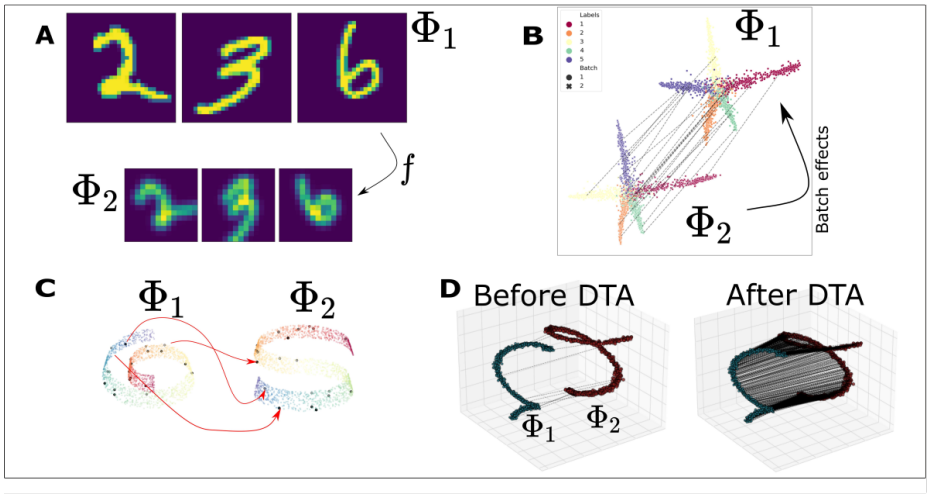

Fig. 1. Motivating examples for DTA. In all of these examples we have data measured in two distinct domains Φ_1 and Φ_2, and we possess a small subset of matching observations $\mathcal{C}$. This corresponds to the scenario where obtaining corresponding measurements may be costly, e.g. via expert annotation. The goal of DTA is to leverage the small subset of known correspondences to align the remaining observations. **A) Distorted MNIST digits**. Here Φ_1 consists of the original MNIST digits, while Φ_2 consists of distorted images after applying multiple transformations: rotation, downscaling, and Gaussian blurring. To learn a parametric function that maps from one domain to the other, the small set of correspondences is not enough. Thus, we need to find a greater set of matching data. **B) Splatter simulation with batch effects [37]**. A common problem when dealing with biological data is the distortion produced by the measurement protocols, introducing what is known as batch effects. Accurate alignment would overcome theses batch effects. **C) Swiss roll and S curve**. This case presents the ideal scenario where the two domains are a smooth mapping from a common latent space. Black points indicate correspondences with three of them (red arrows) highlighted. **D) Two helixes**. Here we use a dataset from [32] and display the effect of DTA after leveraging the known correspondences to align both manifolds.

P_{Φ_2}, are built by a standard approach. First, we compute an affinity matrix with an α-decay kernel [27]:

$$K_{k,\alpha}(x_i, x_j) = \frac{1}{2}\exp\left(-\frac{||x_i - x_j||^\alpha}{\sigma_k^\alpha(x_i)}\right) + \frac{1}{2}\exp\left(-\frac{||x_i - x_j||^\alpha}{\sigma_k^\alpha(x_j)}\right), \tag{1}$$

where $\sigma_k(x_i)$ is the k-nearest neighbor distance of x_i and $\alpha > 0$. This kernel has two hyper-parameters α and k, which provide a trade-off between connectivity in the graph and local geometry preservation. Methods that employ this kernel are typically robust to the choice of these hyper-parameters [27]. The diffusion operator P is then computed by row-normalizing the kernel matrix. In this way P can be viewed as a probability transition matrix, representing a Markov chain between observations. The probabilities of transitioning from one point to any other within a $t-$step random walk are obtained by powering the diffusion operator P^t. This particular kernel choice is not required

for our method, and the construction of the diffusion operator can be adapted to the particular problem.

DTA computes the transition probabilities between observations in Φ_1 and Φ_2 and elements in $\mathcal{C}$ in their respective domain by diffusing the process several steps, obtaining $P^t_{\Phi_1}$ and $P^t_{\Phi_2}$. The entries (i, c) of $P^t_{\Phi_k}$ with $c \in \mathcal{C}$ contain the transition probabilities from each observation $i \in \Phi_k$ to the observations in $\mathcal{C}$. Thus, we can extract the columns and rows of $P^t_{\Phi_1}$ and $P^t_{\Phi_2}$ associated with the elements in $\mathcal{C}$, obtaining the submatrices: $\Gamma_{\Phi_1} \in \mathrm{R}^{n\times|\mathcal{C}|}$, $\Gamma_{\Phi_2} \in \mathrm{R}^{m\times|\mathcal{C}|}$.

This construction provides a common feature representation, and thus, a natural way to compute inter-domain distances:

$$D_{ij} = \left(1 - \frac{\langle \Gamma_{\Phi_1}(i,:), \Gamma_{\Phi_2}(j,:)\rangle}{||\Gamma_{\Phi_1}(i,:)||||\Gamma_{\Phi_2}(j,:)||}\right). \tag{2}$$

We resort to cosine over euclidean distances since it resulted in superior performance.

