Journal of Machine Learning Research 27 (2026) 1-60 Submitted 2/23; Published 7/26

Keypoint-Guided Optimal Transport: Models, Algorithms, and Applications

Xiang Gu XIANGGU@XJTU.EDU.CN
School of Mathematics and Statistics
Xi'an Jiaotong University, Xi'an, China
Peng Cheng Laboratory, Shenzhen, China

Yucheng Yang YCYANG@STU.XJTU.EDU.CN
School of Mathematics and Statistics
Xi'an Jiaotong University, Xi'an, China

Wei Zeng WZ@XJTU.EDU.CN
School of Mathematics and Statistics
Xi'an Jiaotong University, Xi'an, China

Jian Sun* JIANSUN@XJTU.EDU.CN
School of Mathematics and Statistics
Xi'an Jiaotong University, Xi'an, China

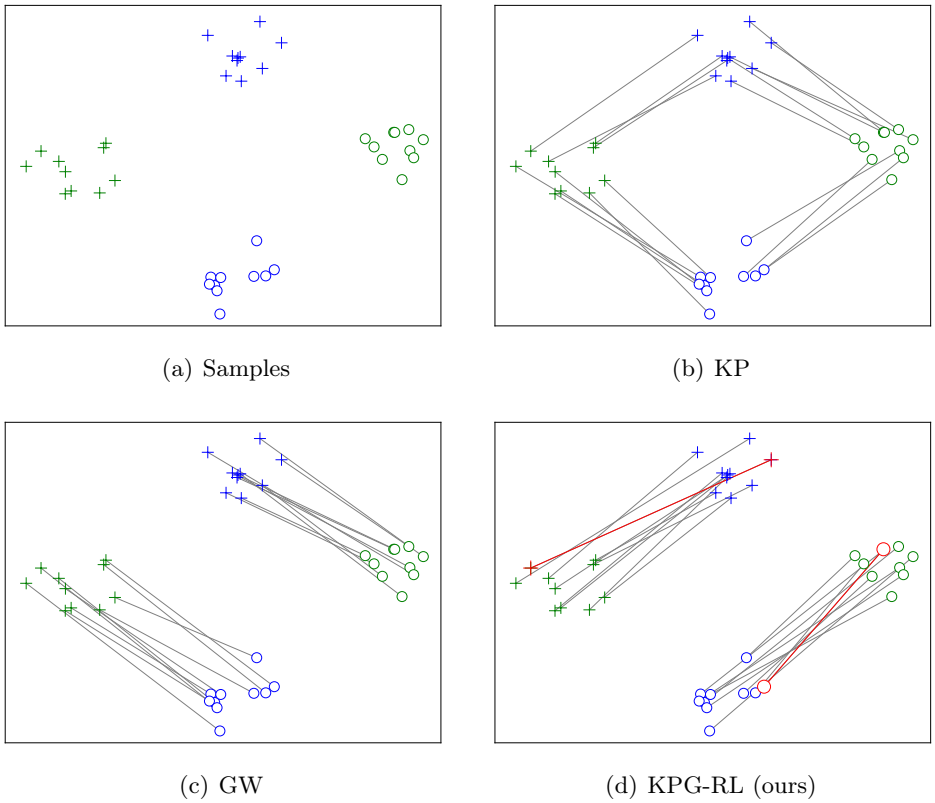
Zongben Xu ZBXU@XJTU.EDU.CN
School of Mathematics and Statistics
Xi'an Jiaotong University, Xi'an, China

Editor: Marco Cuturi

Abstract

Existing Optimal Transport (OT) methods mainly derive the optimal transport plan/matching under the criterion of transport cost/distance minimization, which may cause incorrect matching in some cases. In real applications, annotating a few matched keypoints across domains is reasonable or even effortless in annotation burden. It is valuable to investigate how to leverage the annotated keypoints to guide the correct matching in OT. In this paper, we propose a novel KeyPoint-Guided model by ReLation preservation (KPG-RL) that searches for the optimal matching (*i.e.*, transport plan) guided by the keypoints in OT. KPG-RL exploits a mask-based constraint of the transport plan to preserve the matching of keypoint pairs in transport, and guides the matching by relation of each data point to the keypoints. The KPG-RL is developed in both balanced and unbalanced/partial transport settings in Kantorovich and Gromov-Wasserstein formulations. Moreover, we deduce the dual formulation of χ^2-regularized KPG-RL model, from which we learn the transport based on deep learning techniques, scaling better to larger numbers of data. With the learned transport plan, two novel neural transport strategies, named manifold barycentric projection and manifold sampling, are developed to transport source data to the target domain data manifold. As applications, we apply the proposed approach to the heterogeneous domain adaptation, multi-omic single-cell alignment, and image-to-image translation. Experiments verified the effectiveness of our approach.

*. Jian Sun is the corresponding author.



(a) Samples (b) KP

(c) GW (d) KPG-RL (ours)

Figure 1: (a) Positive (cross) and negative (circle) samples of source (in blue) and target (in green) distributions. (b) KP distorts the data structure, leading to mismatch of some positive and negative samples. (c) GW model better preserves the data structure, but completely mismatches the samples. (d) Our proposed KPG-RL model utilizes some keypoints (red pairs) to guide the transport and produce correct matching.

Keywords: Keypoint-guided optimal transport, relation preservation, masked plan, neural transport, heterogeneous domain adaptation, image-to-image translation, multi-omic single-cell alignment

1. Introduction

Optimal Transport (OT) (Villani, 2009) is a mathematical tool for distribution alignment, mass transport, *etc.* OT has gained increasing attention in the machine learning community. OT aims to derive a transport map or plan between a source and a target distribution, such that the transport cost is minimized. Due to its capacity to exploit the geometric property of data, OT has been employed in many applications, *e.g.*, computer vision (Bonneel et al., 2011; Solomon et al., 2015), natural language processing (Kusner et al., 2015; Huang et al., 2016; Alvarez-Melis and Jaakkola, 2018; Yurochkin et al., 2019), generative adversarial

network (Arjovsky et al., 2017), domain adaptation (Courty et al., 2017), clustering (Ho et al., 2017; Huynh et al., 2021), anomaly detection (Tong et al., 2022), dataset comparison (Alvarez-Melis and Fusi, 2020), *etc.* OT has two typical formulations, *i.e.*, Monge's formulation (Monge, 1781) and Kantorovich's formulation (Kantorovich, 1942). Kantorovich's formulation (Kantorovich, 1942) relaxes Monge's formulation and attracts broader studies in applications. We mainly deal with Kantorovich's formulation in this paper. The original OT model, *i.e.*, the Kantorovich Problem (Kantorovich, 1942) (KP), is a linear program that is computationally expensive. The entropy-regularized OT (Cuturi, 2013; Dessein et al., 2018) introduces the entropy as regularization to the OT model, which is solved by the computationally cheaper Sinkhorn-Knopp algorithm (Sinkhorn and Knopp, 1967; Lin et al., 2022), allowing the use of automatic differentiation (Genevay et al., 2018). KP transports all the mass of source distribution to exactly match the mass of target distribution. But in some cases, only partial mass of source and target distributions should be matched, *e.g.*, the source or target samples contain outliers. To overcome this limitation, the partial OT (Guittet, 2002; Figalli, 2010; Caffarelli and McCann, 2010; Chapel et al., 2020), unbalanced OT (Chizat et al., 2018; Liero et al., 2018), and robust OT (Balaji et al., 2020; Mukherjee et al., 2021; Le et al., 2021a) models are presented, which allow to only transport partial mass of distribution. Another extension of KP is the Gromov-Wasserstein (GW) model (Sturm, 2006; Mémoli, 2011) that computes the distance between metrics defined within each domain rather than between samples across domains as in KP.

In most of the above OT models, the main criterion for the optimal transport plan/matching is the minimization of total transport distance or distortion over all samples. Though having achieved promising results in many applications, without any additional guidance, the OT models may lead to incorrect matching or undesired alignment of samples. Figures 1(b) and 1(c) illustrate examples of incorrect matching produced by KP and GW models. In real applications, it is reasonable to annotate some paired keypoints across domains for guiding the matching in OT. For instance, in non-rigid point set or image registration (Myronenko and Song, 2010; Crum et al., 2004), a few keypoint pairs that should be matched are annotated in two point sets/images. In semi-supervised or heterogeneous domain adaptation (Saito et al., 2019; Tsai et al., 2016), along with a large amount of labeled source domain data, there are a few labeled target domain data available, that could be directly taken as keypoints. Therefore, it is valuable and important to investigate how to take advantage of those keypoints to guide the correct matching in OT. Figure 1(d) shows an example that if the guidance of a few keypoints is properly imposed, the correctness of matching can be improved.

In this paper, we propose a novel KeyPoint-Guided model by ReLation preservation (KPG-RL) for leveraging the annotated keypoints to guide the matching in OT. The KPG-RL model preserves the matching of keypoint pairs in OT using a mask-based constraint of the transport plan, and enforce the matching of the data points near to paired keypoints across domains by aligning the relation of each data point to the keypoints. KPG-RL is applicable even when distributions lie in different spaces. The above mask and relation-based techniques are employed in KP and GW to impose the keypoint guidance, developing keypoint-guided KP and GW. We show that the keypoint-guided KP and GW respectively provide a metric and divergence between distributions. To handle the applications where only partial mass should be transported, we extend the KPG-RL model to both partial OT and unbalanced OT settings in KP and GW formulations, and develop corresponding solution algorithms.

Meanwhile, we present the dual formulation of χ^2-regularized KPG-RL model, enabling us to learn the transport plan using mini-batch-based deep learning techniques, scaling better to larger numbers of data points. With the learned transport plan, we propose two novel neural transport strategies, named manifold barycentric projection (dubbed KPG-RL-MBP) and manifold sampling (dubbed KPG-RL-MSP), to transport source data to the target domain data manifold. KPG-RL-MBP transports the source samples to the barycenter of the conditional transport plan constrained into the target data manifold, and KPG-RL-MSP aims to sample data on the target data manifold from the conditional transport plan implicitly given the source samples.

As applications, we apply the proposed KPG-RL to the heterogeneous domain adaptation (HDA) (Tsai et al., 2016), multi-omic single-cell alignment (Demetci et al., 2022), and image-to-image (I2I) translation (Isola et al., 2017) tasks. 1) *HDA* is a transfer learning task that aims to transfer the knowledge of large amounts of labeled source domain data to the target domain where a few labeled and larger amounts of unlabeled data are available for training. The "heterogeneity" implies that the source and target domain data are in heterogeneous feature spaces, *e.g.*, generated by different deep networks. This heterogeneity poses a major obstacle in adapting the source-trained model to the target domain. We take the labeled target domain data and source class centers as keypoints and transport source domain data to the target domain by our KPG-RL model (or partial KPG-RL model). Upon the transported source domain data and the labeled target domain data, a classification model is trained that is transferable to the target domain. 2) For *Multi-Omic Single-Cell Alignment* task, the goal is to match multiple modalities of cells, where we only observe the multiple modalities simultaneously for a bunch of cells, which is of great practical importance in biological data analysis. In this task, we take the setting that a few matched cross-modality cells and a larger number of unmatched cells of each modality are provided. We take the matched cells as keypoints. Considering that the cell types in different modalities are often unbalanced, we apply our partial and unbalanced KPG-RL models to search for the matching of different cell modalities. 3) The *I2I Translation* task is for evaluating our neural transport strategies, *i.e.*, KPG-RL-MBP and KPG-RL-MSP. In I2I translation task, we are given the images of two distinct domains with different styles, objects, *etc.* We consider the task setting that a large number of unpaired and a few paired images across the source and target domains are given in training. The goal is to transport the source images to the target domain using the paired images as guidance. We take the paired images as keypoints and apply the proposed KPG-RL-MBP and KPG-RL-MSP to transport the source images to the target domain. Extensive experimental results in the above three tasks demonstrated the effectiveness of our approach to real data.

Our contributions are summarized as follows:

- We propose the novel KPG-RL model to leverage the given paired keypoints to guide the correct matching in OT. KPG-RL employs a mask-based constraint on the transport plan to enforce the matching of keypoint pairs, and impose the guidance of keypoints ueing the relation of each data point to the keypoints.

- We develop the KPG-LR in both balanced and unbalanced/partial OT settings for both Kantorovich and Gromov-Wasserstein formulations, for which the solution algorithms are devised and the metric properties are analyzed.

- We deduce the dual formulation of the KPG-RL model, based on which two neural transport strategies, *i.e.*, manifold barycentric projection and manifold sampling, are proposed to transport the source data to the target domain data manifold.
- We apply the proposed method to heterogeneous domain adaptation, multi-omic single-cell alignment, and image-to-image translation tasks. Extensive experiments verified the effectiveness of our approach.

This paper is extended from our conference version (Gu et al., 2022). In the conference version (Gu et al., 2022), we developed the keypoint-guided OT models in balanced setting for Kantorovich and Gromov-Wasserstein formulations, and in partial transport setting for Kantorovich formulation, which are evaluated in HDA application. The major extensions of this journal paper over conference version are the neural transport strategies based on deduced dual formulation of KPG-RL, the unbalanced keypoint-guided Kantorovich and Gromov-Wasserstein models, and applications to multi-omic single-cell alignment and I2I translation. We summarize additional contributions as follows: (1) We develop the keypoint-guided unbalanced OT in both Kantorovich and Gromov-Wasserstein formulations, solved by generalized Sinkhorn iteration. Meanwhile, the keypoint-guided partial OT is extended to the Gromov-Wasserstein model. (2) We present the dual formulation for the χ^2-regularized KPG-RL model, which is an unconstrained optimization problem. To solve the dual problem, we parameterize the potentials using neural networks that are trained using mini-batch-based optimization algorithms. The transport plan is given by the strong duality using the learned potentials. (3) With the learned transport plan, we propose two neural transport strategies, namely manifold barycentric projection and manifold sampling, to transport source data to the target data manifold. (4) We additionally conduct experiments of image-to-image translation task on MNIST and real natural animal data, and multi-omic single-cell alignment task on real cell data. Extensive experimental results demonstrate the effectiveness of the additionally proposed techniques.

In the following sections, we discuss the related works in Sect. 2, and introduce the background of OT in Sect. 3. In Sect. 4, we detail our KPG-RL model with extensions to Kantorovich Problem and Gromov-Wasserstein model in partial and unbalanced OT settings. In Sect. 5, we elaborate on the dual KPG-RL model with the neural transport strategies of manifold barycentric projection and manifold sampling. In Sect. 6, we apply our approach to HDA, multi-omic single-cell alignment and I2I translation. Section 7 concludes this paper.

Notations. $\Sigma^m = \{\boldsymbol{p} \in \mathbb{R}_+^{m+1} | \sum_i p_i = 1\}$ is the probability simplex. $\langle \cdot, \cdot \rangle_F$ is the Frobenius dot product of two matrices. $\odot$ stands for the Hadamard product. $\mathbb{1}_m$ denotes m-dimensional all-one vector. $\mathbb{1}_{m \times n}$ denotes the $m \times n$ matrix with all-one elements. $\pi_{i,:}$ and $\pi_{:,j}$ are respectively the i-th row and j-th column of matrix π. We use bolded lowercase letters to denote vectors, and corresponding unbolded letters to denote their elements, *e.g.*, for $\boldsymbol{p} \in \mathbb{R}^m$, $\boldsymbol{p} = (p_1, p_2, \cdots, p_m)$. In the absence of ambiguity, we also denote empirical distributions using bolded lowercase letters. We finally denote $[n] = \{1, 2, \cdots, n\}$.

2. Related Works

We review below the most related OT models, HDA methods, multi-omic single-cell alignment methods, and I2I translation approaches to our work.

OT models. The GW model (Mémoli, 2011) seeks a "distance-preserving" transport plan such that the distance between transported points in the target domain is the same as the distance between the original points in source domain. Our KPG-RL model aims to use keypoint pairs to guide the matching (*i.e.*, transport plan) in OT by preserving the relation of each point to the keypoints. Our "relation-preserving" scheme preserves the relation of data *w.r.t.* the given keypoints, different from the pairwise distance-preserving constraint in GW. We experimentally verified the effectiveness of the relation-preserving scheme for introducing the guidance of keypoints in OT. From the computational point of view, GW is a non-convex quadratic program, while KPG-RL is a linear program. Lin et al. (2021) use the anchors to encourage clustering of data and to impose rank constraints on the transport plan to improve its robustness to outliers. The "anchors" in (Lin et al., 2021) are intermediate points in computation for improving robustness, different from the "keypoints" in this paper, which are the annotated paired data for guiding the matching in OT. FlowAlign (Le et al., 2021b) and AEAW (Sato et al., 2020) aim to reduce the computation of comparison of probability measures in heterogeneous spaces by aligning/preserving the distance between data points to anchor points in each space, sharing similar spirits with our method in methodology. FlowAlign (Le et al., 2021b) considers the tree metric space and aligns flows (*i.e.*, tree distances) from a root/anchor to other nodes to find the transport plan. AEAW (Sato et al., 2020) represents each data point as its distance to all the other points (anchors), based on which the energy and Wasserstein distances are defined. Different from FlowAlign (Le et al., 2021b) and AEAW (Sato et al., 2020), our method aims to utilize a few paired keypoints annotated by humans to guide correct transport, which is realized by preserving the relation scores (defined in Sect. 4.2) of each data point to keypoints.

Hierarchical OT (Yurochkin et al., 2019; Lee et al., 2019; Xu et al., 2020) transports points by dividing them into some subgroups and then derives the transport of these subgroups using OT. Different in goal and methodology from Hierarchical OT, we impose the guidance of keypoints for pursuing correct matching in OT by preserving the relation to the keypoints. We do not explicitly divide the points into subgroups, and there is no hierarchy in our method. Courty et al. (2017) constrain the cost function to encourage the matching of labeled data across source and target domains that share the same class labels for domain adaptation. They use the Laplacian regularization to preserve the data structure. Differently, we explicitly model the guidance of keypoints matching to the other data points in our OT formulation. The matching of paired keypoints is enforced by our mask-based constraint on the transport plan. Zhang et al. (2022) propose Masked OT model as a regularization term to preserve the local feature invariances between fine-tuned and pretrained graph neural network (GNN) for the fine-tuning of GNNs. Though the Masked OT (Zhang et al., 2022) shares similar spirits to the mask-based modeling in our method, our main contribution is the relation preservation for imposing the guidance of keypoints, different from (Zhang et al., 2022). For our mask-based modeling, it is utilized to impose the matching of keypoints with theoretical guarantee. While the mask in (Zhang et al., 2022) aims to preserve the local information of pretrained models. The motivation and design of our mask are different from those in (Zhang et al., 2022).

Heterogeneous domain adaptation. HDA methods could be roughly categorized into cross-domain mapping and common subspace learning methods. The cross-domain mapping

approaches (Tsai et al., 2016; Yan et al., 2018; Shen and Guo, 2018; Mozafari and Jamzad, 2016; Zhou et al., 2019b) learn a transform to map the source features or model parameters to target domain to achieve adaptation. The common subspace learning approaches (Wang and Mahadevan, 2011; Wu et al., 2021; Yan et al., 2017; Zhou et al., 2019a; Wang and Breckon, 2022; Fang et al., 2023) learn domain-specific projections to map source and target domain data into a common subspace such that their distributions are aligned. Our method for HDA belongs to the first category, and could be mostly related to (Yan et al., 2018). The method in (Yan et al., 2018) transports source samples to target domain using GW model regularized by the distance between the center of transported source samples and the center of labeled target samples having the same class labels. Different from (Yan et al., 2018), we take each labeled target data and its corresponding source class center as a keypoint pair and preserve the relation of each data point to the set of keypoints when conducting OT.

Multi-omic single-cell alignment. Biological data translation/alignment has achieved much attention recently (Tong et al., 2024a,b; Pariset et al., 2024; Uscidda and Cuturi, 2023; Demetci et al., 2022). The multi-omic single-cell alignment aims to match cells of different modalities. Previous multi-omic single-cell alignment methods (Demetci et al., 2022; Cao et al., 2021; Singh et al., 2023) often align cells based on partial/unbalanced Gromov-Wasserstein models (Cao et al., 2020, 2021; Demetci et al., 2022) or GAN (Singh et al., 2023), in an unsupervised manner. This paper handles a semi-supervised setting that a few matched cross-modality cells are provided along with the unmatched cells. We take the matched cells as keypoints, and apply our proposed keypoint-guided partial and unbalanced Gromov-Wasserstein models to find the matching of cells. We show in experiments that the matching performance can be improved by our approach, given a few keypoint pairs.