The matrix D contains inter-domain distances, but does not provide a direct alignment of the domains. The final step in DTA is to solve a partial optimal transport problem with D as the cost matrix:

$$\begin{aligned} \min_{T} \quad & \sum_{i=1}^{n}\sum_{j=1}^{m} D_{ij}T_{ij} \\ \text{s.t.} \quad & \sum_{i=1}^{n} T_{ij} \le q_j,\ \forall j \in \{1,\dots,m\};\ \ \sum_{j=1}^{m} T_{ij} \le v_i,\ \forall i \in \{1,\dots,n\} \\ & \sum_{i=1}^{n}\sum_{j=1}^{m} T_{ij} = M;\ \ T_{ij} \ge 0,\ \ \forall i \in \{1,\dots,n\}, \forall j \in \{1,\dots,m\}. \end{aligned} \tag{3}$$

Optimal transport has been extensively used in data science [28], and is a common tool for transfer learning and domain adaptation [6, 9, 10, 25]. It provides a principled framework to compute a distance between probability distributions, also known as the Wasserstein distance, by finding the minimal effort required to "transport" the mass of one distribution to another. Our formulation deviates from the original optimal transport problem by constraining the total mass M to be transported. As we show in Section 3.1, M can be selected in a data-driven fashion, permitting alignments that respect domain-specific regions that are not present in the other domain.

The user-defined parameters q_j and v_i indicate the mass assigned to each observation. For instance, to find a hard assignment from each observation in Φ_1 to Φ_2, and if $n \le m$, we can set $v_i = 1/n$, $q_j = 1/n$ and $M = 1$, which is the case for the experiments in Section 3. Soft assignments can be obtained by different choices of masses. Alternatively an entropy regularization $\epsilon \sum_{i,j} T_{ij}\log(T_{ij})$ can be added to the objective function. In this work we focus on hard assignments since we want to learn one-to-one correspondences. Nevertheless, we state the general formulation, which is useful when there is less confidence in the existence of one-to-one correspondences.

The coupling T contains the information required to combine both manifolds. After a *min-max* normalization denoted by $\tilde{T}$, we can find a projection of a given sample $x_i \in \Phi_1$ on Φ_2 by its barycentric projection $x_i \mapsto \sum_j \tilde{T}_{ij} y_j$. Alternatively, we can build a cross-modality similarity matrix $W_{\Phi_1\Phi_2} = (W_{\Phi_1}\tilde{T} + \tilde{T}W_{\Phi_2})$, where W_{Φ_k} are the similarities in each domain (computed using Eq. (1) in this paper). Using a similar

construction as in [16] we can build a joint manifold learning loss:

$$\mathcal{L} = \mu \sum_{ij} ||f_i - f_j|| W^{ij}_{\Phi_1} + \mu \sum_{ij} ||g_i - g_j|| W^{ij}_{\Phi_2} + (1-\mu) \sum_{ij} ||f_i - g_j|| W^{ij}_{\Phi_1\Phi_2}. \quad (4)$$

The parameter μ controls the preservation of the intra-domain geometry. The solution of (4) provides a shared embedding where $\boldsymbol{f}$ and $\boldsymbol{g}$ represent the embedding coordinates for both domains. They are the generalized eigenvectors of the graph Laplacian matrix associated with the joint similarity matrix:

$$W = \begin{bmatrix} \mu W_{\Phi_1} & (1-\mu) W_{\Phi_1\Phi_2} \\ (1-\mu) W'_{\Phi_1\Phi_2} & \mu W_{\Phi_2} \end{bmatrix}. \quad (5)$$

DTA differs from [16] in several ways. First, their method starts by solving (4), with a T matrix instead of $W_{\Phi_1\Phi_2}$, which encodes only the *a priori* known correspondences, containing a 1 in entry (i, j) if $x_i \in \Phi_1$ corresponds to $y_j \in \Phi_2$ and 0 otherwise. Inter-domain correspondences for the rest of the data are obtained in the latent space produced by the solution. In contrast, DTA first finds a matrix T that couples all the data, and then builds the inter-domain similarities based on these correspondences. Second, using only T in (4) assigns a 0 similarity between x_i and the neighbors of y_j. We argue that a more natural way to construct the off-diagonal matrices of W is to include the neighbors of y_j as being similar to x_i as well, motivating our particular construction of $W_{\Phi_1\Phi_2}$.

3 Experimental results

To demonstrate DTA's effectiveness in finding a coupling between domains, we compare DTA with semi-supervised manifold alignment (SSMA) [16], manifold alignment with Procrustes analysis (MA-PA) [34], and MAGAN [1]. For consistency, we use the same α-decay Kernel in Eq. (1) for the graph-based methods DTA, SSMA, and MA-PA, with $\alpha = 10$ and $k = 10$. For MAGAN we use the same architecture provided by the author's code[4]. MAGAN's architecture is composed of two generators, one mapping from Φ_1 to Φ_2 and the other in the opposite direction, and two discriminators, one for each domain. The model is trained via a *min-max* game between the generators and discriminators, with a cycle consistency loss [38], and a correspondence loss that tries to preserve the known correspondences. We found that MAGAN usually needs an extra penalization parameter ρ in the correspondence loss to improve its performance, which was not included in the original paper.

Given the nature of the problem, it is difficult to tune the hyper-parameters present in each method. Thus, we set the same values for each method across all the experiments. This leave us with one hyperparameter t for DTA, which we set equal to 10 for all the experiments. SSMA and MA-PA require a predefined number of dimensions for the latent space. We selected all eigenvectors associated with non-zero eigenvalues. We set $\rho = 1000$ for MAGAN.

We used four simulated datasets shown in Figure 1. **MNIST-Double**: one domain contains the original MNIST digits, while the other is constructed by downscaling the images to 14x14 pixels, applying a rotation, and adding Gaussian blurring. **SWISSR-SCURVE**: starting from a common 2D latent space we apply two different transformations resulting in the well known swiss roll and s-curve manifolds embedded in a 3D

[4] https://github.com/KrishnaswamyLab/MAGAN/tree/master/MAGAN

space. **STL10**: a popular dataset for computer vision [7]. The first domain contains the original images, and we generated the second by applying brightness, gray scaling, and Gaussian blurring. We performed feature extraction using the 512 outputs after the last convolution layer in ResNet-18. **SPLATTER-BE**: we simulated single-cell RNA-sequencing data using Splatter [37]. The difference between Φ_1 and Φ_2 is due to batch effects, which often arise in biological experiments. For real data, we used the single-cell dataset from the *Multimodal Single-Cell Data Integration* challenge, NeurIPS competition track 2021. The data contains two sets with jointly measured observations for both domains, providing us ground truth information about the coupling between domains. The first set measures gene expression (RNA) and protein abundance (ADT), while the second measures RNA and chromatin accessibility (ATAC). The samples are taken from different donors and batches. We selected batches "s1d1" in both sets for our experiments. Both RNA and ATAC domains are preprocessed, reducing their dimensionality to 1000 features via truncated SVD.

Inter-domain feature mapping. Our first comparison metric is the regression performance when mapping between the two domains. When the prior known correspondences are insufficient to successfully train a model, we can improve the training data by expanding the correspondences using each of the considered manifold alignment methods. For DTA, we use hard assignments where for each observation in Φ_1 we assign an unique counterpart in Φ_2. The correspondences in SSMA and MA-PA are computed as suggested in [16], where the assigned counterpart for each observation in Φ_1 corresponds to its nearest sample from Φ_2 in the shared latent space. For MAGAN, once the model is trained, we map the data from the first domain into the second using one of the generators. The assigned correspondence is the closest sample. The newly found correspondences serve as the training data for the regression task.

To reduce the dependency on a given regression model, we trained both a fully-connected neural network and a Kernel Ridge Regression (KRR) model. Since the true one-to-one correspondences are accessible to us, the regression models are also trained with the complete data, as well as the *a priori* known correspondences. This provides a baseline to show the improvement due to the new information acquired after each of the manifold alignment models, and how well they perform compared to the full correspondence case.

The results are summarized in Table 1 with the test MSE values for each model as well as for the regression trained using all of the correct correspondences. DTA is the most consistent method as it almost always outperforms the other methods across different datasets and different levels of prior known correspondences.

Domain adaptation. Now we compare the methods on a domain adaptation problem. Table 2 contains the test error for two k-nearest neighbor classifiers, with $k = 1$ and $k = 10$. The classification models are trained on Φ_2 and then tested on the barycentric projections of Φ_1 onto Φ_2. The matrix $\tilde{T}$ is computed for SSMA, MA-PA, and MAGAN from the assigned correspondences as described above. An alternative approach for SSMA and MA-PA is to train and test the classification on the shared latent representation. For MAGAN the testing can be computed in the generator mapping from Φ_1 to Φ_2. Overall, DTA outperforms the other methods as it typically has the best performance and is in second otherwise. In contrast, while other methods occasionally outperform DTA on some datasets (e.g. MAGAN on MNIST-Double), these methods perform worse on other datasets.

Fraction of samples closer than the true match (FOSCTTM). Lastly, a common metric to measure the goodness of alignment was proposed in [23] and further employed by [4, 12] among others. The idea is to measure the proportion

Table 1. Regression MSE average over 10 runs. When both models (Neural network and KRR) are trained with all the ground truth correspondences a lower MSE is obtained, and if only the *a priori* known correspondences are used the worst results are obtained for the majority of scenarios.