Image-to-image translation. In the earlier supervised I2I translation works (Isola et al., 2017; Wang et al., 2018; Zhu et al., 2017b; Zhang et al., 2020; Bansal et al., 2018), researchers use many aligned cross-domain image pairs to obtain the translation model that translates the source images to the desired target images. However, training supervised translation is not very practical because of the difficulty and high cost of acquiring these large, paired training data in real-world tasks. Unsupervised I2I translation methods (Zhu et al., 2017a; Kim et al., 2017; Yi et al., 2017; Choi et al., 2021; Meng et al., 2022; Albergo et al., 2023; Bortoli et al., 2021; Chen et al., 2022; Shi et al., 2023; Pooladian et al., 2023; Tong et al., 2024b,a) use two large but unpaired sets of training images to convert images between representations. Though promising, the unsupervised methods require additional knowledge, *e.g.*, domain-common knowledge (Choi et al., 2021), to guide the desired translation. Semi-supervised I2I translation approach (Mustafa and Mantiuk, 2020) leverages source images alongside a few source-target aligned image pairs for training, reducing the cost of human labeling or expert guidance in supervised I2I. In this paper, we tackle the semi-paired I2I translation task in which a large number of unpaired source and target images, and a few source-target aligned image pairs are available for training, because obtaining unlabeled target images could be reasonable in many applications. We take the given a few source-target aligned image pairs as keypoints and utilize our proposed KPG-RL-MBP and KPG-RL-MSP to translate the source images to the target domain. In methodology, the diffusion or flow approaches (Shi et al., 2023; Pooladian et al., 2023; Tong et al., 2024a,b) based on Schrödinger bridges or conventional OT are mostly related to our proposed KPG-RL-MBP and KPG-RL-MSP.

Differently, KPG-RL-MBP and KPG-RL-MSP are implemented using adversarial training based on our keypoint-guided OT plan. We also use our keypoint-guided OT model to replace the conventional OT in (Pooladian et al., 2023; Tong et al., 2024a,b) to guide flow matching, achieving improved performance.

3. Background on Optimal Transport

As background to our approach, we reintroduce the Kantorovich problem, partial and unbalanced OT models, and Gromov-Wassrstein model in this section.

Kantorovich Problem (KP). Optimal transport considers two sets of data points, *i.e.*, the source data $\boldsymbol{X} = \{x_i\}_{i=1}^m$ and the target data $\boldsymbol{Y} = \{y_j\}_{j=1}^n$, of which the empirical distributions are $\boldsymbol{p} = \sum_{i=1}^m p_i \delta_{x_i}$ and $\boldsymbol{q} = \sum_{j=1}^n q_j \delta_{y_j}$. With a slight abuse of notations, we also denote $\boldsymbol{p} = (p_1, p_2, \cdots, p_m)^\top \in \Sigma^{m-1}$ and $\boldsymbol{q} = (q_1, q_2, \cdots, q_n)^\top \in \Sigma^{n-1}$ as the mass supported on $\boldsymbol{X}$ and $\boldsymbol{Y}$, respectively. We define the cost matrix between $\boldsymbol{X}$ and $\boldsymbol{Y}$ as $C = (C_{i,j}) \in \mathbb{R}^{m \times n}$ with $C_{i,j} = c(x_i, y_j)$, where c is a cost function, which is set to the squared L_2-distance between x_i and y_j in our experiments. OT aims to optimally transport $\boldsymbol{p}$ towards $\boldsymbol{q}$ at the smallest cost, formulated as the following Kantorovich Problem (KP):

$$\min_{\pi \in \Pi(\boldsymbol{p}, \boldsymbol{q})} L_{kp}(\pi) \triangleq \langle \pi, C \rangle_F, \text{ s.t. } \Pi(\boldsymbol{p}, \boldsymbol{q}) = \{\pi \in \mathbb{R}_+^{m \times n} | \pi \mathbb{1}_n = \boldsymbol{p}, \pi^\top \mathbb{1}_m = \boldsymbol{q}\}. \tag{1}$$

When c is taken as a distance metric (*aka.* ground metric), the minimum value of the objective function in Eq. (1) is a distance between $\boldsymbol{p}$ and $\boldsymbol{q}$, named Wasserstein distance.

Partial OT model. The KP in Eq. (1) takes the mass preserving assumption that all the mass of $\boldsymbol{p}$ should be transported to exactly match the mass of $\boldsymbol{q}$. In many applications, only partial mass should be transported. The partial OT model (Figalli, 2010; Caffarelli and McCann, 2010) seeks the minimal cost of transporting only s unit mass from $\boldsymbol{p}$ to $\boldsymbol{q}$, where $0 \leqslant s \leqslant \min(\|\boldsymbol{p}\|_1, \|\boldsymbol{q}\|_1)$, formulated as

$$\min_{\pi \in \Pi^s(\boldsymbol{p}, \boldsymbol{q})} L_{kp}(\pi), \text{ s.t. } \Pi^s(\boldsymbol{p}, \boldsymbol{q}) = \{\pi \in \mathbb{R}_+^{m \times n} | \pi \mathbb{1}_n \leqslant \boldsymbol{p}, \pi^\top \mathbb{1}_m \leqslant \boldsymbol{q}, \mathbb{1}_m^\top \pi \mathbb{1}_n = s\}. \tag{2}$$

Note that in partial OT, the total mass $\|\boldsymbol{p}\|_1$ and $\|\boldsymbol{q}\|_1$ are not necessarily equal.

Unbalanced OT model. Unbalanced OT (Liero et al., 2018) allows the variation of total mass during transport. The unbalanced OT model relaxes the constraints in KP (Eq. (2)) as a regularization term added to the objective function, formulated as

$$\min_{\pi \in \mathbb{R}_+^{m \times n}} L_{kp}(\pi) + \mu \mathcal{D}(\pi \mathbb{1}_n | \boldsymbol{p}) + \mu \mathcal{D}(\pi^\top \mathbb{1}_m | \boldsymbol{q}), \tag{3}$$

where $\mathcal{D}$ is a distribution divergence or distance. Without loss of generality, we choose the KL-divergence in this paper, *i.e.*, $\mathcal{D}(\boldsymbol{p}|\boldsymbol{q}) = \sum_i p_i \log\left(\frac{p_i}{q_i}\right)$. μ is the regularization coefficient.

Gromov-Wasserstein (GW) model. The GW (Mémoli, 2011) model minimizes the distortion when transporting the whole set of points from one space to another. The GW model relies on the intra-domain distance of source domain as $C^s = (C^s_{i,k}) \in \mathbb{R}^{m \times m}$ and target domain as $C^t = (C^t_{j,l}) \in \mathbb{R}^{n \times n}$, where $C^s_{i,k}$ is the distance between source domain data x_i

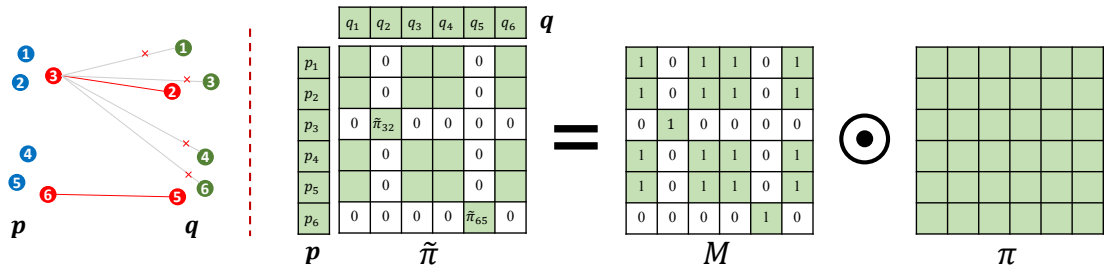


Figure 2: Example of modeling the matching of keypoints (red) using mask. The left part illustrates the matching of data points with keypoint pairs $\mathcal{K} = \{(3,2),(6,5)\}$. To preserve the matching of keypoint index pair (i,j), *e.g.*, $(3,2)$, the i-th row and j-th column of transport plan $\tilde{\pi}$ must be zeros except $\tilde{\pi}_{i,j}$. Therefore, we can model $\tilde{\pi}$ by $\tilde{\pi} = M \odot \pi$, where M is the mask determined by the keypoints, and π is to be optimized.

and x_k, and $C^t_{j,l}$ is the distance between target domain data y_j and y_l. The GW model is given by

$$\min_{\pi \in \Pi(\boldsymbol{p},\boldsymbol{q})} L_{gw}(\pi) \triangleq \sum_{i,k=1}^{m} \sum_{j,l=1}^{n} \pi_{i,j}\pi_{k,l}|C^s_{i,k} - C^t_{j,l}|^2. \tag{4}$$

By replacing L_{kp} with L_{gw} in Eqs. (2) and (3), we obtain the partial and unbalanced Gromov-Wasserstein models, respectively.

We next build our keypoint-guided OT in both balanced (with mass preserving assumption) and partial/unbalanced transport settings in the formulations of KP and GW.

4. Keypoint-Guided Optimal Transport by Relation Preservation

This section details our proposed keypoint-guided model that leverages the keypoints to guide the matching in OT. The guidance is imposed by preserving the matching of keypoint pairs and the relation of each data point to the keypoints. We next discuss how to preserve the matching of keypoints, introduce the modeling of relation, present the keypoint-guided OT models, and discuss extension to KP and GW models in both balanced and unbalanced/partial transport settings. All the proofs of theorems or propositions are given in Appendix.

4.1 Preservation of Matching of Keypoints in Transport

We denote the set of keypoint index pairs as $\mathcal{K} = \{(i_u, j_u)\}_{u=1}^{U}$ with U denoting the number of paired keypoints. We respectively denote $\mathcal{I} = \{i_u\}_{u=1}^{U}$ and $\mathcal{J} = \{j_u\}_{u=1}^{U}$ as the sets of source and target keypoint indexes. For the example illustrated in Fig. 2, $\mathcal{K} = \{(3,2),(6,5)\}, \mathcal{I} = \{3,6\}$, and $\mathcal{J} = \{2,5\}$. To impose the guidance of these keypoints with indexes in $\mathcal{K}$ in deriving the transport plan in OT, we first guarantee the exact matching of the keypoint pairs. As illustrated in Fig. 2, we preserve the matching of keypoints in transport using a mask-based constraint of the transport plan, which is motivated by the

following observation. If the paired keypoints $(i,j) \in \mathcal{K}$ are matched, the optimal transport plan $\tilde{\pi}$ satisfies that the i-th row and j-th column of $\tilde{\pi}$ must be zeros except $\tilde{\pi}_{i,j}$, which means that the all mass of source keypoint x_i must be transported to target keypoint y_j and y_j can only receive the mass from x_i. For the example in Fig. 2, the 3-th row and 2-th column are zeros except that $\tilde{\pi}_{3,2} > 0$. This sparsity of $\tilde{\pi}$ motivates us to model it as the Hadamard product of a mask matrix $M = (M_{i,j}) \in \mathbb{R}^{m \times n}$ and a matrix $\pi \in \mathbb{R}_+^{m \times n}$ with positive entries, *i.e.*,

$$\tilde{\pi} = M \odot \pi, \text{ with } \tilde{\pi}_{i,j} = M_{i,j}\pi_{i,j}. \tag{5}$$

Since $\pi_{i,j}$ is unconstrained when $M_{i,j} = 0$, we enforce $\pi_{i,j} = 0$ in these cases to prevent ill-posedness. We define the admissible solution set for our keypoint-guided OT model as

$$\Pi(\boldsymbol{p}, \boldsymbol{q}; M) = \{\pi \in \mathbb{R}_+^{m \times n} | (M \odot \pi)\mathbb{1}_n = \boldsymbol{p}, (M \odot \pi)^\top \mathbb{1}_m = \boldsymbol{q}, (\mathbb{1}_{m \times n} - M) \odot \pi = 0\}. \tag{6}$$

The entry $M_{i,j}$ of the mask matrix M is set to 0 if $\tilde{\pi}_{i,j}$ needs to be 0, otherwise $M_{i,j}$ is set to 1. Figure 2 illustrates the mask-based modeling of Eq. (5). M is constructed as in Proposition 1.

Proposition 1 *Suppose that the mask matrix M satisfies that*

$$M_{i,j} = \begin{cases} 1, & \text{if } (i,j) \in \mathcal{K}, \\ 0, & \text{if } i \in \mathcal{I} \text{ and } (i,j) \notin \mathcal{K}, \\ 0, & \text{if } j \in \mathcal{J} \text{ and } (i,j) \notin \mathcal{K}, \\ 1, & \text{otherwise (i.e., } i \notin \mathcal{I} \text{ and } j \notin \mathcal{J}). \end{cases} \tag{7}$$

and $p_i = q_j$ for $(i,j) \in \mathcal{K}$. Then, the transport plan $\tilde{\pi} = M \odot \pi$ with $\pi \in \Pi(\boldsymbol{p}, \boldsymbol{q}; M)$ preserves the matching of paired keypoints with index pairs in $\mathcal{K}$.

Proposition 1 indicates that if $p_i = q_j$ for $(i,j) \in \mathcal{K}$, the matching of keypoint pairs is preserved by the mask-based constraint. For the convenience of understanding to readers, we consider the case that $p_i = q_j$, $\forall (i,j) \in \mathcal{K}$. Note that the mask-based modeling in Eq. (5) is also applicable even for the case that there exist some $(i,j) \in \mathcal{K}$ such that $p_i \neq q_j$. For this case, we shall use a different mask matrix. We provide the discussion on such a more general scenario in Appendix A.

4.2 Modeling the Relation to Keypoints

To use the keypoints to guide the matching in transport, we propose to preserve the relation of each point to the set of keypoints in transport. Figure 3 illustrates the relation within points of each distribution. For the data point $x_k \in \boldsymbol{X}$, its relation score to the keypoint x_{i_u} for $i_u \in \mathcal{I}$ (illustrated by red circle in the left of Fig. 3) is defined as

$$R^s_{k,i_u} = \frac{e^{-C^s_{k,i_u}/\tau}}{\sum_{u'=1}^{U} e^{-C^s_{k,i_{u'}}/\tau}}, \ \forall i_u \in \mathcal{I}, \tag{8}$$

where τ is temperature set as $\tau = \rho * \max_{i,k}\{C^s_{i,k}\}$ in experiments. Similarly, for $y_l \in \boldsymbol{Y}$, its relation score to the keypoint y_{j_u} for $j_u \in \mathcal{J}$ is defined as

$$R^t_{l,j_u} = \frac{e^{-C^t_{l,j_u}/\tau'}}{\sum_{u'=1}^{U} e^{-C^t_{l,j_{u'}}/\tau'}}, \ \forall j_u \in \mathcal{J}, \tag{9}$$

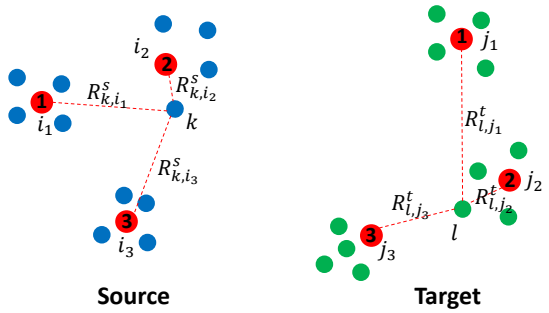


Figure 3: Illustration of relation of each point to the keypoints (red). In the keypoint-guided OT model, the relation should be preserved after transport.

where τ' is temperature set to $\rho * \max_{j,l}\{C^t_{j,l}\}$. In the definition of τ and τ', we first normalize the distances to $[0, 1]$ by dividing by their maximum value, to increase the robustness of the relation score to the scale of the distances.[1] We then introduce ρ to adjust the sharpness further. ρ is a tunable parameter set to 0.1 in our experiments, as 0.1 is a commonly used temperature in the softmax function (Chen et al., 2020; Khosla et al., 2020). We denote $R^s_k = (R^s_{k,i_1}, R^s_{k,i_2}, \cdots, R^s_{k,i_U})$ and $R^t_l = (R^t_{l,j_1}, R^t_{l,j_2}, \cdots, R^t_{l,j_U})$ that represent the relation of data points x_k and y_l to the keypoints in source and target domains, respectively. From the definition of the relation, the cross-domain points near to paired keypoints share a similar relation to the keypoints in corresponding domain. For instance, in Fig. 3, x_k and y_l near to the paired keypoints with indexes (i_2, j_2) have similar relation R^s_k and R^t_l. Meanwhile, if x_k is distant from all the keypoints, R^s_k is close to a uniform probability vector.

4.3 Realizing Keypoint Guidance by Relation Preservation

Based on the relation given above, we define the guiding matrix

$$G = (G_{k,l}) \in \mathbb{R}^{m \times n} \text{ by } G_{k,l} = d(R^s_k, R^t_l), \tag{10}$$

where d measures the dissimilarity of R^s_k and R^t_l. $G_{k,l}$ is smaller if R^s_k and R^t_l are similar. d is taken as the Jensen–Shannon divergence in this paper. We will study the effect of d in experiments. By using the mask-based constraint of transport plan, the *KeyPoint-Guided model by ReLation preservation* (**KPG-RL**) is defined as

$$\min_{\pi \in \Pi(\boldsymbol{p}, \boldsymbol{q}; M)} L_{kpg}(\pi) \triangleq \langle M \odot \pi, G \rangle_F. \tag{11}$$

By the KPG-RL model in Eq. (11), first, the matching of keypoint pairs is enforced by the mask-based constraint of the transport plan. Second, the minimization of the objective function enforces that the optimal transport plan has larger entries in the locations where the entries of G are smaller. Hence the cross-domain points corresponding to these locations

1. We normalize distances by their maximum value to reduce sensitivity to the overall distance scale, which we found effective empirically. Alternatively, one may normalize by a robust statistic such as the mean or median distance. This is an optional implementation choice and does not affect the method formulation.

(*e.g.*, k and l shown in Fig. 3) that are near to paired keypoints tend to be matched. Based on the softmax-based formulations in Eqs. (8) and (9), $d(R_k^s, R_l^t)$ is mainly determined by the relation score to the closest keypoint(s), since relation scores to the distant keypoints are small or close to 0. This implies that the points are mainly guided by the closest keypoints in our KPG-RL model in Eq. (11). For points distant from all keypoints, their corresponding entries of G are similar (close to zero) according to the definition of G. Therefore, the guidance of keypoints to these points is limited. To achieve correct matching of these points, additional information, *e.g.*, point-wise cost or more keypoints, is needed.

Note that given the defined guiding matrix G, one may argue that we can remove the keypoints and the mask in Eq. (10). Then, Eq. (10) has the formulation of KP in Eq. (1), so that the solution algorithms for KP can be directly applied. Finally, the matching of keypoints are added to the optimized transport plan, obtaining the full transport plan. Even though such a strategy could work for searching for the transport plan, the mask-based modeling is necessary in our model. Specifically, the mask-based modeling makes KPG-RL a principled model that automatically enforces the matching of keypoints in searching for the transport plan. With the mask-based modeling, our model provides a principled loss/metric to compare, align, or optimize distributions, as discussed in Sect. 4.4. Meanwhile, the mask-based modeling is a general framework by which we can enforce the other structures in the transport plan, *e.g.*, preserving the matching between groups of points, by defining certain mask matrices. For such cases, it is hard to define the corresponding transport model if removing the mask. We will further discuss the necessity of the mask-based modeling of the transport plan in Sect. 4.4.

Solving KPG-RL. Equation (11) is a linear program and can be solved by linear programming algorithms, *e.g.*, the Simplex algorithm. We give the details for reformulating Eq. (11) as the standard form of linear programming in Appendix B. Since Sinkhorn's algorithm offers a lightspeed computation of the entropy-regularized OT (Cuturi, 2013), a natural question is that, can Sinkhorn's algorithm be applied to the KPG-RL model with entropy regularization? We give the positive answer, and the details of the deduction are given in Appendix B. The iterative formulas are

$$\boldsymbol{u}^{(l+1)} = \frac{\boldsymbol{p}}{K\boldsymbol{v}^l}, \quad \boldsymbol{v}^{(l+1)} = \frac{\boldsymbol{q}}{K^\top \boldsymbol{u}^{(l+1)}}, \tag{12}$$

where $K = M \odot e^{-G/\epsilon}$, ϵ is the coefficient of entropy regularization. The division operator used above is entry-wise. If the iteration stops at $(\boldsymbol{u}, \boldsymbol{v})$, the optimal transport plan is $\mathrm{diag}(\boldsymbol{u})K\mathrm{diag}(\boldsymbol{v})$. We also provide the more stable log-domain version of the Sinkhorn iteration in Appendix B. Note that since the mask-based modeling introduces zeros in K, it is not clear whether the iteration converges. The following proposition shows that for the mask defined in Proposition 1, the iteration converges. We also empirically validate the convergence in experiments, as discussed in Appendix B.

Proposition 2 *Let $M \in \{0,1\}^{m\times n}$ be a fixed mask. Suppose that $\Pi(\boldsymbol{p}, \boldsymbol{q}; M) \neq \emptyset$, and that $G_{ij} < \infty$ for every (i,j) such that $M_{ij} = 1$. Let $K = M \odot \exp(-G/\epsilon)$ and $\epsilon > 0$. Then the transport plans generated by the Sinkhorn iteration in Eq.* (12) *converge to the unique minimizer of the entropy-regularized KPG-RL problem.*

Note that in this KPG-RL model in Eq. (11), the points in $\boldsymbol{X}$ and $\boldsymbol{Y}$ do not necessarily lie in the same space because we only need to compute the distance within each one of $\boldsymbol{X}$

and $\boldsymbol{Y}$. Therefore, the proposed KPG-RL model in Eq. (11) is applicable even when $\boldsymbol{p}$ and $\boldsymbol{q}$ are supported in different spaces. As mentioned in Sect. 1, the GW model is applicable for transport across different spaces. However, GW is a non-convex quadratic program and is often solved by the Frank-Wolfe algorithm, in which a KP-like problem needs to be solved at each iteration by Sinkhorn's algorithm or linear programming. Surprisingly, our KPG-RL model can be directly solved using Sinkhorn's algorithm or linear programming without additional iterations, as discussed above.