		Test MSE (Neural Network)				Test MSE (KRR)			
		1%	2%	5%	10%	1%	2%	5%	10%
Dataset	Model								
MNIST-Double	AllData	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
	PriorInfo	0.012	0.008	0.003	0.001	0.011	0.006	0.002	0.000
	DTA	**0.006** (2)	**0.004** (2)	**0.002** (1)	**0.002** (1)	**0.005** (2)	**0.003** (2)	**0.002** (1)	**0.001** (1)
	MA-PA	0.012 (3)	0.009 (3)	0.006 (3)	0.004 (3)	0.012 (3)	0.009 (3)	0.005 (3)	0.003 (3)
	MAGAN	**0.002** (1)	**0.002** (1)	**0.003** (2)	**0.002** (2)	**0.001** (1)	**0.002** (1)	**0.002** (2)	**0.001** (2)
	SSMA	0.013 (4)	0.010 (4)	0.007 (4)	0.005 (4)	0.012 (4)	0.009 (4)	0.006 (4)	0.004 (4)
RNA-ADT	AllData	0.109	0.108	0.109	0.109	0.104	0.104	0.104	0.105
	PriorInfo	0.718	0.519	0.330	0.243	0.304	0.204	0.177	0.173
	DTA	**0.130** (1)	**0.131** (1)	**0.125** (1)	**0.124** (1)	**0.115** (1)	**0.116** (1)	**0.112** (1)	**0.112** (1)
	MA-PA	0.230 (4)	0.190 (4)	0.147 (4)	0.137 (4)	0.235 (4)	0.180 (4)	0.125 (4)	0.117 (3)
	MAGAN	0.175 (3)	**0.143** (2)	**0.133** (2)	0.133 (3)	0.162 (3)	**0.129** (2)	0.121 (3)	0.122 (4)
	SSMA	**0.170** (2)	0.163 (3)	0.136 (3)	**0.130** (2)	**0.148** (2)	0.140 (3)	**0.118** (2)	**0.115** (2)
RNA-ATAC	AllData	0.369	0.369	0.369	0.370	0.346	0.346	0.346	0.346
	PriorInfo	0.522	0.472	0.431	0.399	0.406	0.376	0.361	0.355
	DTA	**0.422** (1)	**0.404** (2)	0.404 (3)	0.397 (3)	**0.419** (1)	**0.401** (1)	0.397 (3)	0.388 (3)
	MA-PA	**0.430** (2)	**0.403** (1)	**0.386** (1)	**0.387** (1)	0.460 (3)	**0.402** (2)	**0.373** (1)	**0.368** (1)
	MAGAN	0.661 (4)	0.664 (4)	0.648 (4)	0.544 (4)	0.661 (4)	0.662 (4)	0.643 (4)	0.537 (4)
	SSMA	0.443 (3)	0.410 (3)	**0.399** (2)	**0.396** (2)	**0.456** (2)	0.403 (3)	**0.383** (2)	**0.374** (2)
SPLATTER-BE	AllData	0.372	0.396	0.391	0.401	0.376	0.376	0.376	0.377
	PriorInfo	0.440	0.424	0.413	0.405	0.457	0.470	0.414	0.398
	DTA	**0.388** (1)	**0.377** (1)	**0.397** (1)	**0.406** (1)	**0.377** (1)	**0.376** (1)	**0.377** (1)	**0.377** (1)
	MA-PA	0.410 (3)	0.409 (3)	**0.408** (2)	0.409 (3)	0.401 (3)	0.403 (3)	0.393 (3)	0.390 (3)
	MAGAN	0.466 (4)	0.518 (4)	0.466 (4)	0.481 (4)	0.483 (4)	0.527 (4)	0.475 (4)	0.498 (4)
	SSMA	**0.407** (2)	**0.408** (2)	0.408 (3)	**0.409** (2)	**0.387** (2)	**0.386** (2)	**0.386** (2)	**0.386** (2)
STL10	AllData	0.373	0.374	0.374	0.378	0.321	0.322	0.323	0.325
	PriorInfo	0.564	0.530	0.497	0.467	0.557	0.534	0.476	0.433
	DTA	**0.470** (1)	**0.461** (1)	**0.458** (1)	**0.454** (1)	**0.460** (1)	**0.444** (1)	**0.438** (1)	**0.433** (1)
	MA-PA	0.532 (3)	0.503 (3)	0.479 (3)	**0.468** (2)	0.554 (3)	0.507 (3)	0.471 (3)	0.452 (3)
	MAGAN	0.552 (4)	0.537 (4)	0.562 (4)	0.498 (4)	0.564 (4)	0.532 (4)	0.546 (4)	0.469 (4)
	SSMA	**0.503** (2)	**0.484** (2)	**0.476** (2)	0.469 (3)	**0.489** (2)	**0.474** (2)	**0.464** (2)	**0.451** (2)
SWISSR-SCURVE	AllData	0.002	0.003	0.001	0.001	0.000	0.000	0.000	0.000
	PriorInfo	0.682	0.648	0.263	0.151	0.610	0.311	0.036	0.004
	DTA	**0.043** (2)	**0.015** (1)	**0.003** (1)	**0.001** (1)	**0.036** (2)	**0.008** (1)	**0.001** (1)	**0.000** (1)
	MA-PA	**0.018** (1)	0.064 (3)	0.044 (3)	0.021 (4)	**0.014** (1)	0.061 (3)	0.043 (3)	0.017 (4)
	MAGAN	0.620 (4)	0.546 (4)	0.088 (4)	**0.004** (2)	0.682 (4)	0.513 (4)	0.088 (4)	**0.002** (2)
	SSMA	0.267 (3)	**0.039** (2)	**0.012** (2)	0.006 (3)	0.204 (3)	**0.027** (2)	**0.010** (2)	0.003 (3)

Table 2. Domain adaptation classification accuracy results under different correspondence percentages. Overall DTA achieves the best results as it is consistently in the top two.

		KNN-1				KNN-10			
		1%	2%	5%	10%	1%	2%	5%	10%
DATASET	MODEL								
MNIST-Double	DTA	**0.79** (2)	**0.87** (2)	**0.92** (2)	**0.94** (2)	**0.79** (2)	**0.85** (2)	**0.88** (2)	**0.89** (2)
	MA-PA	0.65 (3)	0.75 (3)	0.80 (3)	0.84 (3)	0.64 (3)	0.75 (3)	0.78 (3)	0.81 (3)
	MAGAN	**0.96** (1)	**0.95** (1)	**0.95** (1)	**0.97** (1)	**0.89** (1)	**0.88** (1)	**0.88** (1)	**0.89** (1)
	SSMA	0.42 (4)	0.55 (4)	0.65 (4)	0.75 (4)	0.42 (4)	0.56 (4)	0.65 (4)	0.73 (4)
RNA-ADT	DTA	**0.67** (1)	**0.68** (1)	**0.73** (1)	**0.73** (1)	**0.67** (1)	**0.67** (1)	**0.72** (1)	**0.72** (1)
	MA-PA	0.61 (3)	0.64 (3)	**0.70** (2)	**0.71** (2)	0.52 (4)	0.58 (4)	0.61 (4)	0.63 (4)
	MAGAN	0.61 (4)	0.62 (4)	0.69 (4)	0.65 (4)	**0.60** (2)	**0.61** (2)	**0.66** (2)	**0.66** (2)
	SSMA	**0.64** (2)	**0.66** (2)	0.69 (3)	0.70 (3)	0.58 (3)	0.60 (3)	0.63 (3)	0.65 (3)
RNA-ATAC	DTA	**0.66** (1)	**0.72** (1)	**0.77** (1)	**0.78** (1)	**0.61** (1)	**0.67** (1)	**0.70** (1)	**0.71** (1)
	MA-PA	**0.61** (2)	**0.70** (2)	**0.76** (2)	**0.76** (2)	0.54 (3)	**0.62** (2)	**0.66** (2)	**0.66** (2)
	MAGAN	0.30 (4)	0.32 (4)	0.42 (4)	0.53 (4)	0.31 (4)	0.33 (4)	0.44 (4)	0.54 (4)
	SSMA	0.59 (3)	0.65 (3)	0.70 (3)	0.72 (3)	**0.56** (2)	0.61 (3)	0.63 (3)	0.65 (3)
SPLATTER-BE	DTA	**0.83** (1)	**0.84** (1)	**0.84** (1)	**0.83** (1)	**0.79** (1)	**0.80** (1)	**0.80** (1)	**0.80** (1)
	MA-PA	**0.65** (2)	**0.57** (2)	**0.61** (2)	0.61 (3)	**0.65** (2)	**0.57** (2)	**0.62** (2)	**0.61** (2)
	MAGAN	0.30 (4)	0.30 (4)	0.42 (4)	0.46 (4)	0.31 (4)	0.30 (4)	0.43 (4)	0.47 (4)
	SSMA	0.51 (3)	0.54 (3)	0.58 (3)	**0.61** (2)	0.51 (3)	0.54 (3)	0.57 (3)	0.61 (3)
STL10	DTA	**0.75** (1)	**0.80** (1)	**0.81** (1)	**0.82** (1)	**0.71** (2)	**0.75** (1)	**0.76** (1)	**0.76** (1)
	MA-PA	**0.73** (2)	**0.73** (2)	**0.74** (2)	**0.72** (2)	**0.74** (1)	**0.73** (2)	**0.74** (2)	**0.73** (2)
	MAGAN	0.51 (4)	0.61 (3)	0.56 (4)	0.71 (3)	0.52 (4)	0.63 (3)	0.59 (4)	0.72 (3)
	SSMA	0.53 (3)	0.61 (4)	0.65 (3)	0.69 (4)	0.53 (3)	0.61 (4)	0.65 (3)	0.69 (4)

of observations that are closer to the true match after alignment, and average over the entire dataset. Thus, the lower this number, the better are the samples aligned with their counterparts in the opposite domain. Since this metric can be measured in different spaces after alignment, we include three different cases in Table 3. After

alignment, we can compute the distances after computing the barycentric projection in the ambient space. Alternatively, it is possible to find a low dimensional representation after computing the spectral embedding using the matrix W, and find the neighbors and distances in this new representation. In particular, we computed the FOSCTTM metric in both, the 2 and 10 dimensional embeddings.

Table 3. FOSCTTM average over 10 runs. DTA consistently achieves the best or second best performance.