4.4 Introducing Keypoint to Kantorovinch and Gromov-Wassestein Models

We now discuss how to impose the keypoint guidance in KP and GW models. To impose the keypoint guidance in KP, we can add $L_{kpg}(\pi)$ as a regularization term to KP, obtaining the following **KPG-RL-KP** model:

$$\min_{\pi\in\Pi(\boldsymbol{p},\boldsymbol{q};M)} \{\alpha L_{kp}(M\odot\pi)+(1-\alpha)L_{kpg}(\pi)=\langle M\odot\pi,\alpha C+(1-\alpha)G\rangle_F\}, \alpha\in(0,1). \quad (13)$$

The KPG-RL-KP can be solved by Sinkhorn's algorithm or linear programming, the same as the solution of KPG-RL model. Similarly, we define the **KPG-RL-GW** model in GW by

$$\min_{\pi\in\Pi(\boldsymbol{p},\boldsymbol{q};M)} \alpha L_{gw}(M\odot\pi)+(1-\alpha)L_{kpg}(\pi), \alpha\in(0,1). \quad (14)$$

The KPG-RL-GW model is solved using the Frank-Wolfe algorithm, of which the main operations are first computing the gradient and then projecting it to the solution set. For the continuity of reading, we give the details of this algorithm in Appendix B. The KPG-RL-KP/KPG-RL-GW models take the advantages of KP/GW and KPG-RL models, and could be helpful, especially when the number of paired keypoints is small. For the coefficient α, we simply set it to 0.5 in experiments.

4.4.1 Theoretical Properties

We next discuss the theoretical properties of KPG-RL-KP and KPG-RL-GW. We will prove that the KPG-RL-KP model provides a proper distance (named "keypoint-guided Wasserstein distance") for distributions supported in the same space, and the KPG-RL-GW model provides a divergence (in the sense of isomorphism) for distributions in distinct spaces, under mild conditions. Since the discrete distributions $\boldsymbol{p}=\frac{1}{m}\sum_i^m \delta_{x_i}$ (resp. $\boldsymbol{q}=\frac{1}{n}\sum_j^n \delta_{y_j}$) are invariant to the permutation of $\{x_i\}_{i=1}^m$ (resp. $\{y_j\}_{j=1}^n$), we assume that any two paired keypoints across domains share the same index. Therefore, the index set of paired keypoints is $\mathcal{K}=\{(i_u,i_u)\}_{u=1}^U$. We denote $\mathcal{P}_{\mathcal{I}}^{\mathcal{X}}$ as the set of discrete probability distributions supported on m points in ground space $\mathcal{X}$ such that all distributions in $\mathcal{P}_{\mathcal{I}}^{\mathcal{X}}$ share the keypoint index set $\mathcal{I}=\{i_u\}_{u=1}^U$.

KPG-RL-KP providing a proper metric. For distributions supported in the same space, we make the following assumption.

Assumption 3 *For any $\boldsymbol{p}$ and $\boldsymbol{q}$ in $\mathcal{P}_{\mathcal{I}}^{\mathcal{X}}$, if $\boldsymbol{p}=\boldsymbol{q}$, then the associated paired keypoints (x_{i_u},y_{i_u}) are equal, i.e., $x_{i_u}=y_{i_u}$, $\forall i_u\in\mathcal{I}$.*

Assumption 3 indicates that if the source and target distributions are identical, the two points in a keypoint pair should be the same point. We then have the following theorem.

Theorem 4 (Keypoint-guided Wasserstein distance) *Suppose c is a proper distance in space $\mathcal{X}$ and d is a proper distance in probability simplex Σ^{U-1}. Let*

$$\mathcal{S}_{krk}(\boldsymbol{p},\boldsymbol{q}) = \min_{\pi\in\Pi(\boldsymbol{p},\boldsymbol{q};M)} \sum_{i,j} M_{i,j}\pi_{i,j}(\alpha C_{i,j} + (1-\alpha)G_{i,j}), \tag{15}$$

where $\alpha \in (0,1)$. Then, for any $\boldsymbol{p}$ and $\boldsymbol{q}$ in $\mathcal{P}_{\mathcal{I}}^{\mathcal{X}}$ satisfying Assumption 3, $\mathcal{S}_{krk}(\boldsymbol{p},\boldsymbol{q})$ is a proper distance between $\boldsymbol{p}$ and $\boldsymbol{q}$.

According to Theorem 4, $\mathcal{S}_{krk}(\boldsymbol{p},\boldsymbol{q})$ is a proper distance. To be consistent with the Wasserstein distance, we name $\mathcal{S}_{krk}(\boldsymbol{p},\boldsymbol{q})$ "keypoint-guided Wasserstein distance".

KPG-RL-GW providing a divergence. For any distribution $\boldsymbol{p} \in \mathcal{P}_{\mathcal{I}}^{\mathcal{X}}$ and $\boldsymbol{q} \in \mathcal{P}_{\mathcal{I}}^{\mathcal{Y}}$, $\boldsymbol{p}$ and $\boldsymbol{q}$ are said to be isomorphic if there exists a bijection $\sigma : [m] \longmapsto [m]$ such that $c(x_i, x_k) = c'(y_{\sigma(i)}, y_{\sigma(k)})$, and $p_i = q_{\sigma(i)}$, where $[m] = \{1, 2, \cdots, m\}$, and c and c' are respectively proper distances in spaces $\mathcal{X}$ and $\mathcal{Y}$. For distributions in different spaces, we make the following assumption.

Assumption 5 *For any $\boldsymbol{p} \in \mathcal{P}_{\mathcal{I}}^{\mathcal{X}}$ and $\boldsymbol{q} \in \mathcal{P}_{\mathcal{I}}^{\mathcal{Y}}$, if $\boldsymbol{p}$ and $\boldsymbol{q}$ are isomorphic with corresponding bijection σ, then the associated keypoint pair (x_{i_u}, y_{i_u}) satisfy $\sigma(i_u) = i_u$, $\forall i_u \in \mathcal{I}$.*

Assumption 5 means that if $\boldsymbol{p}$ and $\boldsymbol{q}$ are isomorphic, σ maps each source keypoint to its paired target keypoint, *i.e.*, $\sigma(i_u) = i_u$. We denote

$$\begin{aligned}\mathcal{S}_{krg}(\boldsymbol{p},\boldsymbol{q}) = \min_{\pi\in\Pi(\boldsymbol{p},\boldsymbol{q};M)} \sum_{i,j} \Big[\alpha\Big(\sum_{k,l}(M\odot\pi)_{i,j}(M\odot\pi)_{k,l}|C^s_{i,k} - C^t_{j,l}|^2\Big) \\ + (1-\alpha)(M\odot\pi)_{i,j}G_{i,j}\Big],\end{aligned} \tag{16}$$

where $\alpha \in (0,1)$. We then have the following theorem.

Theorem 6 *Suppose c and c' are proper distances in spaces $\mathcal{X}$ and $\mathcal{Y}$. Suppose d is a divergence in probability simplex Σ^{U-1}. Then, for any $\boldsymbol{p}$ in $\mathcal{P}_{\mathcal{I}}^{\mathcal{X}}$ and $\boldsymbol{q}$ in $\mathcal{P}_{\mathcal{I}}^{\mathcal{Y}}$ satisfying Assumption 5, we have $\mathcal{S}_{krg}(\boldsymbol{p},\boldsymbol{q}) = 0$ if and only if $\boldsymbol{p}$ and $\boldsymbol{q}$ are isomorphic.*

The proofs of Theorems 4 and 6 are given in Appendix C. With the metric or divergence properties, Theorems 4 and 6 provide reliability for utilizing keypoint-guided KP and GW to align or compare distributions.

Further discussion on the necessity of mask-based modeling. Building on the mask-based modeling, we have proved that our proposed KPG-RL-KP provides a proper distance between distributions, *i.e.*, the keypoint-guided Wasserstein distance. The keypoint-guided Wasserstein distance $\mathcal{S}_{krk}(\boldsymbol{p},\boldsymbol{q})$ can be written as $\mathcal{S}_{krk}(\boldsymbol{p},\boldsymbol{q}) = \sum_{i\in\mathcal{I},j\in\mathcal{J}} M_{i,j}\pi_{i,j}(\alpha C_{i,j} + (1-\alpha)G_{i,j}) + \sum_{i\notin\mathcal{I},j\notin\mathcal{J}} \pi_{i,j}(\alpha C_{i,j} + (1-\alpha)G_{i,j})$. If removing the mask-based modeling and the keypoints in our model, the corresponding model becomes $\mathcal{S}'^{M}_{krk}(\boldsymbol{p},\boldsymbol{q}) = \sum_{i\notin\mathcal{I},j\notin\mathcal{J}} \pi_{i,j}(\alpha C_{i,j} + (1-\alpha)G_{i,j})$, it is no longer a proper metric because $\mathcal{S}'^{M}_{krk}(\boldsymbol{p},\boldsymbol{q}) = 0$ does not imply $\boldsymbol{p} = \boldsymbol{q}$ (the paired keypoints could be different).

Being a proper metric, the keypoint-guided Wasserstein distance $\mathcal{S}_{krk}(\boldsymbol{p},\boldsymbol{q})$ can be utilized to compare distributions or as a principled loss to learn distributions in generative

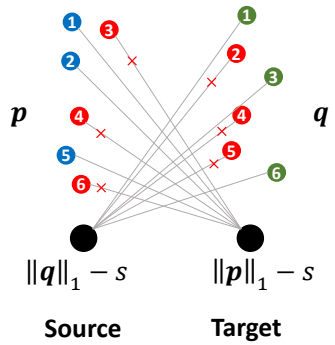


Figure 4: Illustration of dummy points (black circles) for source and target domains.

models (Arjovsky et al., 2017; Genevay et al., 2018), multi-label learning (Frogner et al., 2015), *etc.* When using $\mathcal{S}_{krk}(\boldsymbol{p}, \boldsymbol{q})$ as a distribution distance metric, e.g., in learning distribution transform in a generative model, the first term over the paired keypoints will be imposed as guidance to learn the distribution transform guaranteeing the correspondence of keypoints, while preserving the relations to these keypoints. However, if removing the keypoint correspondence in $\mathcal{S}_{krk}(\boldsymbol{p}, \boldsymbol{q})$ and posing the keypoint correspondence as a follow-up step, the learning of the distribution transform will be separated as two steps, i.e., first learning distribution transform over unpaired data by $\mathcal{S}_{krk}^{/M}$, followed by consistency between paired keypoints. On the contrary, $\mathcal{S}_{krk}$ learns the transform in a principled way. Although practical implementations (e.g., Sinkhorn's algorithm) may involve approximations, the metric structure imposes a rigorous geometric prior at the model level, defining a mathematically well-posed objective and providing a principled foundation for the learning process.

4.5 Extension to Partial and Unbalanced Transport Settings

We now extend the above KPG-RL model to the more practical partial OT and unbalanced OT settings that allows to transport only partial mass.

4.5.1 Extension to Partial Transport Setting

To develop the keypoint-guided partial transport model, we first apply the above mask-based constraint of transport plan to the partial OT model in Eq. (2). We then add more constraints to enforce that all the mass of keypoints is transported to preserve the matching of keypoints. Finally, we define the **partial KPG-RL** model as

$$\min_{\pi \in \Pi^s(\boldsymbol{p}, \boldsymbol{q}; M)} \{L_{kpg}(M \odot \pi) = \langle M \odot \pi, G \rangle_F\}, \tag{17}$$

where $\Pi^s(\boldsymbol{p}, \boldsymbol{q}; M) = \{\pi \in \mathbb{R}_+^{m \times n} | (M \odot \pi)\mathbb{1}_n \leqslant \boldsymbol{p}, (M \odot \pi)^\top \mathbb{1}_m \leqslant \boldsymbol{q}, \mathbb{1}_m^\top (M \odot \pi)\mathbb{1}_n = s; (M \odot \pi)_{i,:}\mathbb{1}_n = p_i, \forall i \in \mathcal{I}; \mathbb{1}_m^\top (M \odot \pi)_{:,j} = q_j, \forall j \in \mathcal{J}; (\mathbb{1}_{m \times n} - M) \odot \pi = 0\}$.

Chapel et al. (2020) propose a compelling method to solve the original partial OT problem in Eq. (2) by transforming the partial OT problem into a KP-like problem. Inspired by Chapel et al. (2020), we add a dummy point with mass $\|\boldsymbol{q}\|_1 - s$ for source domain (the left black circle in Fig. 4) and a dummy point with mass $\|\boldsymbol{p}\|_1 - s$ for target domain (the

right black circle in Fig. 4). We denote $\bar{\boldsymbol{p}} = (\boldsymbol{p}^\top, \|\boldsymbol{q}\|_1 - s)^\top$ and $\bar{\boldsymbol{q}} = (\boldsymbol{q}^\top, \|\boldsymbol{p}\|_1 - s)^\top$. As illustrated in Fig. 4, we aim to design the extended guiding matrix $\bar{G}$ and extended mask matrix $\bar{M}$ such that performing KPG-RL between $\bar{\boldsymbol{p}}$ and $\bar{\boldsymbol{q}}$ will transport $\|\boldsymbol{p}\|_1 - s$ mass from source real data points to the target dummy point and transport $\|\boldsymbol{q}\|_1 - s$ mass from source dummy point to target real data points. As a sequence, only s mass of source and target real data points are matched. Meanwhile, the keypoints should not be matched to the dummy points because they are annotated data to guide the matching. To do this, we extend G, M by

$$\bar{G} = \begin{bmatrix} G & \xi \mathbb{1}_n \\ \xi \mathbb{1}_m^\top & 2\xi + A \end{bmatrix}, \bar{M} = \begin{bmatrix} M & \boldsymbol{a} \\ \boldsymbol{b}^\top & 1 \end{bmatrix},$$

where $A > 0$ and ξ are two fixed scalars, and $\boldsymbol{a} \in \mathbb{R}^m, \boldsymbol{b} \in \mathbb{R}^n$. The element a_i of $\boldsymbol{a}$ is 0 if $i \in \mathcal{I}$, otherwise 1. The element b_j of $\boldsymbol{b}$ is 0 if $j \in \mathcal{J}$, and 1 otherwise. By the following theorem, solving the optimal transport plan of problem (17) boils down to solving the problem $\min_{\bar{\pi} \in \Pi(\bar{\boldsymbol{p}}, \bar{\boldsymbol{q}}; \bar{M})} \langle \bar{M} \odot \bar{\pi}, \bar{G} \rangle_F$.

Theorem 7 *Suppose $A > 0$, $\sum_{i \in \mathcal{I}} p_i < s$, and $\sum_{j \in \mathcal{J}} q_j < s$, then the optimal transport plan $M \odot \pi^*$ of problem (17) is the m-by-n block in the upper left corner of the optimal transport plan $\bar{M} \odot \bar{\pi}^*$ of problem $min_{\bar{\pi} \in \Pi(\bar{\boldsymbol{p}}, \bar{\boldsymbol{q}}; \bar{M})} \langle \bar{M} \odot \bar{\pi}, \bar{G} \rangle_F$.*

According to Theorem 7, problem (17) is solved as follows. We fisrt solve the problem $\min_{\bar{\pi} \in \Pi(\bar{\boldsymbol{p}}, \bar{\boldsymbol{q}}; \bar{M})} \langle \bar{M} \odot \bar{\pi}, \bar{G} \rangle_F$ by linear programming or Sinkhorn's algorithm as discussed in Sect. 4.3, and then take the m-by-n block in the upper left corner of the corresponding optimal transport plan as the optimal transport plan for partial KPG-RL in Eq. (17).

Developing partial keypoint-guided GW model. We now develop the partial keypoint-guided GW model. By replacing the solution set $\Pi(\boldsymbol{p}, \boldsymbol{q}; M)$ in KPG-RL-GW model (Eq. (14)) with the solution set $\Pi^s(\boldsymbol{p}, \boldsymbol{q}; M)$, the partial KPG-RL-GW model is defined as

$$\min_{\pi \in \Pi^s(\boldsymbol{p}, \boldsymbol{q}; M)} \alpha L_{gw}(M \odot \pi) + (1 - \alpha) L_{kpg}(\pi), \alpha \in (0, 1). \tag{18}$$

Inspired by (Chapel et al., 2020), the partial KPG-RL-GW model is solved using the Frank-Wolfe algorithm, which solves a partial KPG-RL-like problem at each iteration. Details are given in Appendix B.

4.5.2 Extension to Unbalanced Transport Setting

To develop the keypoint-guided unbalanced optimal transport, we apply the mask-based modeling of the transport plan and the relation preservation objective function to the unbalanced optimal transport model in Eq. (3). To further ensure the transport of mass of keypoints in unbalanced setting, we employ a spatially varying KL-divergence as regularization. The **unbalanced KPG-RL** model is defined as

$$\min_{\pi \geq 0, (\mathbb{1}_{m \times n} - M) \odot \pi = 0} L_{kpg}(\pi) + \mu \bar{\mathcal{D}}((M \odot \pi) \mathbb{1}_n | \boldsymbol{p}) + \mu \bar{\mathcal{D}}((M \odot \pi)^\top \mathbb{1}_m | \boldsymbol{q}), \tag{19}$$

where $\bar{\mathcal{D}}(\boldsymbol{r}|\boldsymbol{p}) = \sum_i a_i r_i \log \frac{r_i}{p_i}$ is the spatially varying KL-divergence. $a_i = \infty$ if i is an index of keypoints of distribution $\boldsymbol{p}$, otherwise 1. We can see that $r_i = p_i$ (i is an index of keypoints)

as long as $\bar{\mathcal{D}}(\boldsymbol{r}|\boldsymbol{q}) < \infty$. The spatially varying KL-divergence ensures that the keypoints are transported in the unbalanced KPG-RL model. μ is simply set to 1 in experiments.

According to (Séjourné et al., 2023), we solve the unbalanced KPG-RL model with entropy regularization by the generalized Sinkhorn's algorithm. The iteration formulation is given by

$$\boldsymbol{u}^{(l+1)} = \Big(\frac{\boldsymbol{p}}{K\boldsymbol{v}^l}\Big)^{\boldsymbol{\rho}}, \;\; \boldsymbol{v}^{(l+1)} = \Big(\frac{\boldsymbol{q}}{K^\top \boldsymbol{u}^{(l+1)}}\Big)^{\boldsymbol{\rho}}, \tag{20}$$

where the power operator is entry-wise. The element ρ_i of $\boldsymbol{\rho}$ is 1 if $i \in \mathcal{I}$, otherwise $\frac{\mu}{\mu+\epsilon}$, where ϵ is the coefficient of entropy regularization.

Developing unbalanced kepoint-guided GW model. To develop the keypoint-guided unbalanced Gromov-Wasserstein model, *i.e.*, unbalanced KPG-RL-GW, we apply the quadratic divergence as in unbalanced GW (Séjourné et al., 2021) to KPG-RL-GW. The unbalanced KPG-RL-GW is defined by

$$\min_{\pi\geq 0, (\mathbb{1}_{m\times n} - M)\odot\pi = 0} \alpha L_{gw}(\tilde{\pi}) + (1-\alpha) L_{kpg}(\pi) + \mu\bar{\mathcal{D}}^{\otimes}(\tilde{\pi}_1 \times \tilde{\pi}_1 | \boldsymbol{p}\times\boldsymbol{p}) + \mu\bar{\mathcal{D}}^{\otimes}(\tilde{\pi}_2 \times \tilde{\pi}_2 | \boldsymbol{q}\times\boldsymbol{q}) \tag{21}$$

where $\tilde{\pi} = M \odot \pi$, $\tilde{\pi}_1 = \tilde{\pi}\mathbb{1}_n$, and $\tilde{\pi}_2 = \tilde{\pi}^\top \mathbb{1}_m$. $\bar{\mathcal{D}}^{\otimes}(\boldsymbol{r}\times\boldsymbol{r}'|\boldsymbol{p}\times\boldsymbol{p}) = \sum_{i,j} a_{i,j} r_i r'_j \log \frac{r_i r'_j}{p_i p_j}$ is the spatially varying quadratic KL-divergence, where $a_{i,j} = \infty$ if both i and j are indexes of keypoints of distribution $\boldsymbol{p}$, otherwise 1. The unbalanced KPG-RL-GW model is solved similarly to the unbalanced GW model (Séjourné et al., 2021) with entropic regularization, in which an unbalanced KPG-RL-like problem is solved at each iteration. Details are given in Appendix B.