		10-dim Emb.		2-dim Emb.		Barycentric proj.	
		1%	10%	1%	10%	1%	10%
DATASET	MODEL						
MNIST-Double	DTA	**0.01** (2)	**0.00** (1)	**0.03** (2)	**0.01** (1)	**0.05** (2)	**0.01** (2)
	MA-PA	0.14 (3)	0.01 (3)	0.08 (3)	0.03 (3)	0.14 (3)	0.04 (3)
	MAGAN	**0.01** (1)	**0.00** (2)	**0.02** (1)	**0.01** (2)	**0.01** (1)	**0.01** (1)
	SSMA	0.26 (4)	0.18 (4)	0.28 (4)	0.22 (4)	0.22 (4)	0.06 (4)
RNA-ADT	DTA	**0.20** (1)	**0.14** (1)	**0.11** (1)	**0.10** (1)	**0.10** (1)	**0.09** (1)
	MA-PA	0.40 (3)	0.22 (3)	0.19 (3)	0.26 (3)	0.16 (4)	0.12 (4)
	MAGAN	**0.25** (2)	**0.22** (2)	**0.14** (2)	**0.12** (2)	**0.12** (2)	**0.10** (2)
	SSMA	0.40 (4)	0.36 (4)	0.43 (4)	0.41 (4)	0.13 (3)	0.10 (3)
RNA-ATAC	DTA	**0.29** (1)	**0.20** (2)	**0.17** (1)	**0.13** (1)	**0.37** (1)	**0.33** (1)
	MA-PA	**0.36** (2)	**0.19** (1)	**0.25** (2)	**0.27** (2)	**0.38** (2)	**0.33** (2)
	MAGAN	0.49 (4)	0.41 (4)	0.44 (3)	0.32 (3)	0.46 (4)	0.41 (4)
	SSMA	0.44 (3)	0.34 (3)	0.45 (4)	0.42 (4)	0.38 (3)	0.35 (3)
SPLATTER-BE	DTA	**0.14** (1)	**0.13** (1)	**0.14** (1)	**0.14** (1)	**0.27** (1)	**0.26** (1)
	MA-PA	**0.30** (2)	**0.22** (2)	**0.22** (2)	**0.20** (2)	**0.32** (2)	0.34 (3)
	MAGAN	0.42 (4)	0.31 (3)	0.44 (4)	0.33 (3)	0.40 (4)	**0.32** (2)
	SSMA	0.42 (3)	0.39 (4)	0.42 (3)	0.44 (4)	0.37 (3)	0.34 (4)
STL10	DTA	**0.07** (1)	**0.05** (2)	**0.10** (1)	**0.07** (1)	**0.17** (1)	**0.13** (2)
	MA-PA	**0.24** (2)	0.10 (3)	**0.18** (2)	0.14 (3)	**0.21** (2)	0.16 (3)
	MAGAN	0.27 (3)	**0.05** (1)	0.24 (3)	**0.08** (2)	0.23 (3)	**0.11** (1)
	SSMA	0.36 (4)	0.32 (4)	0.40 (4)	0.36 (4)	0.26 (4)	0.17 (4)
SWISSR-SCURVE	DTA	**0.01** (1)	**0.00** (1)	**0.02** (2)	**0.00** (1)	**0.03** (2)	**0.00** (1)
	MA-PA	**0.05** (2)	0.00 (3)	**0.02** (1)	0.01 (3)	**0.01** (1)	0.02 (4)
	MAGAN	0.15 (4)	**0.00** (2)	0.19 (4)	**0.01** (2)	0.17 (4)	**0.00** (2)
	SSMA	0.14 (3)	0.08 (4)	0.15 (3)	0.09 (4)	0.13 (3)	0.01 (3)

Overall, DTA achieves the best results in this metric for the various types of comparisons. MAGAN performs considerably well for MNIST-Double, but it tends to have the worst performance in the more complex single-cell datasets.

3.1 Partial alignment

Here we show the ability of DTA to perform partial alignment. Figure 2 demonstrates this scenario where the data in one or both domains is not completely represented in the other. If, for instance, we use MAGAN to perform the alignment, the nature of its *min-max* game will map samples from one domain into high density regions of the other. This causes false positive correspondences, and an incorrect alignment for some portions of the data. In contrast, DTA can handle this scenario in a data-driven way. The idea is to select a value of M in (3), that corresponds to the mass from Φ_1 that has an actual counterpart in Φ_2. We select M using the normalized transportation cost: $NTC = \frac{\sum_{ij} D_{ij} T_{ij}}{M}$.

After selecting a grid of values for M ranging from 0 to 1, we solve (3) for each particular value and compute its corresponding NTC. The transportation cost for observations far away from the known correspondences (i.e. points that are present in only one of the domains) starts to increase rapidly after a certain threshold that likely corresponds to the case where all of the shared points have been aligned. Thus the selected mass M to be transported is computed by identifying a knee point in the NTC vs M plot (Figure 2B).