5. Learning Neural Transport for Keypoint-Guided Optimal Transport

In Sect. 4, we have discussed the keypoint-guided optimal transport model in primal formulation (*i.e.*, Eq. (11)) that is solved by linear programming or Sinkhorn's algorithm. Though Sinkhorn's algorithm is promising, it faces challenges in terms of time and memory cost when the number of data points is larger. As mentioned in (Seguy et al., 2018), each iteration of Sinkhorn's algorithm has $\mathcal{O}(n^2)$ complexity. Imagine that if n is a large number, *e.g.*, $n > 10^5$ or even larger values, there are at least 10^{10} variables to be optimized, and the computational time cost would be extremely high. Meanwhile, storing such high-dimensional matrices poses challenges to ordinary computers (see Table 13 in Appendix G).

Considering that deep learning has shown success in tackling large numbers of data points, in this section, we will develop a deep-learning-based approach to estimate the transport plan and map based on the dual formulation of KPG-RL. Inspired by Seguy et al. (2018), we will first develop the dual formulation of the χ^2-regularized KPG-RL model and solve it by training deep neural networks. Based on the learned transport plan, we further propose two novel neural transport strategies, namely Manifold Barycentric Projection (MBP) and Manifold SamPling (MSP), to transport source samples to the target domain.

5.1 Dual Formulation of χ^2-regularized KPG-RL

The χ^2-regularization has been investigated in conventional OT (Seguy et al., 2018; Blondel et al., 2018), of which the duality is an unconstrained optimization problem friendly to

deep learning. We apply the χ^2-regularization to our KPG-RL model. The χ^2-regularized KPG-RL model is given by

$$\min_{\pi \in \Pi(\boldsymbol{p},\boldsymbol{q};M)} \langle M \odot \pi, G \rangle_F + \epsilon \chi^2(M \odot \pi \| \boldsymbol{p} \times \boldsymbol{q}), \tag{22}$$

where $\chi^2(M \odot \pi \| \boldsymbol{p} \times \boldsymbol{q}) = \sum_{i=1}^m \sum_{j=1}^n \left(\frac{M_{i,j}\pi_{i,j}}{p_i q_j} \right)^2 p_i q_j$. The following theorem provides the dual formulation of problem (22).

Theorem 8 *The χ^2-regularized KPG-RL model in Eq. (22) has the dual formmulation*

$$\max_{\phi,\psi} \sum_{i=1}^m \phi(x_i) p_i + \sum_{j=1}^n \psi(y_j) q_j - \frac{1}{4\epsilon} \sum_{i,j} M_{i,j} [(\phi(x_i) + \psi(y_j) - G_{i,j})_+]^2 p_i q_j, \tag{23}$$

where $a_+ = \max\{a, 0\}$. Let ϕ^*, ψ^* be the optimal solution of Eq. (23), then the optimal transport plan $M \odot \pi^*$ of Eq. (22) satisfies

$$(M \odot \pi^*)_{i,j} = \frac{1}{2\epsilon} M_{i,j} (\phi^*(x_i) + \psi^*(y_j) - G_{i,j})_+ p_i q_j. \tag{24}$$

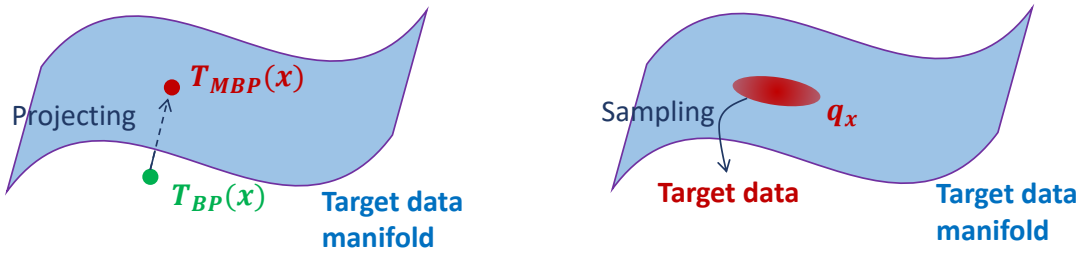
Equation (23) is an unconstrained optimization problem *w.r.t.* continuous functions ϕ, ψ. We parameterize ϕ, ψ using neural networks ϕ_θ, ψ_θ respectively and utilize mini-batch stochastic gradient descent to optimize θ. Specifically, in each iteration, we sequentially sample mini-batch samples from $\boldsymbol{p}, \boldsymbol{q}$, calculate the objective function in Eq. (23) on the mini-batch samples, compute gradient *w.r.t.* θ using backpropagation, and update θ by gradient descent. Such a mini-batch-based optimization process enables the keypoint-guided optimal transport model to scale to the distributions supported on a larger number (m and n) of samples, because we only need a mini-batch of samples to calculate the objective function in each iteration. Once the optimization process is completed, the optimal transport plan is recovered by the strong duality in Eq. (24). It is worth noting that due to the inevitable optimization gap in non-convex network training and the finite capacity of network approximations, the recovered plan $(M \odot \pi^*)$ may not satisfy marginal constraints. Despite this, $(M \odot \pi^*)$ reflects the coupling relationship between source and target domain samples. We next treat the recovered values in Eq. (24) as unnormalized coupling weights to build neural transport strategies for transporting source domain data to target domain.

5.2 Neural Transport Strategies

Section 5.1 offers a neural approach to learn the transport plan for larger numbers of data points. Yet it is unclear how to transport source data to target domain. Seguy et al. (2018) propose the Barycentric Projection (BP) to transport the source data to the target domain, which can be directly applied to our approach using our learned transport plan. The BP (T_{BP}) is given by

$$T_{BP} = \arg\min_{T'} \sum_{i=1}^m \sum_{j=1}^n (M \odot \pi^*)_{i,j} \bar{d}(y_j, T'(x_i)), \tag{25}$$

where $\bar{d}$ is a difference measure. However, for the source sample x, the transported sample $T_{BP}(x)$ by BP may be blurry, as shown in Fig. 11(b). This is mainly because $T_{BP}(x)$ is close to the barycenter under $\bar{d}$.



(a) Manifold barycentric projection (MBP)

(b) Manifold sampling (MSP)

Figure 5: Illustration of our (a) manifold barycentric projection and (b) manifold sampling for transporting source data to the target domain. In (a), the image $T_{BP}(x)$ of barycentric projection is outside the target data manifold and is blurry. The transported data $T_{MBP}(x)$ by our proposed manifold barycentric projection is constrained to the target data manifold. In (b), the transport plan provides a conditional distribution $\boldsymbol{q}_x$ on target data manifold for a given source sample x. Our manifold sampling aims to sample target data from $\boldsymbol{q}_x$.

We analyze that the blur caused by the BP is probably because the projected data $T_{BP}(x)$ may not be in the target data manifold.[2] Intuitively, if $\bar{d}$ can ideally reflect the intrinsic structure of the target data manifold, the projected data $T_{BP}(x)$ could be in the target data manifold. However, such an ideal $\bar{d}$ is difficult to define. To obtain clear transported data, we present two neural transport strategies, namely manifold barycentric projection and manifold sampling, to transport source data to target domain based on learned transport plan. We next discuss the details of the manifold barycentric projection and manifold sampling.

Manifold Barycentric Projection. The goal of Manifold Barycentric Projection (MBP) is to enforce the transported data near to $T_{BP}(x)$ constrained into the target data manifold implicitly. Figure 5(a) illustrates our idea. To achieve this goal, we add a manifold constraint to the BP model in Eq. (25), forming the following model:

$$T_{MBP} = \arg\min_{T} \lambda \sum_{i=1}^{m}\sum_{j=1}^{n} (M \odot \pi^*)_{i,j} \bar{d}(y_j, T(x_i)) + \mathcal{L}_M(T), \tag{26}$$

where the first term enforces the transported data is near to that of the barycentric projection, and the second term encourages the transported data to lie in the target data manifold λ is a hyper-parameter for balancing the importance of the two terms and is set to 10 in experiments. For simplicity, we use the loss of WGAN-GP (Gulrajani et al., 2017) as $\mathcal{L}_M$, because WGAN-GP is often used to model the data manifold (Pandey et al., 2021). $\mathcal{L}_M$ is

2. Machine learning studies often take the "manifold hypothesis" that many high-dimensional real data sets, *e.g.*, natural images, lie in low-dimensional manifolds inside the high-dimensional space (Cayton, 2005; Fefferman et al., 2016).

given by

$$\mathcal{L}_M(T) = \max_D \sum_{i=1}^{m} D(T(x_i))p_i - \sum_{j=1}^{n} D(y_j)q_j + \beta \mathbb{E}_{\hat{x}\sim\hat{\boldsymbol{p}}}(\|\nabla_{\hat{x}} D(\hat{x})\|_2 - 1)^2, \tag{27}$$

where $\hat{\boldsymbol{p}}$ denotes the samples uniformly along lines between pairs of points sampled from the transported source distribution $T_{\#}\boldsymbol{p}$ and target distribution $\boldsymbol{q}$, and D, named discriminator, is a neural network that outputs a scalar.[3] Following Gulrajani et al. (2017), β is set to 10, and the gradient of $\hat{x}$ *w.r.t.* T is stopped. Equation (26) is optimized by the mini-batch stochastic gradient descent/ascent to update T/D, respectively. With the learned T_{MBP}, given the source domain test sample x even outside the training set, we transport it to the target domain by $T_{MBP}(x)$.

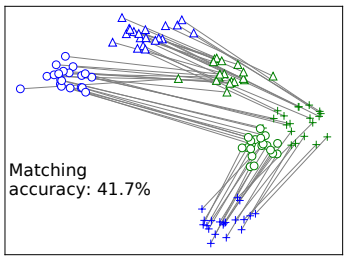
Manifold Sampling. For the Manifold SamPling (MSP), the goal is to sample data from the conditional distribution on the target domain data manifold induced by the transport plan, given the source sample. Specifically, given the learned transport plan $M \odot \pi$ and a source sample x_i, $M \odot \pi$ induces a empirical distribution $\boldsymbol{q}_{x_i}$ in target data manifold conditional on x_i, where $\boldsymbol{q}_{x_i}$ is defined by $\boldsymbol{q}_{x_i} = \frac{1}{\sum_{j'=1}^{n}(M\odot\pi^*)_{i,j'}} \sum_{j=1}^{n}(M \odot \pi^*)_{i,j}\delta_{y_j}$. The main idea of MSP is to sample data from $\boldsymbol{q}_{x_i}$, which is taken as the transported data of x_i, as illustrated in Fig. 5(b). One possible strategy to achieve this goal is to choose a y_j with probability $\frac{1}{\sum_{j'=1}^{n}(M\odot\pi^*)_{i,j'}}(M \odot \pi^*)_{i,j}$ from $\boldsymbol{Y}$. However, such a strategy can not generate new target samples outside the training set $\boldsymbol{Y}$. To address tackle this issue, inspired by GANs (Goodfellow et al., 2014; Gulrajani et al., 2017), we learn a map T_{MSP} mapping source samples along with noise to the target data manifold, which is able to generate target samples outside $\boldsymbol{Y}$. The mathematical formulation for learning T_{MSP} is defined by

$$T_{MSP} = \arg\min_T \max_D \sum_{i=1}^{m} p_i \left(\mathbb{E}_{z\sim\mathcal{N}} D(x_i, T(x_i, z)) - \mathbb{E}_{y\sim\boldsymbol{q}_{x_i}} D(x_i, y)\right) + \beta\mathcal{R}(D), \tag{28}$$

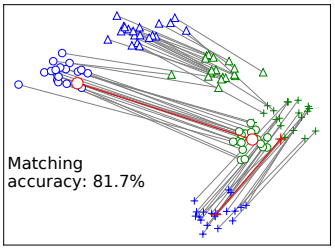
where $\mathcal{N}$ is the standard Gaussian distribution, and $\mathcal{R}(D) = \sum_{i=1}^{m} p_i \mathbb{E}_{\hat{x}\sim\hat{\boldsymbol{p}}}(\|\nabla_{\hat{x}} D(x_i, \hat{x})\|_2 - 1)^2$, where $\hat{\boldsymbol{p}}$ denotes the samples uniformly along lines between pairs of points sampled from $T(x_i, \cdot)_{\#}\mathcal{N}$ and $\boldsymbol{q}_{x_i}$. In Eq. (28), $D(x_i, \cdot)$ is a discriminator specific to x_i, distinguishing the transported samples $T(x_i, z)$ and target sample y of conditional distribution $\boldsymbol{q}_{x_i}$. $T(x_i, \cdot)$ aims to fool $D(x_i, \cdot)$. In training, T/D are updated using mini-batch gradient stochastic descent/ascent. As a result of the adversarial training of T and D, $T(x_i, z)$ will approach a sample from $\boldsymbol{q}_{x_i}$. After training, given a test source sample x even outside of the training set, we randomly sample a Gaussian noise z, and then $T_{MSP}(x_i, z)$ are the transported data.

6. Experiments

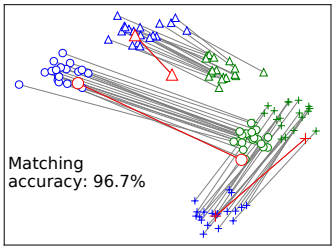
We evaluate our method on toy data, HDA, Multi-omic single-cell alignment, and I2I translation experiments. Codes will be available at `https://github.com/XJTU-XGU/KPG-RL`.



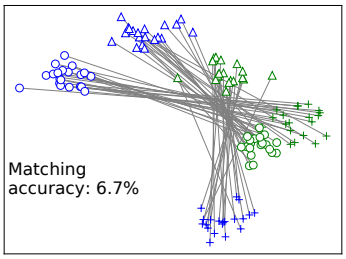
(a) KP



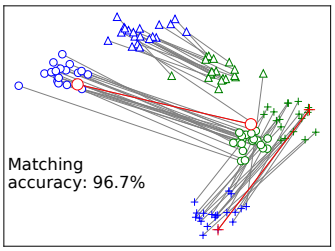
(b) KPG-RL-KP (w/ 2 keypoints)



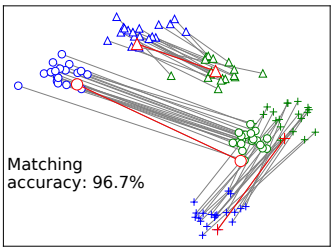
(c) KPG-RL-KP (w/ 3 keypoints)



(d) GW



(e) KPG-RL-GW (w/ 2 keypoints)



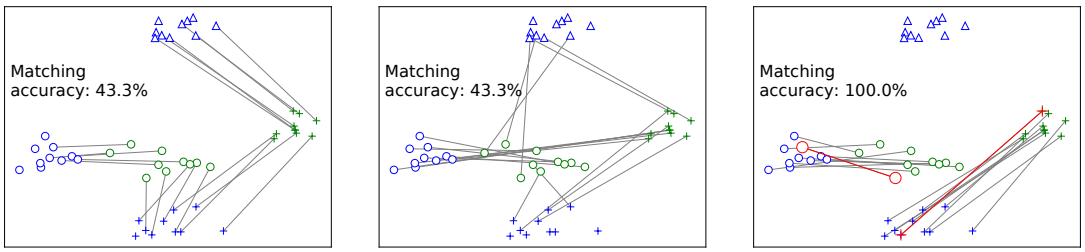
(f) KPG-RL-GW (w/ 3 keypoints)

Figure 6: Matching produced by (a) KP, (b) KPG-RL-KP model given 2 keypoint pairs, (c) KPG-RL-KP model given 3 keypoint pairs, (d) GW model, (e) KPG-RL-GW model given 2 keypoint pairs, and (f) KPG-RL-GW model given 3 keypoint pairs.

6.1 Toy Experiments

Toy experiments for evaluating KPG-RL-KP and KPG-RL-GW. As illustrated in Fig. 6, in this toy data experiment, each of the source (blue) and target (green) distributions is a Gaussian mixture composed of three distinct Gaussian components indicated by different shapes where the same shapes indicate points of the same class. In Fig. 6, we have the following observations. In Fig. 6(a), in the KP model, the points in each component of target distribution are mismatched to points in different classes of source distributions, and only a small fraction of target points are correctly matched to source points belonging to the same class.[4] In Fig. 6(b), given 2 keypoint pairs (from distinct classes), the KPG-RL-KP model apparently improves the correctness of matching.[5] In Fig. 6(c), the KPG-RL-KP model mainly matches the source points to the target points belonging to the same class, thanks to the given 3 keypoint pairs. This suggests that our keypoint-guided model helps recover the correct OT matching by leveraging a few given keypoint pairs. In Figs. 6(d), 6(e) and 6(f),

3. For a random variable $x \sim \boldsymbol{p}$ and a transform T on x, we use $T_{\#}\boldsymbol{p}$ to denote the distribution of $T(x)$.
4. In this experiment, the numbers of source and target points are equal, and the mass of all points (including keypoints) are equal. The linear nature of KPG-RL model in Eq. (11) enables that for any source data point x_i, only one target point y_j in $\boldsymbol{Y}$ takes non-zero entry in the optimal transport plan. Then x_i and y_j are matched and linked by the black line in Fig. 6.
5. A pair of cross-domain points being correctly matched means that they share the same class label. The matching accuracy is the ratio of correctly matched data pairs.



(a) Partial OT (b) Partial GW (c) Partial KPG-RL (ours)

Figure 7: Matching produced by (a) partial OT model, (b) partial GW model, and (c) our proposed partial KPG-RL model.

the proposed KPG-RL-GW model improves the correctness of matching of GW model by leveraging the guidance of given keypoints pairs.

Toy experiments for evaluating partial KPG-RL. Figure 7 illustrates the toy data experiment for evaluating the partial KPG-RL model. In Fig. 7, the source (blue) and target (green) distributions are Gaussian mixtures. The source (resp. target) distribution is composed of three (resp. two) distinct Gaussian components indicated by different shapes where the same shapes indicate the points of the same class. When conducting OT, the source class data represented by "△" should not be transported. The masses of all data points (including keypoints) are set to $1/m$ where m is the number of source data points. Note that the total target mass is less than 1. In Figs. 7(a) and 7(b), we can observe that both the partial-OT model (defined in Eq. (2)) and the partial-GW model (Chapel et al., 2020) wrongly transport some source points of class "△" to target domain and lead to lower matching accuracy. With the guidance of a few keypoints (red pairs), our proposed partial KPG-RL model does not transport the source points of class "△" to target domain and apparently improves the matching accuracy, as in Fig. 7(c).

6.2 Heterogeneous Domain Adaptation

We discuss the experiments of heterogeneous domain adaptation (HDA) in this section, including closed-set and open-set HDA.

6.2.1 Closed-Set Heterogeneous Domain Adaptation

In closed-set HDA (in what follows, we refer to the closed-set HDA by HDA), we are given labeled source data $\{(x_i, t_i)\}_{i=1}^{m}$, a few labeled target data $\{y_j, \bar{t}_j\}_{j=1}^{n_l}$, and unlabeled target domain data $\{y_j\}_{j=n_l+1}^{n}$, where $m \gg n_l$, $n \gg n_l$, x_i and y_j are features, and t_i and $\bar{t}_j$ are respectively the class labels of x_i and y_j. The source and target data share the same label space (the set of classes). The heterogeneity means that x_i and y_j are from different spaces/modalities. The goal of HDA is to train a classification model using the given data to predict the label of unlabeled target domain data, leveraging the knowledge of labeled source domain data. The main challenge is that the domain gap between heterogeneous

source and target distributions supported in distinct spaces hinders the direct employment of the source-trained model in the target domain.

We tackle the problem of HDA using our proposed KPG-RL model as follows. We first transport the source domain data using our KPG-RL model to the target domain. More concretely, in KPG-RL, each labeled target domain sample and the source domain class center of the same class are taken as a matched keypoint pair, implying the source class centers are repeated by k times for k-shot setting. In this experiment, we resample source data such that $m = n - n_l$, and assign uniform mass $\frac{1}{m}$ to all samples, resulting in equal mass for paired keypoints. The KPG-RL model is then performed between empirical distributions of the source domain data along with class centers, and the target domain data (including labeled and unlabeled data). Based on the produced optimal transport plan, we transport the source domain data using the barycentric mapping (Reich, 2013) to the target domain. The barycentric mapping B_π associated to transport plan π is defined by $B_\pi(x_{i_0}) = \frac{1}{\sum_{j=1}^n \pi_{i_0,j}} \sum_{j=1}^n \pi_{i_0,j} y_j$. Note that our MBP (presented in Sect. 5) learns transport map by training deep networks, which may be suitable for applications where we need to transport source samples outside the training set, *e.g.*, I2I translation discussed in Sect. 6.4. For simplicity, we directly adopt the barycentric mapping for the HDA experiments. Finally, we train a kernel SVM on the transported source domain data (using their class label before transport) and the labeled target domain data, which is applied to the target domain test data. In the kernel SVM, we use the Gaussian $k(x, y) = \exp(-\gamma\|x - y\|_2^2)$, where γ is set to the reciprocal of the dimension of x. ϵ is set to 0.005.

We compare our method with the following baseline methods, including 1) "Labeled-target-only" that trains the kernel SVM using the labeled target data; 2) the OT methods of "GW", "HOT", and "TLB" that transport source domain data using barycentric mapping induced by the transport plan of GW (Mémoli, 2011), Hierarchical OT (Lee et al., 2019), and TLB (Mémoli, 2011), and then train the kernel SVM on the transported data and labeled target domain data; 3) the typical HDA methods of "SGW" (Yan et al., 2018) and "STN" (Yao et al., 2019), and the recent HDA methods of "SSAN" (Li et al., 2020), "DDACL" (Yao et al., 2020) and "CDSPP" (Wang and Breckon, 2022). We conduct experiments on Office-31 (Saenko et al., 2010) dataset. On Office-31, we use the DeCAF_6 (Donahue et al., 2014) features and the features extracted by ResNet-50 (He et al., 2016) pretrained on ImageNet (Russakovsky et al., 2015) to respectively build source and target domains for constructing heterogeneous adaptation tasks. In each task, one labeled data (1-shot) for each class is given in the target domain. Note that all the methods use the same training data (including labeled source and target domain data, and unlabeled target domain data) and test data.

Table 1 reports the results on Offce-31 dataset. In Table 1, "KPG-RL (SH)" and "KPG-RL (LP)" denote our entropy-regularized KPG-RL model solved using Sinkhorn's algorithm and our KPG-RL model solved by linear programming, respectively. KPG-RL (SH) achieves marginally better average accuracy than KPG-RL (LP), which could be because the barycentric mapping based on the more dense transport plan optimized by Sinkhorn's algorithm is more robust to incorrect matching. We also observe that KPG-RL-GW achieves slightly lower average accuracy than KPG-RL (SH), and KPG-RL (SH) is more computationally efficient as discussed in Sect. 4. "KPG (w/ dist)" is the approach that imposes the guidance of keypoints using distance preservation in our framework, *i.e.*, R^s_{k,i_u} and R^t_{l,j_u} in Eqs. (8) and (9) are taken as C^s_{k,i_u} and C^t_{l,j_u}, respectively. KPG-RL (SH)

Method	A→A	A→D	A→W	D→A	D→D	D→W	W→A	W→D	W→W	**Avg**
Labeled-target-only	45.3	69.2	67.3	45.3	69.2	67.3	45.3	69.2	67.3	60.6
STN	58.7	84.8	80.0	51.2	91.0	83.8	52.8	95.2	87.4	76.1
SSAN	56.8	89.7	**87.1**	54.2	82.6	**90.3**	50.0	85.8	85.8	75.8
DDACL	44.2	63.2	64.2	43.9	77.4	70.6	39.8	64.5	73.2	60.1
CDSPP	55.5	79.7	76.5	43.2	80.6	84.2	47.4	78.7	84.5	70.0
SGW	49.7	77.7	73.6	49.4	78.4	73.6	48.7	80.3	74.5	67.3
GW	33.6	41.6	35.5	39.7	40.0	31.7	34.8	34.5	29.0	35.6
GW (w/ mask)	41.3	71.6	69.7	41.9	71.2	69.8	40.3	71.6	69.7	60.8
HOT	39.0	44.8	40.0	31.3	52.6	44.8	29.7	60.0	56.5	44.3
HOT (w/ mask)	45.2	60.3	57.4	48.9	63.5	59.2	40.3	67.1	61.4	55.9
TLB	29.4	36.5	43.2	24.5	31.3	51.0	23.6	31.9	49.7	35.7
TLB (w/ mask)	42.5	66.3	64.7	38.5	68.5	65.9	43.1	68.2	67.3	58.3
KPG (w/ dist)	55.2	60.7	71.6	51.3	71.9	77.1	48.7	70.0	77.7	64.9
KPG-RL-GW	58.7	92.9	84.2	57.4	95.5	87.1	55.5	**95.5**	**90.0**	79.6
KPG-RL (LP)	56.5	**93.6**	83.2	**58.1**	94.5	86.8	55.8	95.2	89.7	79.3
KPG-RL (SH)	**60.0**	91.6	83.6	57.4	**95.8**	87.7	**59.1**	95.2	88.4	**79.9**

Table 1: Accuracy on Office-31 for HDA. "A", "W", and "D" are respectively the domains of amazon, webcam, and dslr. "$\cdot \to *$" denotes a heterogeneous adaptation task where $\cdot$ and $*$ are respectively source domain using DeCAF_6 feature and target domain using ResNet-50 feature.

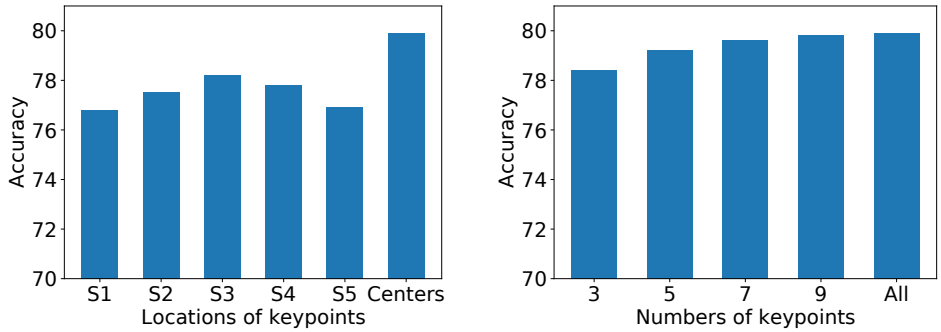
improves the accuracy of KPG (w/ dist) by 15%, implying that our relation-preserving scheme is more effective for imposing the guidance of keypoints than the distance-preserving scheme. Table 1 shows that the results of GW, HOT, and TLB are inferior to those of Labeled-target-only, indicating that GW, HOT, and TLB cause negative transfer. This is reasonable because GW, HOT, and TLB do not take advantage of the guidance of keypoints and can cause wrong matching. We also use our mask-based constraint on the transport plan in GW, HOT, and TLB, of which the corresponding approach are denoted as GW (w/ mask), HOT (w/ mask), TLB (w/ mask). We can observe that with the mask-based constraint, the performances of GW, HOT, and TLB are improved but worse than KPG-RL (SH) and KPG-RL (LP), verifying the importance of the relation preservation scheme in KPG-RL. SGW (Yan et al., 2018) aligns the class centers of labeled target domain data and transported source domain data in the GW model, and realizes positive transfer. Our proposed KPG-RL (SH) outperforms SGW by 12.2%, confirming that our KPG-RL model preserving relation is more effective than SGW preserving class centers for HDA. Compared with the other HDA methods, KPG-RL (SH) achieves the best average accuracy.

6.2.2 Ablation Studies

We next investigate the effect of target/source keypoints, compare different choices of d, and study sensitivity to hyperparameters in closed-set HDA experiment, in this subsection.

Method	1-shot						2-shot						3-shot					
	S1	S2	S3	S4	S5	Avg	S1	S2	S3	S4	S5	Avg	S1	S2	S3	S4	S5	Avg
Labeled-target-only	61.2	58.4	64.1	60.6	58.7	60.6	77.8	76.3	78.3	75.0	74.2	76.3	80.7	76.6	81.0	83.4	82.7	80.9
KPG-RL (SH)	**77.6**	**76.2**	**82.1**	**82.9**	**80.7**	**79.9**	**86.5**	**86.5**	**86.8**	**84.4**	**82.6**	**85.4**	**86.5**	**88.5**	**87.0**	**88.3**	**88.2**	**87.7**

Table 2: Results of methods of Labeled-target-only and KPG given different shots (1, 2, and 3) of labeled target domain data (keypoints) per-class under five distinct samplings (S1,S2,S3,S4, and S5).



(a) (b)

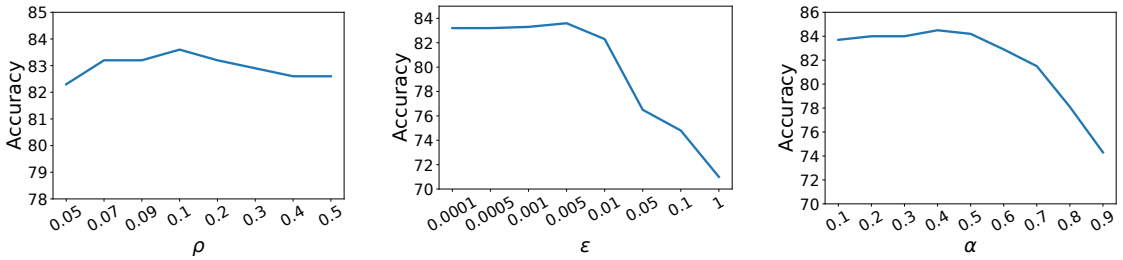
Figure 8: (a) Results for different locations of source keypoints. (b)Results for different numbers of source keypoints.

Effect of target keypoints. The keypoints are important to KPG-RL, since they are used to guide the matching. We show the results with varying keypoints by giving different numbers and samplings of labeled target domain data, in Table 2. It can be observed in Table 2 that under different numbers and samplings of labeled target domain data, our approach KPG-RL (SH) consistently outperforms the baseline Labeled-target-only, achieving positive transfer. We also find that as the number of labeled target domain data increases, the margin between the accuracy of Labeled-target-only and KPG-RL (SH) decreases. This indicates that given more labeled target domain data, the necessity of knowledge transferred from source domain becomes smaller.

Effect of source keypoints. In the paper, the source keypoints are taken as the source class centers. To study the sensitivity to the location of source keypoints, we randomly sample one data point from each class as a keypoint to construct the source keypoints. We run the experiments with five different samplings for constructing the source keypoints (these five runs are denoted as S1, S2, S3, S4, S5, respectively). The results are reported in Fig. 8(a). We can see that using the class center as the keypoints achieves the best results, compared with randomly sampling one data point per class as the keypoints. This may be because the class centers are estimated using all the data of each class, and these centers can better represent each class than a randomly sampled data point of each class. We next study the

Choices of d	A→A	A→D	A→W	D→A	D→D	D→W	W→A	W→D	W→W	**Avg**
KL-ST	59.0	89.7	**83.6**	56.8	95.2	**89.0**	57.7	93.6	88.1	79.2
KL-TS	58.1	89.0	82.3	54.2	93.9	88.1	54.2	93.2	**89.4**	78.0
L_1-distance	57.4	85.8	79.0	58.0	85.8	82.9	58.4	92.6	83.6	75.9
χ^2-distance	52.3	85.8	81.3	53.2	91.3	82.3	52.6	90.3	82.9	74.7
GW	42.0	71.6	70.0	41.6	71.0	69.4	42.3	71.3	70.0	61.0
JS	**60.0**	**91.6**	**83.6**	**57.4**	**95.8**	87.7	**59.1**	**95.2**	88.4	**79.9**

Table 3: Results of different choices of d in HDA experiment on Office-31.



(a) (b) (c)

Figure 9: (a) Sensitivity of KPG-RL to hyper-parameter ρ. (b) Sensitivity of KPG-RL to hyper-parameter ϵ. (c) Sensitivity of KPG-RL-GW to hyper-parameter α. The results are for the Closed-Set HDA task A→W.

sensitivity to the number of source keypoints, of which the results are reported in Fig. 8(b). We randomly sample 3/5/7/9 samples (keypoints) or use all the source samples (keypoints) for each class in the source domain to compute the source class centers, which are paired with labeled target samples for constructing the keypoint pairs. The results in Fig. 8(b) show that as the number of source keypoints increases, the accuracy gradually increases. The best result is obtained when all source samples are used to compute the class centers.

Comparison of different choices for $\boldsymbol{d}$. Since R_k^s and R_l^t are in the probability simplex, it is reasonable to measure their difference by a distribution divergence/distance. The widely used distribution divergences/distances include the KL-divergence, JS-divergence, and Wasserstein distance. The KL-divergence is not symmetric, so we need to determine the order of inputs. For the Wasserstein distance, one should define the ground metric first. A possible strategy is to set the ground metric to 0 if the two keypoints are paired, otherwise 1. Such a ground metric makes the Wasserstein distance equal to the L_1-distance. In this work, d is taken as the JS-divergence. We compare the performance of different choices of d in the experiment of HDA on Office-31, as in Table 3. In Table 3, KL-ST and KL-TS denote the KL-divergence $KL(R_k^s, R_l^t)$ and $KL(R_l^t, R_k^s)$, respectively. GW is the Gromov-Wasserstein distance between R_k^s and R_l^t where the source/target cost is taken as the χ^2-distance of source/target keypoints.

Ratio	1%	2%	3%	5%	10%	15%	20%	30%	40%	50%
Accuracy	87.6	87.7	87.2	87.1	87.2	86.8	86.2	84.4	79.3	69.2

Table 4: Accuracy (%) of our method under different ratios of mismatched keypoint pairs.

We find that the JS-divergence achieves the best performance, compared with KL-ST, KL-TS, L_1-distance, χ^2-distance, and Gromov-Wasserstein.

Sensitivity to hyper-parameters. We show the sensitivity of our method to hyperparameters ρ, ϵ, and α in Fig. 9. ϵ is the coefficient of entropy regularization. ρ is used to set τ and τ' for defining the relation in Eqs. (8) and (9). α is in the KPG-RL-GW model. The best result for ρ is obtained at 0.1. For ϵ, the results are relatively stable in range [0.0001,0.005]. It can be observed that the best value of α is 0.4 in this task, and the results are relatively stable when α ranges in [0.2, 0.5].

Performance under mismatched keypoint pairs. To assess robustness to noisy keypoint correspondences, we randomly corrupt a fraction of matched pairs by keeping the target keypoints fixed and replacing their paired source keypoints with class centers randomly sampled from an incorrect class. The results under 3-shot setting are reported in Table 4. We observe that KPG-RL is relatively stable under moderate mismatch (up to 20%, with accuracy around 87%), and degrades gradually at 30% (84.4%). When the mismatch becomes severe ($\geq 40\%$), the accuracy drops markedly (79.3% at 40% and 69.2% at 50%), indicating that excessive cross-class mismatches can bias the keypoint guidance.

6.2.3 Open-Set Heterogeneous Domain Adaptation

We conduct open-set HDA experiments to evaluate our proposed partial KPG-RL model. The problem setting of open-set HDA is the same as HDA, except that the unlabeled target domain data contain samples of unknown classes absent in the categories of labeled data. The goal of open-set HDA is to correctly classify the common class samples and to detect the unknown class samples. We first consider the case that the proportion η of unknown class samples in the unlabeled target domain data is given. We then extend the approach to the case with unknown η.

Open-set HDA with given $\boldsymbol{\eta}$. To apply the partial KPG-RL model defined in Eq. (17) to open-set HDA, for each labeled target domain data, we take its corresponding source class center to construct a keypoint pair. We then resample the source domain data such that the total number of resampled source domain data and the source keypoints is $m' = (1-\eta)n$. We define the source distribution as $\boldsymbol{p} = \frac{1-\eta}{m'}(\sum_{j=1}^{n_l} \delta_{c_j} + \sum_{j=n_l+1}^{m'} \delta_{x'_j})$, where x'_j is a resampled source domain sample and c_j is the source class center corresponding to the target labeled sample y_j. The target distribution is defined as $\boldsymbol{q} = \frac{1}{n}\sum_{j=1}^{n} \delta_{y_j}$. The partial KPG-RL model is conducted to transport mass from $\boldsymbol{p}$ to $\boldsymbol{q}$ with $s = 1-\eta$. Such a mass construction strategy results in equal mass across corresponding keypoints in the two domains. After transport, the η-fraction unlabeled target data receiving smallest mass from source domain are detected as unknown class, and the rest unlabeled target data are taken as common class ones. Finally,

Method	A→A			A→D			A→W			D→A			D→D		
	OS*	UNK	HOS	OS*	UNK	HOS	OS*	UNK	HOS	OS*	UNK	HOS	OS*	UNK	HOS
Baseline	40.0	71.0	51.2	37.3	85.3	51.9	29.1	82.7	43.0	40.0	71.0	51.2	37.3	85.3	51.9
STN	**48.2**	**80.6**	**60.3**	**67.3**	**86.6**	**75.7**	54.6	78.4	64.3	46.3	76.6	57.8	63.6	84.4	72.6
SSAN	25.4	66.7	36.8	29.1	68.4	40.8	**64.5**	**83.1**	**72.7**	34.6	70.6	46.4	22.7	64.1	33.6
Partial KPG-RL	41.8	77.1	54.2	48.2	74.9	58.6	38.2	73.2	50.2	**52.7**	**82.3**	**64.3**	**80.0**	**92.6**	**85.9**

Method	D→W			W→A			W→D			W→W			**Avg**		
	OS*	UNK	HOS	OS*	UNK	HOS	OS*	UNK	HOS	OS*	UNK	HOS	OS*	UNK	HOS
Baseline	29.1	82.7	43.0	40.0	71.0	51.2	37.3	85.3	51.9	29.1	82.7	43.0	35.5	79.7	48.7
STN	54.6	79.2	64.6	49.9	78.4	60.4	60.0	82.6	69.5	55.5	79.2	65.2	55.6	80.7	65.6
SSAN	29.1	66.4	40.4	31.8	68.4	43.3	19.1	64.5	29.4	26.4	67.9	37.9	31.4	68.9	42.4
Partial KPG-RL	**73.6**	**89.2**	**80.7**	**52.7**	**82.7**	**64.4**	**78.2**	**90.9**	**84.1**	**71.8**	**88.3**	**79.2**	**59.7**	**83.5**	**69.1**

Table 5: Accuracy of common class sample (OS*), Accuracy of unknown class sample (UNK), and their harmonic mean (HOS) on Office-31 for open-set HDA ($\eta = 0.67$).

we train the kernel SVM on the transported source domain data and labeled target domain data to classify the unlabeled target domain common class data.

The natural baseline for open-set HDA is the approach that rejects the η-proportion unlabeled samples with the largest distance to labeled target domain data as unknown, and trains a kernel SVM on the labeled target domain data to classify common class data. We also evaluate the HDA methods STN (Yao et al., 2019) and SSAN (Li et al., 2020) in the open-set HDA task. For STN (Yao et al., 2019) and SSAN (Li et al., 2020), we reject the η-proportion unlabeled samples with the lowest prediction confidence as unknown class. Table 5 reports the results. We use the open-set evaluation metrics (Bucci et al., 2020), including accuracy of common class sample (OS*), accuracy of unknown class sample (UNK), and their harmonic mean HOS $= 2\frac{\text{OS}^* \times \text{UNK}}{\text{OS}^* + \text{UNK}}$. We can observe in Table 5 that our proposed partial KPG-RL achieves the best results in terms of average OS*, UNK, and HOS, indicating that the partial KPG-RL is effective for both classifying common class data and identifying unknown class data. Compared with the Baseline, partial KPG-RL outperforms it by 20.4% in terms of average HOS, confirming the positive transfer achieved by our method.

Open-set HDA with unknown $\boldsymbol{\eta}$. For the more practical open-set HDA setting that η is unknown, researchers can design methods to estimate η and then apply our method using the estimate of η, or take η as a hyper-parameter and design methods to tune it. We directly use the positive-unlabeled learning (Bekker and Davis, 2020) method (Zeiberg et al., 2020) to estimate the fraction of common class data among the target domain unlabeled data, by taking the labeled target data as positive samples. The results of different methods for open-set HDA using the estimate $\hat{\eta}$ of η are given in Table 6. According to Table 6, the positive transfer is achieved by our method. We can see that $\hat{\eta}$ in all tasks is lower than the true η, implying that fewer unknown class samples are detected. Correspondingly, the UNK value (73.2%) achieved by partial KPG-RL using $\hat{\eta}$ in Table 6 is smaller than that (83.5%) using η in Table 5. Surprisingly, the OS* value (67.5%) of partial KPG-RL in Table 6 is higher than that (59.7%) in Table 5. As a balance, the HOS value (69.5%) achieved by

Method	A→A ($\hat{\eta}=0.57$)			A→D ($\hat{\eta}=0.48$)			A→W ($\hat{\eta}=0.62$)			D→A ($\hat{\eta}=0.57$)			D→D ($\hat{\eta}=0.48$)		
	OS*	UNK	HOS	OS*	UNK	HOS	OS*	UNK	HOS	OS*	UNK	HOS	OS*	UNK	HOS
Baseline	38.2	61.9	47.2	20.0	**69.3**	31.0	28.2	**80.1**	41.7	38.2	61.9	47.2	20.0	**69.3**	31.0
Partial KPG-RL	**49.1**	**70.1**	**57.8**	**61.8**	59.3	**60.5**	**54.5**	73.2	**62.5**	**59.1**	**73.6**	**65.5**	**83.6**	66.7	**74.2**

Method	D→W ($\hat{\eta}=0.62$)			W→A ($\hat{\eta}=0.57$)			W→D ($\hat{\eta}=0.48$)			W→W ($\hat{\eta}=0.62$)			**Avg**		
	OS*	UNK	HOS	OS*	UNK	HOS	OS*	UNK	HOS	OS*	UNK	HOS	OS*	UNK	HOS
Baseline	28.2	80.1	41.7	38.2	61.9	47.2	20.0	**69.3**	31.0	28.2	80.1	41.7	28.8	70.4	40.0
Partial KPG-RL	**78.2**	**87.4**	**82.6**	**60.9**	**74.5**	**67.0**	**81.8**	66.7	**73.5**	**78.2**	**87.0**	**82.4**	**67.5**	**73.2**	**69.5**

Table 6: Results on Office-31 for open-set HDA with unknown η. $\hat{\eta}$ is the estimate of η (the true $\eta = 0.67$).