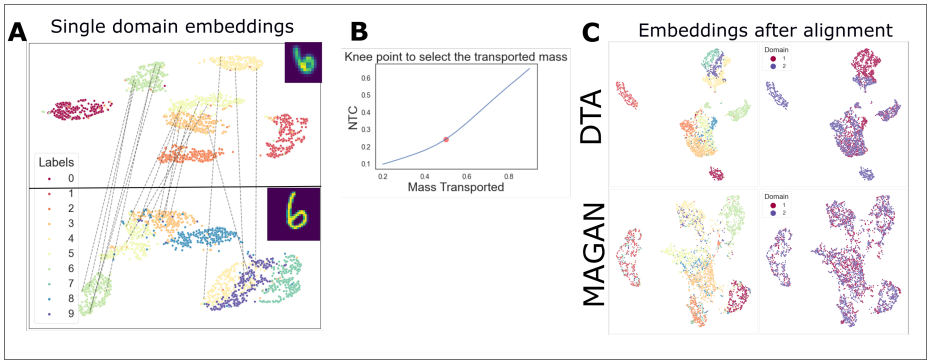

Fig. 2. Partial alignment. We subset both domains of the MNIST-Double dataset such that both domains contain specific regions with no counterpart in the other domain. **A)** Domain specific 2D UMAP [26] embeddings and dashed lines connecting the *a priori* known correspondences. **B)** Knee plot used to indentify the optimal mass M to be transported. **C)** Joint embedding of both domains after alignment, colored by labels and domain membership. DTA is able to retain domain-specific regions separate, while combining successfully the true counterparts. In contrast, MAGAN maps regions of Φ_1 to non-corresponding counterparts in Φ_2.

A quantitative evaluation of DTA and MAGAN in this scenario is presented in Table 4. After finding the *min-max* normalized coupling matrix T, we compute W via (5) and transform it to a distance matrix used in a kNN classifier. The test accuracy values are reported and, as expected, the results show how MAGAN maps observations close to incompatible regions on Φ_2, deteriorating the performance of the classifier.

Table 4. Test accuracy for the partial alignment experiments. DTA outperforms MAGAN.

		KNN-1				KNN-10			
		1%	2%	5%	10%	1%	2%	5%	10%
DATASET	MODEL								
MNIST-Double (P)	DTA	**0.821**	**0.861**	**0.882**	**0.887**	**0.900**	**0.917**	**0.924**	**0.926**
	MAGAN	0.583	0.663	0.720	0.743	0.753	0.801	0.827	0.836
RNA-ADT (P)	DTA	**0.820**	**0.831**	**0.844**	**0.849**	**0.910**	**0.910**	**0.912**	**0.919**
	MAGAN	0.627	0.655	0.675	0.679	0.692	0.719	0.726	0.726

4 Conclusion

We introduced Diffusion Transport Alignment (DTA), a manifold alignment method that exploits prior known correspondences between two related domains. We showed that DTA is superior to previous state-of-the-art manifold alignment methods by various metrics of comparison. DTA is able to recover meaningful connections that can be leveraged for downstream analysis tasks that may be otherwise difficult to perform. We also showed that partial manifold alignment can be handled by DTA, reducing the likelihood of falsely connecting points between domains, whereas previous methods are not naturally equipped to tackle this case.

Acknowledgments. This research was supported in part by Canada CIFAR AI chair [G.W.], in part by NSERC under Discovery Grant 03267 [G.W.], in part by the NIH under Grant R01GM135929 [G.W.], and in part by the NSF under Grant 2212325 [K.M.].

References

1. Amodio, M., Krishnaswamy, S.: Magan: Aligning biological manifolds. In: International Conference on Machine Learning. pp. 215–223. PMLR (2018)
2. Belkin, M., Niyogi, P.: Laplacian eigenmaps for dimensionality reduction and data representation. Neural computation **15**(6), 1373–1396 (2003)
3. Cao, K., Bai, X., Hong, Y., Wan, L.: Unsupervised topological alignment for single-cell multi-omics integration. Bioinformatics **36**, 48–56 (2020)
4. Cao, K., Hong, Y., Wan, L.: Manifold alignment for heterogeneous single-cell multi-omics data integration using pamona. Bioinformatics **38**(1), 211–219 (2022)
5. Chang, B., Kruger, U., Kustra, R., Zhang, J.: Canonical correlation analysis based on hilbert-schmidt independence criterion and centered kernel target alignment. In: International Conference on Machine Learning. pp. 316–324. PMLR (2013)
6. Chapel, L., Alaya, M.Z., Gasso, G.: Partial optimal transport with applications on positive-unlabeled learning. arXiv preprint arXiv:2002.08276 (2020)
7. Coates, A., Ng, A., Lee, H.: An analysis of single-layer networks in unsupervised feature learning. In: Proceedings of the fourteenth international conference on artificial intelligence and statistics. pp. 215–223. JMLR Workshop and Conference Proceedings (2011)
8. Coifman, R.R., Lafon, S.: Diffusion maps. Applied and computational harmonic analysis **21**(1), 5–30 (2006)
9. Courty, N., Flamary, R., Habrard, A., Rakotomamonjy, A.: Joint distribution optimal transportation for domain adaptation. Advances in Neural Information Processing Systems **30** (2017)
10. Courty, N., Flamary, R., Tuia, D.: Domain adaptation with regularized optimal transport. In: Joint European Conference on Machine Learning and Knowledge Discovery in Databases. pp. 274–289. Springer (2014)
11. Cui, Z., Chang, H., Shan, S., Chen, X.: Generalized unsupervised manifold alignment. Advances in Neural Information Processing Systems **27** (2014)
12. Demetci, P., Santorella, R., Sandstede, B., Noble, W.S., Singh, R.: Scot: Single-cell multi-omics alignment with optimal transport. Journal of Computational Biology **29**(1), 3–18 (2022)
13. Gao, G., Ma, H.: Multi-modality movie scene detection using kernel canonical correlation analysis. In: Proceedings of the 21st International Conference on Pattern Recognition (ICPR2012). pp. 3074–3077. IEEE (2012)
14. Gravina, R., Alinia, P., Ghasemzadeh, H., Fortino, G.: Multi-sensor fusion in body sensor networks: State-of-the-art and research challenges. Information Fusion **35**, 68–80 (2017)
15. Ham, J.H., Lee, D.D., Saul, L.K.: Learning high dimensional correspondences from low dimensional manifolds (2003)
16. Ham, J., Lee, D., Saul, L.: Semisupervised alignment of manifolds. In: International Workshop on Artificial Intelligence and Statistics. pp. 120–127. PMLR (2005)
17. Hu, J., Hong, D., Zhu, X.X.: Mima: Mapper-induced manifold alignment for semi-supervised fusion of optical image and polarimetric sar data. IEEE Transactions on Geoscience and Remote Sensing **57**(11), 9025–9040 (2019)