η	0.50	0.55	0.60	0.65	0.70	0.75	0.80
Average HOS	67.2	69.2	69.9	68.7	69.2	68.5	65.5

Table 7: Average HOS of partial KPG-RL using varying magnitude of η (the unknown true value of η is 0.67).

partial KPG-RL using $\hat{\eta}$ is similar to the HOS value (69.1%) of partial KPG-RL using the true η. In Table 7, we take η as a hyper-parameter and show the average HOS achieved by partial KPG-RL using varying magnitudes of η. It is observed that the average HOS is relatively stable to η in a relatively large range of $[0.55, 0.75]$.

6.3 Multi-Omic Single-Cell Alignment

We discuss the multi-omic single-cell alignment experiment in this section. In single-cell analysis, different technologies can interrogate different molecular aspects of the cell, resulting in multi-modal cell data, *e.g.*, gene expression, protein synthesis, chromatin accessibility, *etc.* Aside from a few recent co-assay procedures that simultaneously isolate separate molecular material for each measurement, applying multiple assays on the same single cell is impossible. The goal of multi-omic single-cell alignment task is to align the different modalities of cells. We consider the setting that along with a large number of unpaired cross-modal cells, a few paired cross-modal cells with 1-1 correspondence are provided. We take the paired cross-modal cells as keypoints, and then employ our keypoint-guided OT models to find the matching/correspondence of cells of different modalities, to realize the alignment further.

Datasets. Following (Demetci et al., 2022), we use two sets of single-cell multi-omics data to evaluate the usefulness of our methods for real data sets. Both data sets are generated by co-assays, and thus, we have known cell-level correspondence information for benchmarking. The first data set is generated using the *scGEM* assay (Cheow et al., 2016), which simultaneously profiles gene expression and DNA methylation. The data set

(Sequence Read Archive accession SRP077853) is derived from human somatic cell samples undergoing conversion to induced pluripotent stem cells and shows a continuous trajectory. We preprocessed the data as described in (Cheow et al., 2016; Demetci et al., 2022) and ended up with dimensions 177×34 for the gene expression data and 177×27 for the DNA methylation data. We take the gene expression as source domain, and the DNA methylation as target domain. The second data set is generated by the *SNAREseq* assay (Chen et al., 2019), which links chromatin accessibility with gene expression. The data (Gene Expression Omnibus accession GSE126074) is derived from a mixture of human cell lines: BJ, H1, K562, and GM12878, and shows distinct cell type clusters. We preprocess the data sets following (Chen et al., 2019; Demetci et al., 2022). The resulting data matrices for the SNARE-seq data set were of size 1047×19 and 1047×10 for ATAC-seq and RNA-seq, respectively. We take the ATAC-seq as source domain, and the RNA-seq as target domain.

Compared methods. We compare the following methods, including the typical multi-omic single-call alignment methods of UnionCom (Cao et al., 2020) and MMD-MA (Singh et al., 2020), the OT-based methods of Pamona (Cao et al., 2021) and SCOTv2 (Demetci et al., 2022) that are based on partial and unbalanced Gromov-Wasserstein models, respectively, the recent method of scTopoGAN (Singh et al., 2023), and our proposed partial KPG-RL, partial KPG-RL-GW, unbalanced KPG-RL, and unbalanced KPG-RL-GW. MMD-MA and scTopoGAN align distributions in a latent space using an MMD loss and a GAN objective, respectively. For a fair comparison, we augment them with an additional MSE loss on matched cells, adapting them to the semi-supervised setting. In contrast, incorporating matched-cell supervision into the other baseline methods is non-trivial.

Implementation details. To match the real-world scenario that different modalities are often unbalanced, we consider two kinds of imbalance in experiments, namely "MissType" and "Subsample". For MissType, we discard one type of cells in each domain. For Subsample, we discard 50 percent of cells from both the first half of types in source domain and the last half of types in target domain. In both kinds, we select one matched cell pair for each shared type across domains as annotated data. We use matched cells as cross-domain keypoints. Each source sample is assigned mass $1/m$, where m is the number of source samples. For each matched pair, the target keypoint is assigned the same mass as its corresponding source keypoint, while the remaining (unpaired) target samples are assigned mass $1/n$, where n is the number of target samples. Note that the total mass may differ between the source and target domains. We then apply our keypoint-guided OT models to search for the transport plan. With the transport plan, we employ the embedding technique in (Cao et al., 2020) to realize the alignment of cells in an embedding space. For partial KPG-RL and partial KPG-RL-GW, s is set to 0.9, same as Pamona (Cao et al., 2021).

Evaluation metric. For ideal alignment, the source data in the embedding space should be aligned with target data from the same cell type. To testify the correctness of alignment, we train a k-NN classifier on the target-aligned data (*i.e.*, target data in embedding space after alignment) and apply it to the source-aligned data (*i.e.*, source data in embedding space after alignment). The classification accuracy is taken as the evaluation metric.

Results. Table 8 reports the alignment accuracy of different methods. It can be observed that with the guidance of a few paired data, our proposed partial KPG-RL, partial KPG-

Method	Type	scGEM		SNARE		
		MissType	Subsample	MissType	Subsample	Avg
UnionCom	U	27.9	37.8	56.7	23.1	36.4
Pamona	U	11.7	48.1	2.8	2.4	16.3
SCOTv2	U	11.7	12.4	31.7	12.5	17.1
MMD-MA	S	31.6	25.1	36.9	38.9	33.1
scTopoGAN	S	29.1	25.2	24.4	38.2	29.2
Partial KPG-RL	S	58.6	47.4	61.2	94.5	65.4
Partial KPG-RL-GW	S	46.8	44.5	58.4	94.0	60.9
Unbalanced KPG-RL	S	56.7	**49.6**	**73.2**	**94.6**	**68.5**
Unbalanced KPG-RL-GW	S	**61.3**	46.7	68.9	94.4	67.8

Table 8: Accuracy (%) of multi-omic single-cell alignment methods on scGEM and SNARE datasets. Methods are categorized as unsupervised (U) or semi-supervised (S), where S uses paired cross-modal cells while U does not.

RL-GW, unbalanced KPG-RL, and unbalanced KPG-RL-GW improve the accuracy of the baseline methods by more than 24%. Partial KPG-RL-GW outperforms Pamona, which uses the Gromov-Wasserstein model by 44.6%, confirming the effectiveness of keypoint guidance imposed by our approach in partial Gromov-Wasserstein model. Similarly, the performance improvement (50.1%) achieved by unbalanced KPG-RL-GW over SCOTv2 based on unbalanced Gromov-Wasserstein verifies the usefulness of keypoints guidance realized by our approach in unbalanced Gromov-Wasserstein model. Compared with the semi-supervised variants of MMD-MA and scTopoGAN, our proposed approaches achieve substantially higher performance, demonstrating the effectiveness of our keypoint-based strategy. We also find that the unbalanced KPG-RL/KPG-RL-GW performs slightly better than partial KPG-RL/KPG-RL-GW, respectively.

6.4 Image-to-Image Translation

The I2I translation experiments are for evaluating the proposed KPG-RL-MBP and KPG-RL-MSP discussed in Sect. 5. We consider the "semi-paired" I2I translation task that a large number of unpaired along with a few paired cross-domain images are given for training. We aim to leverage the paired cross-domain images to guide the desired translation. We take the paired images as keypoints and use our proposed KPG-RL-MBP and KPG-RL-MSP to translate the source images to the target domain. To do this, we first learn the optimal transport plan based on Theorem 8, and then learn the transport map through Eqs. (26) and (28). In experiments, we set the difference measure $\bar{d}$ in Eq. (26) as the squared L_2-distance in feature space. The experimental details are given in Appendix F.

Datasets. The experiments are conducted on digits and natural animal images. For *Digits*, we take the MNIST (LeCun et al., 1998) and Chinese-MNIST[6] datasets as source and target

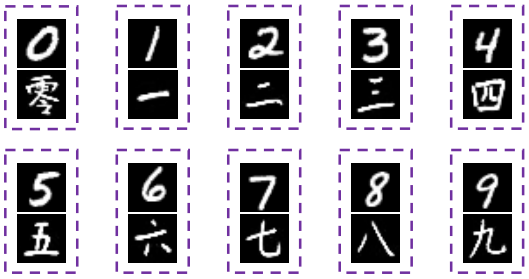
6. https://www.kaggle.com/datasets/gpreda/chinese-mnist

Figure 10: Examples of images from MNIST and Chinese-MNIST. In each dashed box, the first line is the digit corresponding to the images on the second and last lines. The second and last lines respectively show images from MNIST and Chinese-MNIST.

distributions, respectively. The MNIST and Chinese-MNIST contain the digits (from 0 to 9) in different modalities, respectively. Examples of images from MNIST and Chinese-MNIST are illustrated in Fig. 10. In experiments, we annotate 10 keypoint pairs, each corresponding to a digit, as shown in Fig. 11(a). We expect that with the guidance of the 10 keypoint pairs, the source images can be transported to the target images that represent the same digit. For *Natural Animal Images*, we take three species (cat, fox, leopard) of animals from AFHQ (Choi et al., 2020) as source distribution, and another three species (lion, tiger, wolf) as target distribution. We randomly choose 1000 images for each species. We resize the images to 256×256. Three keypoint pairs are given, as shown in Fig. 12. By the guidance of the keypoint pairs, we expect that the cat, fox, and leopard images are transported to the images of lion, tiger, and wolf, respectively.

Metrics. We adopt two evaluation metrics, *i.e.*, Frechet Inception Distance (FID) (Heusel et al., 2017) and Accuracy (ACC). "FID" is a commonly adopted metric in deep generative models for measuring how well the transported images resemble the target images. Lower FID indicates better image quality. The metric of "ACC" measures how well the source images are transported to be target images of ground-truth transported classes. For digit images, the ground-truth transported classes are defined as the digits of original source images. For animal images, the ground-truth transported classes for source images of cat, fox, and leopard are defined as lion, tiger, and wolf, respectively. Specifically, we train a classifier on the target data to recognize their class labels (*i.e.*, digits or animal species). We then predict the class labels of the transported source images using the trained classifier, and calculate the accuracy of the predictions against corresponding ground-truth transported class labels. Higher accuracy indicates that the source images are better transported to ground-truth transported classes, and thus the guidance of keypoints is better realized.

Compared methods. We compare the following I2I translation methods (see Table 9), including the typical GAN methods of Cycle-GAN (Zhu et al., 2017a), TCR (Mustafa



(a) Keypoints pairs

(b) KPG-RL-BP

(c) KPG-RL-MBP

(d) KPG-RL-MSP

Figure 11: (a) Annotated keypoint pairs for digits. Each keypoint pair is in a box. (b-d) Original and transported source images by (b) KPG-RL-BP, (c) KPG-RL-MBP, and (d) KPG-RL-MSP on digits. In (b-d), the odd columns are the source images, and their right side are the corresponding transported images.

and Mantiuk, 2020)[7], and W2GAN (Korotin et al., 2021); the recent GAN methods of KNOT (Korotin et al., 2023); the diffusion methods of EGSDE (Zhao et al., 2022), DSBM (Shi et al., 2023) and DDIB (Su et al., 2023); the flow methods of ReFlow (Liu et al., 2023), OT-CFM (Tong et al., 2024a), and SF2M (Tong et al., 2024b). For the GAN methods, we additionally use a MSE loss on the paired data in training. Among these methods, W2GAN, KNOT, ReFlow, OT-CFM, and SF2M are based on OT. In particular, OT-CFM and SF2M use the mini-batch OT to guide the training of flow matching. The compared methods also include our proposed KPG-RL-BP, KPG-RL-MBP, and KPG-RL-MSP. We additionally utilize the mini-batch version of our KPG-RL to replace the mini-batch OT in OT-CFM and SF2M to guide the training of flow matching, and the corresponding methods are denoted as KPG-RL-CFM and KPG-RL-SF2M, respectively.

Results on digits for I2I translation In Figs. 11(b), 11(c), and 11(d), we visualize the transported images by KPG-RL-BP, KPG-RL-MBP, and KPG-RL-MSP. Figure 11(b) shows that most transported source images by KPG-RL-BP belong to corresponding ground-truth transported classes, but are blurry. In Figs. 11(c) and 11(d), it can be observed that most

7. Note that TCR (Mustafa and Mantiuk, 2020) is assumes that besides the paired images, the unpaired images are only from the source domain. Since unpaired target data are given in our setting, we additionally employ the adversarial training loss on unpaired data for TCR for a fair comparison.

Method	Type	Digits		Natural animal images	
		FID ↓	ACC (%) ↑	FID ↓	ACC (%) ↑
Cycle-GAN	GAN	6.99	22.72	78.56	30.27
TCR	GAN	6.90	36.21	74.13	32.60
W2GAN	OT, GAN	12.04	34.21	121.86	28.40
KNOT	OT, GAN	3.18	9.25	118.26	27.33
EGSDE	Diffusion	34.72	11.78	52.11	29.33
DDIB	Diffusion	9.47	9.15	28.45	32.44
DSBM	Diffusion	6.52	12.34	24.53	27.33
ReFlow	OT, Flow	32.68	11.57	56.04	29.33
OT-CFM	OT, Flow	13.27	10.09	23.78	31.33
SF2M	OT, Flow	12.83	9.47	22.46	28.67
KPG-RL-CFM	OT, Flow	13.86	68.62	23.58	80.27
KPG-RL-SF2M	OT, Flow	12.96	65.13	23.15	78.33
KPG-RL-BP	OT	98.56	74.51	305.43	58.00
KPG-RL-MBP	OT, GAN	5.36	**76.28**	17.34	**87.60**
KPG-RL-MSP	OT, GAN	**3.05**	72.36	**16.62**	87.27

Table 9: FID and accuracy of the transported source images. Smaller FID indicates higher quality of transported source images. Higher accuracy implies better imposing the guidance of the keypoints in transport.

transported source images in even columns by KPG-RL-MBP and KPG-RL-MSP share the same digits (class labels) as the original source images in odd columns. Meanwhile, KPG-RL-MBP and KPG-RL-MSP produce transported images with less blur than KPG-RL-BP, which is attributed to our manifold constraints $\mathcal{L}_M$ in Eq. (26) and manifold sampling (28), respectively. We give the quantitative results in Table 9. In the left half of Table 9, we can observe that our approaches, *i.e.*, KPG-RL-CFM, KPG-RL-SF2M, KPG-RL-BP, KPG-RL-MBP, and KPG-RL-MSP, outperform the compared methods by more than 25% in terms of accuracy, demonstrating that our keypoint-guided models indeed realize the guidance of paired data to produce better translation. Meanwhile, our approaches except for KPG-RL-BP achieve comparable FID compared with the baselines. The KPG-RL-CFM and KPG-RL-SF2M respectively achieve comparable FID compared with OT-CFM and SF2M, but improve the accuracy of OT-CFM and SF2M by more than 50%, indicating that our keypoint-guidance techniques can improve the performance of OT-CFM and SF2M.

Results on natural animal images for I2I translation. The results on natural animal images for I2I translation are shown in the right half of Table 9 and Fig. 12. Table 9 shows that our proposed approach KPG-RL-MBP and KPG-RL-MSP achieves a comparable FID, compared with the other methods. The accuracy achieved by our KPG-RL-MBP/KPG-RL-MSP is higher than that of the previous compared methods by more than 50% as in Table 9. The results indicate that our proposed KPG-RL-MBP and KPG-RL-MSP better

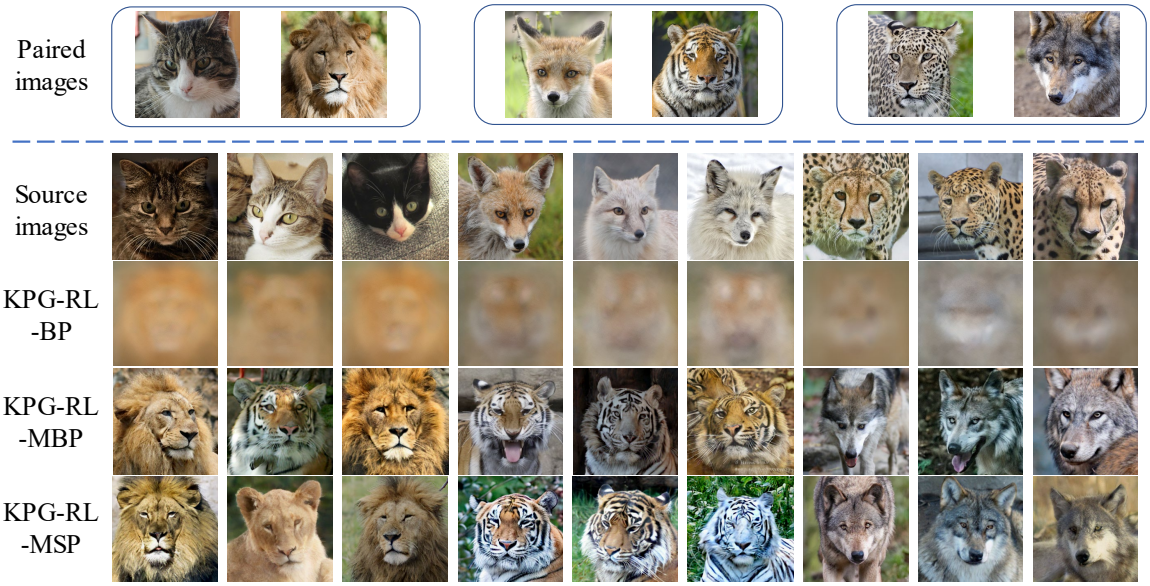


Figure 12: Top: annotated paired images (keypoints). Bottom: transported images of corresponding source images by KPG-RL-BP, KPG-RL-MBP, and KPG-RL-MSP. By the guidance of paired keypoints, the images of cat/fox/leopard are expected to be transported to images of lion/tiger/wolf.

transport the source images to corresponding ground-truth transported classes[8]. We explicitly model the guidance of paired keypoints to the unpaired data in our approaches, while the other approaches do not model the relation of samples to keypoints. This may account for the higher accuracy of KPG-RL-MBP and KPG-RL-MSP than that of the previous approaches. The KPG-RL-BP produces blurry images as in Fig. 12. KPG-RL-MBP and KPG-RL-MSP produce clearer images than KPG-RL-BP, as shown in Fig. 12, verifying the effectiveness of the manifold barycentric projection and manifold sampling. Note that ideally, most transported images by KPG-RL-BP though should be of the ground-truth transported classes according to Eq. (25), are blurry as in Fig. 12. Hence the classifier may not recognize the transported images by KPG-RL-BP correctly. This leads to lower accuracy than KPG-RL-MBP and KPG-RL-MSP in Table 9. We can also observe that KPG-RL-CFM and KPG-RL-SF2M achieve comparable FID compared with OT-CFM and SF2M, and improve their accuracy by more than 40%. This further verifies the importance of the keypoint guidance in OT-CFM and SF2M.

Effect of the number of keypoints. We study the effect of the number of paired images in semi-paired I2I translation experiments. Table 10 shows that for both KPG-RL-MBP and

8. The ground-truth transported classes for source images of cat, fox, and leopard are lion, tiger, and wolf, respectively.

Number		3	6	9	12	Full
KPG-RL-MBP	FID ↓	17.34	16.69	15.53	15.69	14.72
	ACC (%) ↑	87.60	89.40	91.33	92.67	98.13
KPG-RL-MSP	FID ↓	16.62	16.48	16.23	16.15	15.81
	ACC (%) ↑	87.27	88.44	89.40	90.67	96.67

Table 10: Results of KPG-RL-MBP and KPG-RL-MSP on natural animal images with varying numbers of keypoint pairs.

KPG-RL-MSP, as the number of paired data increases, the accuracy increases, and the FID marginally decreases. This implies that our approach can impose the guidance of different amounts of paired data to translate source images to desired classes.

7. Conclusion

This paper proposes a novel KPG-RL model that leverages keypoints to guide the correct matching (*i.e.*, transport plan) in OT. We devise a mask-based constraint to preserve the matching of keypoints pairs, and propose to preserve the relation of each point to the keypoints to impose the guidance of these keypoints in OT. The keypoint-guided OT is built in KP/GW formulations for both balanced and partial/unbalanced transport settings. We further propose neural transport strategies of manifold barycentric projection and manifold sampling to transport source data to target domain, based on the developed duality of KPG-RL. The effectiveness of the proposed KPG-RL model is verified in the application of HDA, multi-omic single-cell alignment, and I2I translation. In the future, we are interested in more applications, *e.g.*, Multimodal Large Language Models, of keypoint-guided OT.

Acknowledgments

We thank the editors and reviewers for the valuable suggestions, enabling us to improve the quality of this work. This work was supported by National Key R&D Program 2021YFA1003000, NSFC (12501709, 12125104, 12426313, 623B2084), the Major Key Project of PCL under Grant PCL2024A06, China National Postdoctotal Program for Innovative Talents (BX20240276), China Fundamental Research Funds for the Central Universities (xzy022025047), China Postdoctoral Science Foundation (2025M773058), and Shaanxi Province "Sanqin Bochuang" Talent Support Program (2024SQBC021).

Appendix A. Proof and Generalization of Proposition 1

Proposition 1 in the paper is for the case that $p_i = q_j$, for all $(i, j) \in \mathcal{K}$. As stated in the paper, the mask-based modeling of the transport plan is applicable even for the case that there exist some $(i, j) \in \mathcal{K}$ such that $p_i \neq q_j$. To see this, we first mathematically give the

definition of preserving the matching of keypoint pairs and then prove Proposition 10, a generalization of Proposition 1.

Definition 9 *Given the marginal distributions $\boldsymbol{p}$ and $\boldsymbol{q}$, we say that the transport plan $\pi \in \Pi(\boldsymbol{p}, \boldsymbol{q})$ preserves the matching of a keypoint pair with index $(i, j) \in \mathcal{K}$, if π satisfies one of the following conditions:*

1. *If $p_i = q_j$, π satisfies that $\pi_{i,j'} = 0, \forall j' \neq j; \pi_{i',j} = 0, \forall i' \neq i; \pi_{i,j} = p_i = q_j$.*
2. *If $p_i > q_j$, π satisfies that $\pi_{i',j} = 0, \forall i' \neq i; \pi_{i,j} = q_j$.*
3. *If $p_i < q_j$, π satisfies that $\pi_{i,j'} = 0, \forall j' \neq j; \pi_{i,j} = p_i$.*

The left part of Fig. 13 illustrates these conditions. Specifically, the first condition implies that if $p_i = q_j$ (*e.g.*, (i, j) is taken as (4, 4) in Fig. 13), the all mass p_i of x_i will be transported to y_j and y_j can only receive mass from x_i. The second condition implies that if $p_i > q_j$ (*e.g.*, (i, j) is taken as (3, 2) in Fig. 13), y_j can only receive mass from x_i and consequently the partial mass $p_i - q_j$ of x_i is allowed to be transported to the target points apart from y_j. The third condition indicates that if $p_i < q_j$ (*e.g.*, (i, j) is taken as (6, 5) in Fig. 13), the all mass p_i of x_i will be transported to y_j and y_j is enabled to receive partial mass $q_j - p_i$ from the source points apart from x_i. For the convenience of description, for each pair $(i, j) \in \mathcal{K}$, we denote $j = \kappa(i)$ and $i = \kappa'(j)$.

Proposition 10 *Suppose that the mask matrix M satisfies that*

$$M_{i,j} = \begin{cases} 1, & \text{if } (i,j) \in \mathcal{K}, \\ 0, & \text{if } i \in \mathcal{I},\ p_i \leq q_{\kappa(i)},\ \text{and } (i,j) \notin \mathcal{K}, \\ 0, & \text{if } j \in \mathcal{J},\ p_{\kappa'(j)} \geq q_j,\ \text{and } (i,j) \notin \mathcal{K}, \\ 1, & \text{if } i \in \mathcal{I},\ p_i > q_{\kappa(i)},\ \text{and } (i,j) \notin \mathcal{K}, \\ 1, & \text{if } j \in \mathcal{J},\ p_{\kappa'(j)} < q_j,\ \text{and } (i,j) \notin \mathcal{K}, \\ 1, & \text{otherwise (i.e., } i \notin \mathcal{I}, j \notin \mathcal{J}). \end{cases} \tag{29}$$

Then, the transport plan $\tilde{\pi} = M \odot \pi$ with $\pi \in \Pi(\boldsymbol{p}, \boldsymbol{q}; M)$ preserves the matching of keypoint pairs with index in $\mathcal{K}$.

According to the definition of M, $M_{i,j} = 1$ for the keypoint pair $(i, j) \in \mathcal{K}$, implying that $\tilde{\pi}_{i,j}$ could take non-zero value. For $i \in \mathcal{I}$, $(i, j) \notin \mathcal{K}$ and $p_i \leq q_{\kappa(i)}$, $M_{i,j}$ is set to 0, enforcing that the i-th row of $\tilde{\pi}$ are zeros except for the location $\kappa(i)$ of the target keypoint paired with i (*e.g.*, the 4-th and 6-th rows of $\tilde{\pi}$ in Fig. 13). Similarly, for $j \in \mathcal{J}$, $(i, j) \notin \mathcal{K}$, and $p_{\kappa'(j)} \leq q_j$, we set $M_{i,j} = 0$, enforcing that the j-th column of $\tilde{\pi}$ are zeros except for the location $\kappa'(j)$ of the source keypoint paired with j (*e.g.*, the 2-th and 4-th columns of $\tilde{\pi}$ in Fig. 13). For the other points (corresponding to the last three cases in Eq. (29)), we set $M_{i,j} = 1$, indicating that there is no additional constraint on $\tilde{\pi}_{i,j}$. If $p_i = q_j$, for all $(i, j) \in \mathcal{K}$ (*i.e.*, $p_i = q_{\kappa(i)}, \forall i \in \mathcal{I}$), Proposition 10 degenerates to Proposition 1 in the paper.

Proof of Proposition 10: For any $(i, j) \in \mathcal{K}$, we next prove that $\tilde{\pi}$ preserves the matching of keypoint pair (i, j).

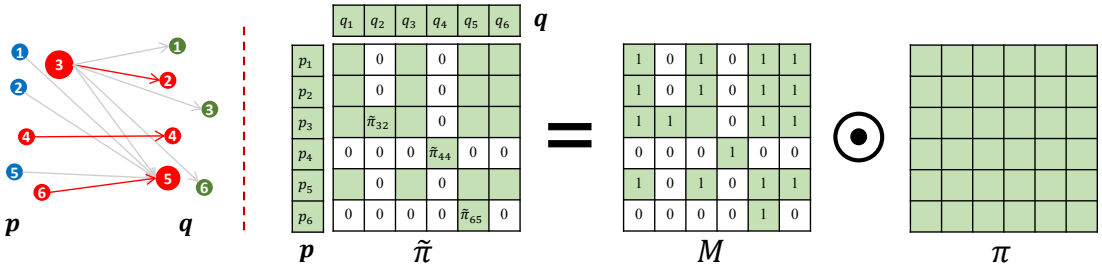


Figure 13: Example of modeling the matching of keypoints (red) using mask, where $\mathcal{K} = \{(3,2),(4,4),(6,5)\}$ with $p_3 > q_2, p_4 = q_4, p_6 < q_5$.

- If $p_i = q_j$, from the definition of M, we have $M_{i,j'} = 0$ for all $j' \neq j$ and $M_{i,j} = 1$. Then, we have $\tilde{\pi}_{i,j'} = M_{i,j'}\pi_{i,j'} = 0$ for all $j' \neq j$. Since $\sum_{j'=1}^{n} \tilde{\pi}_{i,j'} = p_i$, we have $\tilde{\pi}_{i,j} = p_i$. Similarly, we have $M_{i',j} = 0$ for all $i' \neq i$. Then, we have $\tilde{\pi}_{i',j} = M_{i',j}\pi_{i',j} = 0$ for all $i' \neq i$, and $\tilde{\pi}_{i,j} = \sum_{i'=1}^{m} \tilde{\pi}_{i',j} = q_j$.
- If $p_i > q_j$, from the definition of M, we have $M_{i',j} = 0$ for all $i' \neq i$ and $M_{i,j} = 1$. Then, we have $\tilde{\pi}_{i',j} = M_{i',j}\pi_{i',j} = 0$ for all $i' \neq i$, and $\tilde{\pi}_{i,j} = \sum_{i'=1}^{m} \tilde{\pi}_{i',j} = q_j$.
- $p_i < q_j$, from the definition of M, we have $M_{i,j'} = 0$ for all $j' \neq j$ and $M_{i,j} = 1$. Then $\tilde{\pi}_{i,j'} = M_{i,j'}\pi_{i,j'} = 0$ for all $j' \neq j$, and $\tilde{\pi}_{i,j} = \sum_{j'=1}^{n} \tilde{\pi}_{i,j'} = p_i$.

Thus, for any keypoint pair with index $(i,j) \in \mathcal{K}$, $\tilde{\pi}$ satisfies the conditions in Definition 9. This means that $\tilde{\pi}$ preserves the matching of keypoint pairs with index in $\mathcal{K}$.

■

Appendix B. Algorithms

B.1 Linear Programming for Solving KPG-RL

We cast the matrix G (resp. M, π) as the vector $\boldsymbol{c}$ (resp. $\boldsymbol{m}, \boldsymbol{x}$) $\in \mathbb{R}^{mn}$, such that the $(i + m(j-1))$-th element of $\boldsymbol{c}$ is G_{ij}. By denoting

$$A = \begin{bmatrix} \mathbb{1}_n^\top \otimes I_m \\ I_n \otimes \mathbb{1}_m \end{bmatrix}, \quad \boldsymbol{h} = \begin{bmatrix} \boldsymbol{p} \\ \boldsymbol{q} \end{bmatrix}, \tag{30}$$

where I_m is the identity matrix of size n and $\otimes$ is the Kronecker product, the KPG-RL model in Eq. (11) in the paper reads

$$\begin{aligned} &\min_{\boldsymbol{x}} \boldsymbol{c}^\top \boldsymbol{x} \\ &s.t.\ \ \boldsymbol{x} \geq 0, \\ &\quad A\boldsymbol{x} = \boldsymbol{h}, \\ &\quad \mathrm{diag}(\mathbb{1}_{mn} - \boldsymbol{m})\boldsymbol{x} = 0. \end{aligned} \tag{31}$$

With the standard form of linear programming in Eq. (31), the Simplex algorithm can be directly used to solve the KPG-RL model.

B.2 Sinkhorn's Algorithm for Solving KPG-RL

The entropy-regularized model for KPG-RL is

$$\begin{aligned} &\min_{\pi} \langle M \odot \pi, G \rangle_F - \epsilon E(M \odot \pi) \\ &s.t.\ \ \pi \geq 0, (M \odot \pi)\mathbb{1}_n = \boldsymbol{p}, (M \odot \pi)^\top \mathbb{1}_m = \boldsymbol{q}, (\mathbb{1}_{m \times n} - M) \odot \pi = 0, \end{aligned} \tag{32}$$

where $E(M \odot \pi) = -(\langle M \odot \pi, \log(M \odot \pi) \rangle_F - \mathbb{1}_m^\top (M \odot \pi)\mathbb{1}_n)$ is the entropy of the transport plan $M \odot \pi$. The Lagrangian function is

$$\begin{aligned} L(\pi, \boldsymbol{f}, \boldsymbol{g}) =& \langle M \odot \pi, G \rangle_F + \epsilon \left(\langle M \odot \pi, \log(M \odot \pi) \rangle_F - \mathbb{1}_m^\top (M \odot \pi)\mathbb{1}_n \right) \\ &- \langle \boldsymbol{f}, (M \odot \pi)\mathbb{1}_n - \boldsymbol{p} \rangle_F - \langle \boldsymbol{g}, (M \odot \pi)^\top \mathbb{1}_m - \boldsymbol{q} \rangle_F, \end{aligned} \tag{33}$$

where $\boldsymbol{f} \in \mathbb{R}^m$ and $\boldsymbol{g} \in \mathbb{R}^n$. The first-order conditions then yield

$$\frac{\partial L}{\partial \pi_{i,j}} = M_{i,j} G_{i,j} + \epsilon M_{i,j} \log(M_{i,j} \pi_{i,j}) - M_{i,j} f_i - M_{i,j} g_j = 0. \tag{34}$$

If $M_{i,j} = 0$, we have $\pi_{i,j} = 0$ according to the constraints, and if $M_{i,j} = 1$, we have $\pi_{i,j} = e^{f_i/\epsilon} e^{-G_{i,j}/\epsilon} e^{g_j/\epsilon}$. Therefore, we can unify the expression as $\pi_{ij} = M_{ij} e^{f_i/\epsilon} e^{-C_{ij}/\epsilon} e^{g_j/\epsilon}$. The matrix form is $\pi = \mathrm{diag}(\boldsymbol{u}) K \mathrm{diag}(\boldsymbol{v})$ where $\boldsymbol{u} = e^{\boldsymbol{f}/\epsilon}, K = M \odot e^{-G/\epsilon}$, and $\boldsymbol{v} = e^{\boldsymbol{g}/\epsilon}$. The constraints are

$$\mathrm{diag}(\boldsymbol{u}) K \mathrm{diag}(\boldsymbol{v}) \mathbb{1}_n = \boldsymbol{p}, \quad (\mathrm{diag}(\boldsymbol{u}) K \mathrm{diag}(\boldsymbol{v}))^\top \mathbb{1}_m = \boldsymbol{q}. \tag{35}$$

Since the entries of K are non-negative, the Sinkhorn's algorithm can be applied (Sinkhorn and Knopp, 1967). The iteration formulas are

$$\boldsymbol{u}^{(l+1)} = \frac{\boldsymbol{p}}{K \boldsymbol{v}^l}, \ \ \boldsymbol{v}^{(l+1)} = \frac{\boldsymbol{q}}{K^\top \boldsymbol{u}^{(l+1)}}. \tag{36}$$

The division operator used above is entry-wise.

Log-domain Sinkhorn iteration. For KP, the Sinkhorn iteration in the log-domain is more stable Schmitz et al. (2018). We next deduce the log-domain Sinkhorn iteration for our KPG-RL. In the log-domain, the left equation in Eq. (36) is

$$\frac{1}{\epsilon} f_i^{(l+1)} = \log(p_i) - \log \left(\sum_{j=1}^{n} M_{i,j} e^{\frac{-G_{i,j} + g_j^{(l)}}{\epsilon}} \right). \tag{37}$$

Let

$$H_G(\boldsymbol{f}, \boldsymbol{g}) = \frac{1}{\epsilon}(-G + \boldsymbol{f}\mathbb{1}_n^\top + \boldsymbol{g}\mathbb{1}_m^\top), \tag{38}$$

then

$$\begin{aligned} f_i^{(l+1)} =& \epsilon \log(p_i) - \epsilon \log \left(\sum_{j=1}^{n} M_{i,j} e^{H_G(\boldsymbol{f}^{(l)}, \boldsymbol{g}^{(l)})_{i,j}} e^{-f_i^{(l)}/\epsilon} \right) \\ =& \epsilon \log(p_i) - \epsilon \log \left(e^{-f_i^{(l)}/\epsilon} \sum_{j=1}^{n} M_{i,j} e^{H_G(\boldsymbol{f}^{(l)}, \boldsymbol{g}^{(l)})_{i,j}} \right) \\ =& \epsilon \log(p_i) - \epsilon \log \left(\sum_{j=1}^{n} M_{i,j} e^{H_G(\boldsymbol{f}^{(l)}, \boldsymbol{g}^{(l)})_{i,j}} \right) + f_i^{(l)} \\ =& \epsilon \log(p_i) - \epsilon \log \left(\sum_{j=1}^{n} e^{\log(M_{i,j}) H_G(\boldsymbol{f}^{(l)}, \boldsymbol{g}^{(l)})_{i,j}} \right) + f_i^{(l)}. \end{aligned} \tag{39}$$

If $M_{i,j} = 0$, $\log(M_{i,j}) H_G(\boldsymbol{f}^{(l)}, \boldsymbol{g}^{(l)})_{i,j} = -\infty$. We define $\bar{H}(\boldsymbol{f}, \boldsymbol{g})$ as

$$\bar{H}_G(\boldsymbol{f}, \boldsymbol{g})_{i,j} = \begin{cases} H_G(\boldsymbol{f}, \boldsymbol{g})_{i,j} & \text{if } M_{i,j} = 1, \\ -\infty & \text{if } M_{i,j} = 0, \end{cases} \tag{40}$$

and define the log-sum-exp function $\mathrm{LogSumExp} : \mathbb{R}^{m \times n} \to \mathbb{R}^m$ as

$$\mathrm{LogSumExp}(A) = \left(\log(\sum_j e^{A_{1,j}}), \log(\sum_j e^{A_{2,j}}), \cdots, \log(\sum_j e^{A_{m,j}}) \right)^\top. \tag{41}$$

Then, the matrix form of Eq. (39) becomes

$$\boldsymbol{f}^{(l+1)} = \epsilon \log(\boldsymbol{p}) - \epsilon \mathrm{LogSumExp} \left(\bar{H}_G(\boldsymbol{f}^{(l)}, \boldsymbol{g}^{(l)}) \right) + \boldsymbol{f}^{(l)}. \tag{42}$$

Similarly, the corresponding iteration formula in the log-domain of the right equation in (36) is

$$\boldsymbol{g}^{(l+1)} = \epsilon \log(\boldsymbol{q}) - \epsilon \mathrm{LogSumExp} \left(\bar{H}_G(\boldsymbol{f}^{(l+1)}, \boldsymbol{g}^{(l)})^\top \right) + \boldsymbol{g}^{(l)}. \tag{43}$$

Equations (42) and (43) form the formulas of the Sinkhorn iteration in the log-domain.

B.3 Convergence of Sinkhorn Iteration

We note that $\Pi(\boldsymbol{p}, \boldsymbol{q}; M)$ is not empty for masks in both Proposition 1 and Proposition 10. Thus, Proposition 2 shows the convergence of the Sinkhorn iteration for masks defined in Proposition 1 and Proposition 10. We next prove Proposition 2 and empirically validate this convergence through experiments.

Proof of Proposition 2. By assumption, $\Pi(\boldsymbol{p}, \boldsymbol{q}; M) \neq \emptyset$. Since $\Pi(\boldsymbol{p}, \boldsymbol{q}; M)$ is a convex polytope and the objective is strictly convex on the admissible entries, the entropy-regularized KPG-RL problem admits a unique global minimizer.

We next show that the Sinkhorn iteration is an alternating KL/Bregman projection scheme. Define

$$\begin{aligned}\mathcal{R} =&\{\pi \in \mathbb{R}_+^{m\times n} : \pi \mathbf{1}_n = \boldsymbol{p},\ (\mathbf{1} - M) \odot \pi = 0\},\\ \mathcal{C} =&\{\pi \in \mathbb{R}_+^{m\times n} : \pi^\top \mathbf{1}_m = \boldsymbol{q},\ (\mathbf{1} - M) \odot \pi = 0\},\end{aligned}$$

so that $\Pi(\boldsymbol{p}, \boldsymbol{q}; M) = \mathcal{R} \cap \mathcal{C}$. Starting from $\pi^{(0)} = K$, alternating KL projections onto $\mathcal{R}$ and $\mathcal{C}$ generate

$$\pi^{(l+1/2)} = P_{\mathcal{R}}^{\mathrm{KL}}(\pi^{(l)}) = \arg\min_{\pi\in\mathcal{R}} \mathrm{KL}(\pi \mid \pi^{(l)}) = \mathrm{diag}\left(\frac{\boldsymbol{p}}{\pi^{(l)}\mathbf{1}_n}\right)\pi^{(l)},$$

and

$$\pi^{(l+1)} = P_{\mathcal{C}}^{\mathrm{KL}}(\pi^{(l+1/2)}) = \arg\min_{\pi\in\mathcal{C}} \mathrm{KL}(\pi \mid \pi^{(l+1/2)}) = \pi^{(l+1/2)}\,\mathrm{diag}\left(\frac{\boldsymbol{q}}{(\pi^{(l+1/2)})^\top \mathbf{1}_m}\right),$$

where the division is element-wise. Since every iterate has the form

$$\pi^{(l)} = \mathrm{diag}(\boldsymbol{u}^{(l)}) K\, \mathrm{diag}(\boldsymbol{v}^{(l)}),$$

the above updates reduce to

$$\boldsymbol{u}^{(l+1)} = \frac{\boldsymbol{p}}{K\boldsymbol{v}^{(l)}}, \qquad \boldsymbol{v}^{(l+1)} = \frac{\boldsymbol{q}}{K^\top \boldsymbol{u}^{(l+1)}},$$

which are exactly the Sinkhorn iterations in Eq. (12).

By the convergence theorem for cyclic Bregman projections onto closed convex sets with nonempty intersection, the transport plans $\pi^{(l)}$ converge to the KL projection of K onto $\mathcal{R} \cap \mathcal{C}$, namely

$$\pi^\star = \arg\min_{\pi\in\Pi(\boldsymbol{p},\boldsymbol{q};M)} \mathrm{KL}(\pi \mid K).$$

This completes the proof.

■

Empirical convergence verification. To empirically validate the convergence of Sinkhorn's algorithm, we repeated the following procedure 100 times under different settings. In each run, we (i) sampled 100 points from a two-dimensional Gaussian distribution for both the source and target domains, (ii) randomly selected U cross-domain pairs as keypoints, and (iii) applied our keypoint-guided OT method (KPG-RL) using Sinkhorn's algorithm with $\epsilon = 0.1$. Table 11 reports the iteration counts under different stopping tolerances, for varying numbers of keypoints and two kinds of keypoint index orderings. The results show that Sinkhorn reaches the prescribed tolerance in a finite number of iterations, and that the convergence behavior is insensitive to the number and ordering of keypoints. Figure 14 presents representative error curves, demonstrating a monotonic decrease as iterations proceed.