18. Katz, O., Talmon, R., Lo, Y.L., Wu, H.T.: Alternating diffusion maps for multimodal data fusion. Information Fusion **45**, 346–360 (2019)
19. Kuchroo, M., Godavarthi, A., Tong, A., Wolf, G., Krishnaswamy, S.: Multimodal data visualization and denoising with integrated diffusion. In: 2021 IEEE 31st International Workshop on Machine Learning for Signal Processing (MLSP). pp. 1–6. IEEE (2021)
20. Lafon, S., Keller, Y., Coifman, R.R.: Data fusion and multicue data matching by diffusion maps. IEEE Transactions on pattern analysis and machine intelligence **28**(11), 1784–1797 (2006)
21. Lahat, D., Adali, T., Jutten, C.: Multimodal data fusion: an overview of methods, challenges, and prospects. Proceedings of the IEEE **103**(9), 1449–1477 (2015)
22. Lindenbaum, O., Yeredor, A., Salhov, M., Averbuch, A.: Multi-view diffusion maps. Information Fusion **55**, 127–149 (2020)
23. Liu, J., Huang, Y., Singh, R., Vert, J.P., Noble, W.S.: Jointly embedding multiple single-cell omics measurements. In: Algorithms in bioinformatics:... International Workshop, WABI..., proceedings. WABI (Workshop). vol. 143. NIH Public Access (2019)
24. Liu, Z., Wang, W., Jin, Q.: Manifold alignment using discrete surface ricci flow. CAAI Transactions on Intelligence Technology **1**(3), 285–292 (2016)
25. Lu, Y., Chen, L., Saidi, A.: Optimal transport for deep joint transfer learning. arXiv preprint arXiv:1709.02995 (2017)
26. McInnes, L., Healy, J., Melville, J.: Umap: Uniform manifold approximation and projection for dimension reduction. arXiv preprint arXiv:1802.03426 (2018)
27. Moon, K.R., van Dijk, D., Wang, Z., Gigante, S., Burkhardt, D.B., Chen, W.S., Yim, K., van den Elzen, A., Hirn, M.J., Coifman, R.R., et al.: Visualizing structure and transitions in high-dimensional biological data. Nature biotechnology **37**(12), 1482–1492 (2019)
28. Peyré, G., Cuturi, M., et al.: Computational optimal transport: With applications to data science. Foundations and Trends® in Machine Learning **11**(5-6), 355–607 (2019)
29. Stanley III, J.S., Gigante, S., Wolf, G., Krishnaswamy, S.: Harmonic alignment. In: Proceedings of the 2020 SIAM International Conference on Data Mining. pp. 316–324. SIAM (2020)
30. Stuart, T., Butler, A., Hoffman, P., Hafemeister, C., Papalexi, E., Mauck III, W.M., Hao, Y., Stoeckius, M., Smibert, P., Satija, R.: Comprehensive integration of single-cell data. Cell **177**(7), 1888–1902 (2019)
31. Thompson, B.: Canonical correlation analysis: Uses and interpretation. No. 47, Sage (1984)
32. Tuia, D., Camps-Valls, G.: Kernel manifold alignment for domain adaptation. PloS one **11**(2), e0148655 (2016)
33. Vieira, S., Pinaya, W.H.L., Garcia-Dias, R., Mechelli, A.: Multimodal integration. In: Machine Learning, pp. 283–305. Elsevier (2020)
34. Wang, C., Mahadevan, S.: Manifold alignment using procrustes analysis. In: Proceedings of the 25th international conference on Machine learning. pp. 1120–1127 (2008)
35. Wang, C., Mahadevan, S.: Manifold alignment without correspondence. In: Twenty-First International Joint Conference on Artificial Intelligence (2009)
36. Wang, C., Mahadevan, S.: Heterogeneous domain adaptation using manifold alignment. In: Twenty-second international joint conference on artificial intelligence (2011)

37. Zappia, L., Phipson, B., Oshlack, A.: Splatter: simulation of single-cell rna sequencing data. Genome biology **18**(1), 1–15 (2017)
38. Zhu, J.Y., Park, T., Isola, P., Efros, A.A.: Unpaired image-to-image translation using cycle-consistent adversarial networks. In: Proceedings of the IEEE international conference on computer vision. pp. 2223–2232 (2017)