We further examine the convergence of the Sinkhorn iteration under the general mask in Proposition 10. We use the unequal-mass setting illustrated in Fig. 13, where the source and target distributions are supported on six points, respectively, and the keypoint pairs are $\mathcal{K} = \{(3,2),(4,4),(6,5)\}$. The prescribed marginals are set as $\boldsymbol{p} =$

	Thresh $= 0.1$		Thresh $= 0.01$		Thresh $= 0.001$	
	Arange	Rand	Arange	Rand	Arange	Rand
$U = 3$	2.00 ± 0.00	2.00 ± 0.00	3.22 ± 0.42	3.23 ± 0.42	4.57 ± 0.61	4.53 ± 0.63
$U = 5$	2.02 ± 0.14	2.00 ± 0.00	3.26 ± 0.44	3.24 ± 0.43	4.61 ± 0.58	4.73 ± 0.58
$U = 10$	2.01 ± 0.10	2.00 ± 0.00	3.18 ± 0.39	3.20 ± 0.40	4.70 ± 0.59	4.76 ± 0.61

Table 11: Number of iterations of the Sinkhorn's algorithm (mean±std over 100 runs) for the mask defined in Proposition 1. "Thresh" is the stopping tolerance for Sinkhorn (based on $\|\boldsymbol{u}^{(l+1)} - \boldsymbol{u}^{(l)}\|_2$). "Arange" places the randomly selected keypoint indices contiguously at the front, while "Rand" keeps the keypoint indices in random positions.

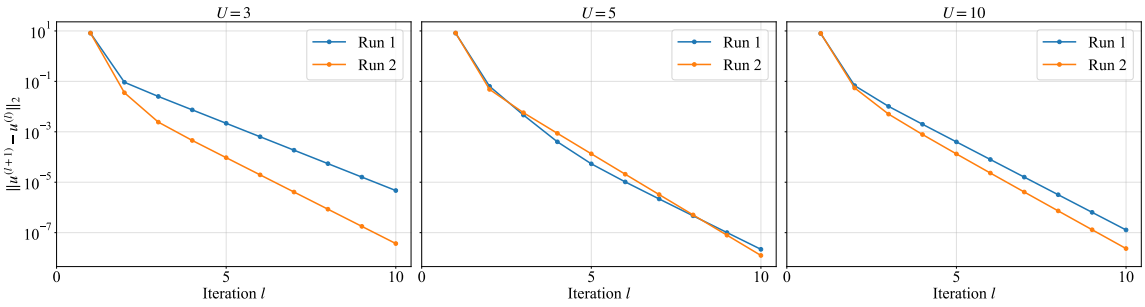


Figure 14: Example curves of $\|\boldsymbol{u}^{(l+1)} - \boldsymbol{u}^{(l)}\|_2$ in two random runs under each number of keypoint pairs, for the mask defined in Proposition 1.

$(0.16, 0.14, 0.22, 0.18, 0.15, 0.15)$ and $q = (0.15, 0.12, 0.15, 0.18, 0.18, 0.22)$, so that $p_3 > q_2$, $p_4 = q_4$, and $p_6 < q_5$. The corresponding mask is generated according to Proposition 10. For each trial, the source and target locations are independently sampled from a two-dimensional standard Gaussian distribution, and the cost matrix is given by the squared Euclidean distance. We run the log-domain Sinkhorn iteration with $\epsilon = 0.1$, and report the same diagnostics as in Table 12 and Fig. 15, including the number of iterations required to reach prescribed tolerances for $\|\boldsymbol{u}^{(l+1)} - \boldsymbol{u}^{(l)}\|_2$ and the final marginal violation. The experiment is repeated 100 times. Table 12 and Fig. 15 indicate the convergence of Sinkhorn iteration.

B.4 Frank-Wolfe Algorithm for Solving (Partial) KPG-RL-GW

Frank-Wolfe algorithm for solving KPG-RL-GW. We define the 4-order tensor $L = (L_{i,j,k,l}) \in \mathbb{R}^{m \times n \times m \times n}$ by $L_{i,j,k,l} = (C^s_{i,k} - C^t_{j,l})^2$, and define the tensor-matrix product $L \circ \pi = ((L \circ \pi)_{i,j}) \in \mathbb{R}^{m \times n}$ by $(L \circ \pi)_{i,j} = \sum_{k,l} L_{i,j,k,l} \pi_{k,l}$. Then, the KPG-RL-GW model in Eq. (14) in

Thresh	0.1	0.01	0.001
$U = 3$	2.22 ± 0.44	8.17 ± 3.24	18.25 ± 5.16

Table 12: Number of iterations of the Sinkhorn's algorithm (mean±std over 100 runs) for the general mask defined in Proposition 10. "Thresh" is the stopping tolerance for Sinkhorn (based on $\|\boldsymbol{u}^{(l+1)} - \boldsymbol{u}^{(l)}\|_2$).

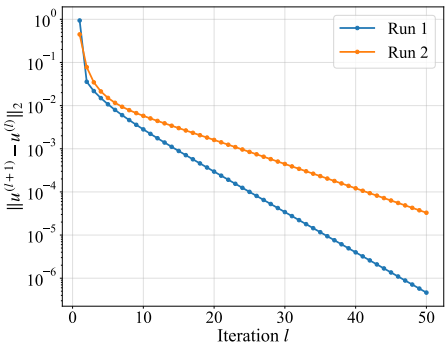


Figure 15: Example curves of $\|\boldsymbol{u}^{(l+1)} - \boldsymbol{u}^{(l)}\|_2$ in two random runs for the general mask defined in Proposition 10.

the paper reads

$$\min_{\pi \in \Pi(\boldsymbol{p},\boldsymbol{q};M)} \langle (M \odot \pi), \alpha L \circ (M \odot \pi) + (1-\alpha)G \rangle_F \triangleq \mathcal{L}(\pi) \tag{44}$$

The gradient of the objective function is

$$\nabla \mathcal{L}(\pi) = M \odot (2\alpha L \circ (M \odot \pi) + (1-\alpha)G). \tag{45}$$

In the k-th iteration of the Frank-Wolfe algorithm, it runs the following three steps:

Step 1. Compute a linear minimization oracle over the set $\Pi(\boldsymbol{p}, \boldsymbol{q}; M)$, *i.e.*,

$$\hat{\pi} \leftarrow \operatorname*{argmin}_{\pi \in \Pi(\boldsymbol{p},\boldsymbol{q};M)} \langle \nabla \mathcal{L}(\pi^{(k)}), \pi \rangle_F. \tag{46}$$

Equation (46) can be rewritten as

$$\hat{\pi} \leftarrow \operatorname*{argmin}_{\pi \in \Pi(\boldsymbol{p},\boldsymbol{q};M)} \langle M \odot \pi, 2\alpha L \circ (M \odot \pi^{(k)}) + (1-\alpha)G \rangle_F, \tag{47}$$

Equation (47) is a KPG-RL-like problem and can be solved using linear programming or Sinkhorn's algorithm.

Step 2. Determine optimal step-size $\beta^{(k)}$ subject to

$$\beta^{(k)} \leftarrow \operatorname*{argmin}_{\beta \in [0,1]} \mathcal{L}((1-\beta)\pi^{(k)} + \beta\hat{\pi}). \tag{48}$$

$\beta^{(k)}$ can be obtained by the line search method as in (Titouan et al., 2019).

Step 3. Update

$$\pi^{(k+1)} = (1-\beta^{(k)})\pi^{(k)} + \beta^{(k)}\hat{\pi}. \tag{49}$$

Frank-Wolfe algorithm for solving partial KPG-RL-GW. The Frank-Wolfe algorithm for solving partial KPG-RL-GW (discussed in Sect. 4.5.1) is the same as that for KPG-RL-GW discussed above, except that the solution set in Step 1 is $\Pi^s(\boldsymbol{p}, \boldsymbol{q}; M)$. Therefore, $\hat{\pi}$ is obtained by solving a partial KPG-RL-like model, *i.e.*,

$$\hat{\pi} \leftarrow \underset{\pi \in \Pi^s(\boldsymbol{p},\boldsymbol{q};M)}{\operatorname{argmin}} \langle M \odot \pi, 2\alpha L \circ (M \odot \pi^{(k)}) + (1-\alpha)G \rangle_F, \tag{50}$$

which is solved similarly to problem (17).

B.5 Algorithm for Solving Unbalanced KPG-RL-GW

Similar to the log-domain Sinkhorn iteration in Appendix B.2, we provide the generalized Sinkhorn iteration (Eqs. (20)) for unbalanced KPG-RL (Eq. (19)) in log-domain as follows:

$$\begin{aligned} \boldsymbol{f}^{(l+1)} =& \boldsymbol{\rho} \odot \left[\epsilon \log(\boldsymbol{p}) - \epsilon \text{LogSumExp}\left(\bar{H}_G(\boldsymbol{f}^{(l)}, \boldsymbol{g}^{(l)}) \right) + \boldsymbol{f}^{(l)} \right] \\ \boldsymbol{g}^{(l+1)} =& \boldsymbol{\rho} \odot \left[\epsilon \log(\boldsymbol{q}) - \epsilon \text{LogSumExp}\left(\bar{H}_G(\boldsymbol{f}^{(l+1)}, \boldsymbol{g}^{(l)})^\top \right) + \boldsymbol{g}^{(l)} \right]. \end{aligned} \tag{51}$$

Following unbalanced GW (Séjourné et al., 2021), we optimize the following relaxation of unbalanced KPG-RL-GW with entropic regularization:

$$\begin{aligned} F_{kgw}(\pi, \gamma) =& \alpha \sum_{i,k=1}^{m} \sum_{j,l=1}^{n} \tilde{\pi}_{i,j} \tilde{\gamma}_{k,l} |C^s_{i,k} - C^t_{j,l}|^2 + (1-\alpha) \sum_{i,j} G_{i,j} \tilde{\pi}_{i,j} \\ &+ \mu_1 \bar{\mathcal{D}}^{\otimes}(\tilde{\pi}_1 \times \tilde{\gamma}_1 | \boldsymbol{p} \times \boldsymbol{p}) + \mu_2 \bar{\mathcal{D}}^{\otimes}(\tilde{\pi}_2 \times \tilde{\gamma}_2 | \boldsymbol{q} \times \boldsymbol{q}) + \epsilon E(\tilde{\pi} \times \tilde{\gamma}), \end{aligned} \tag{52}$$

where $\tilde{\pi} = M \odot \pi$ and $\tilde{\gamma} = M \odot \gamma$. Inspired by (Séjourné et al., 2021), we alternately update π and γ as in Algorithm 1. We show empirically the effectiveness of the algorithm in Sect. 6.3.

Appendix C. Proof of Theorems 4 and 6

In this Appendix, for convenience in description, we denote the mask matrix for distributions $\boldsymbol{p}$ and $\boldsymbol{q}$ as $M^{\boldsymbol{pq}}$. Note that Theorems 4 and 6 are for the case where $p_i = q_j$ for all $(i,j) \in \mathcal{K}$.

C.1 Proof of Theorem 4

We verify the conditions of the proper metric as follows.

(1) Prove that $\mathcal{S}_{krk}(\boldsymbol{p}, \boldsymbol{q}) = 0$ if and only if $\boldsymbol{p} = \boldsymbol{q}$.

(a) If $\boldsymbol{p} = \boldsymbol{q}$, we have $x_i = y_i$ and $p_i = q_i$ for any $i \in [m]$ (since the permutation of support points does not change the distribution). Hence, $C_{i,i} = c(x_i, y_i) = 0$, and $C^s_{i,i_u} = c(x_i, x_{i_u}) = c(y_i, y_{i_u}) = C^t_{i,i_u}, \forall i \in [m]$ and $\forall i_u \in \mathcal{I}$, which implies that $R^s_i = R^t_i$. Then, we have $G_{i,i} = d(R^s_i, R^t_i) = 0$. We define π by $\pi_{i,j} = p_i$ if $i = j$, otherwise 0.

Algorithm 1 Algorithm for Unbalanced KPG-RL-GW.

Let $\zeta(\pi) = \sum_{i,j} \pi_{i,j}$ be the total mass of π
Initialization: $\tilde{\pi} = \tilde{\gamma} = \boldsymbol{p} \times \boldsymbol{q}$
while $\tilde{\gamma}$ not converged **do**
  $\tilde{\gamma} = \tilde{\pi}$
  $\tilde{\epsilon} = \zeta(\tilde{\gamma})\epsilon, \tilde{\boldsymbol{\rho}} = \zeta(\tilde{\gamma})\boldsymbol{\rho}$
  Compute $\tilde{C} = (\tilde{C}_{i,j}) \in \mathbb{R}^{m\times n}, \tilde{C}_{i,j} = \alpha \sum_{k,l} |C^s_{i,k} - C^t_{j,l}|^2 \tilde{\gamma}_{k,l} + (1-\alpha)G_{i,j} + \mu\bar{\mathcal{D}}(\tilde{\gamma}\mathbb{1}_n|\boldsymbol{p}) + \mu\bar{\mathcal{D}}(\tilde{\gamma}^\top \mathbb{1}_m|\boldsymbol{q}) + \epsilon\bar{\mathcal{D}}(\tilde{\gamma}|\boldsymbol{p}\times\boldsymbol{q})$
  for $l = 1, 2, \cdots$ **do**
    $\boldsymbol{f}^{(l+1)} = \tilde{\boldsymbol{\rho}} \odot \left[\epsilon \log(\boldsymbol{p}) - \epsilon \text{LogSumExp}\left(\bar{H}_{\tilde{C}}(\boldsymbol{f}^{(l)}, \boldsymbol{g}^{(l)})\right) + \boldsymbol{f}^{(l)}\right]$
    $\boldsymbol{g}^{(l+1)} = \tilde{\boldsymbol{\rho}} \odot \left[\epsilon \log(\boldsymbol{q}) - \epsilon \text{LogSumExp}\left(\bar{H}_{\tilde{C}}(\boldsymbol{f}^{(l+1)}, \boldsymbol{g}^{(l)})^\top\right) + \boldsymbol{g}^{(l)}\right]$
  end for
  $\boldsymbol{u} = e^{\boldsymbol{f}/\tilde{\epsilon}}, \boldsymbol{v} = e^{\boldsymbol{g}/\tilde{\epsilon}}, K = M \odot e^{-\tilde{C}/\tilde{\epsilon}}, \tilde{\pi} = \text{diag}(\boldsymbol{u}) K \text{diag}(\boldsymbol{v})$
  $\tilde{\pi} = \sqrt{\zeta(\tilde{\gamma})/\zeta(\tilde{\pi})}\tilde{\pi}$
end while
return $\tilde{\gamma}$

Obviously, $M^{\boldsymbol{pq}} \odot \pi$ is in $\Pi(\boldsymbol{p}, \boldsymbol{q}; M^{\boldsymbol{pq}})$ and $\sum_{i,j} M^{\boldsymbol{pq}}_{i,j} \pi_{i,j}(\alpha C_{i,j} + (1-\alpha)G_{i,j}) = 0$. Therefore, $\mathcal{S}_{krk}(\boldsymbol{p}, \boldsymbol{q}) = 0$.

(b) We denote π^* as the optimal solution of problem (15). If $\mathcal{S}_{krk}(\boldsymbol{p}, \boldsymbol{q}) = 0$, we have $\langle M^{\boldsymbol{pq}} \odot \pi^*, C\rangle_F = 0$. This means that the KP problem $\min_{\pi\in\Pi(\boldsymbol{p},\boldsymbol{q})} \langle \pi, C\rangle_F = 0$. Using the Proposition 2.2 in (Peyré et al., 2019), we have $\boldsymbol{p} = \boldsymbol{q}$.

(2) Prove that $\mathcal{S}_{krk}(\boldsymbol{p}, \boldsymbol{q}) = \mathcal{S}_{krk}(\boldsymbol{q}, \boldsymbol{p})$.

From the definition of mask matrix in Proposition 7, we have $M^{\boldsymbol{pq}}_{i,j} = M^{\boldsymbol{qp}}_{j,i}$. C and G are symmetric because c and d are distances. For any $\pi \in \Pi(\boldsymbol{p}, \boldsymbol{q}; M^{\boldsymbol{pq}})$, we define π' as $\pi'_{i,j} = \pi_{j,i}$, and then $\pi' \in \Pi(\boldsymbol{q}, \boldsymbol{p}; M^{\boldsymbol{qp}})$. Then, we have

$$\begin{aligned}
\mathcal{S}_{krk}(\boldsymbol{p}, \boldsymbol{q}) =& \min_{\pi\in\Pi(\boldsymbol{p},\boldsymbol{q};M^{\boldsymbol{pq}})} \sum_{i,j} M^{\boldsymbol{pq}}_{i,j} \pi_{i,j}(\alpha C_{i,j} + (1-\alpha)G_{i,j}) \\
=& \min_{\pi\in\Pi(\boldsymbol{p},\boldsymbol{q};M^{\boldsymbol{pq}})} \sum_{i,j} M^{\boldsymbol{qp}}_{j,i} \pi_{i,j}(\alpha C_{j,i} + (1-\alpha)G_{j,i}) \\
=& \min_{\pi'\in\Pi(\boldsymbol{q},\boldsymbol{p};M^{\boldsymbol{qp}})} \sum_{j,i} M^{\boldsymbol{qp}}_{j,i} \pi'_{j,i}(\alpha C_{j,i} + (1-\alpha)G_{j,i}) \\
=& \mathcal{S}_{krk}(\boldsymbol{q}, \boldsymbol{p}).
\end{aligned} \tag{53}$$

(3) Prove that $\mathcal{S}_{krk}(\boldsymbol{p}, \boldsymbol{q}) \le \mathcal{S}_{krk}(\boldsymbol{p}, \boldsymbol{r}) + \mathcal{S}_{krk}(\boldsymbol{r}, \boldsymbol{q})$ *for any* $\boldsymbol{r} = \sum_{k=1}^m r_k \delta_{z_k} \in \mathcal{P}^{\mathcal{X}}_{\mathcal{I}}$.

Let $M^{\boldsymbol{pr}} \odot \pi^{\boldsymbol{pr}}$ and $M^{\boldsymbol{rq}} \odot \pi^{\boldsymbol{rq}}$ be the optimal transport plans corresponding to $\mathcal{S}_{krk}(\boldsymbol{p}, \boldsymbol{r})$ and $\mathcal{S}_{krk}(\boldsymbol{r}, \boldsymbol{q})$, respectively. We define

$$\tilde{\gamma} = (M^{\boldsymbol{pr}} \odot \pi^{\boldsymbol{pr}}) \text{diag}\left(\frac{1}{\tilde{\boldsymbol{r}}}\right) (M^{\boldsymbol{rq}} \odot \pi^{\boldsymbol{rq}}), \tag{54}$$

where the element $\tilde{r}_j$ of $\tilde{\boldsymbol{r}}$ is r_j if $r_j > 0$, and 1 otherwise. We notice that

$$\tilde{\gamma}\mathbb{1}_m = (M^{\boldsymbol{pr}} \odot \pi^{\boldsymbol{pr}})\mathrm{diag}\left(\frac{1}{\tilde{\boldsymbol{r}}}\right)\boldsymbol{r} = (M^{\boldsymbol{pr}} \odot \pi^{\boldsymbol{pr}})\left(\frac{\boldsymbol{r}}{\tilde{\boldsymbol{r}}}\right) = (M^{\boldsymbol{pr}} \odot \pi^{\boldsymbol{pr}})\tilde{\mathbb{1}}_m, \tag{55}$$

where the j-th location of $\tilde{\mathbb{1}}_m$ is 1 if $r_j > 0$, otherwise 0. Note that for j such that $r_j = 0$, we have $\sum_i (M^{\boldsymbol{pr}} \odot \pi^{\boldsymbol{pr}})_{i,j} = r_j = 0$, which implies $(M^{\boldsymbol{pr}} \odot \pi^{\boldsymbol{pr}})_{i,j} = 0$ for any i. Hence,

$$\tilde{\gamma}\mathbb{1}_m = (M^{\boldsymbol{pr}} \odot \pi^{\boldsymbol{pr}})\tilde{\mathbb{1}}_m = (M^{\boldsymbol{pr}} \odot \pi^{\boldsymbol{pr}})\mathbb{1}_m = \boldsymbol{p}. \tag{56}$$

Similarly, $\tilde{\gamma}^\top \mathbb{1}_m = \boldsymbol{q}$. Since the indexes of paired keypoints across any two distribution in $\mathcal{P}_{\mathcal{I}}^{\mathcal{X}}$ are the same, we have for any $i_u \in \mathcal{I}$, the i_u-th row and column of $M^{\boldsymbol{pr}}$ and $M^{\boldsymbol{rq}}$ are zeros except for that $M^{\boldsymbol{pr}}_{i_u,i_u} = M^{\boldsymbol{rq}}_{i_u,i_u} = 1$. So the i_u-th row and column of $\tilde{\gamma}$ are zeros except for $\tilde{\gamma}_{i_u,i_u}$. Then, we can write $\tilde{\gamma} = M^{\boldsymbol{pq}} \odot \gamma$ with $\gamma \in \mathbb{R}_+^{m\times m}$. Therefore, we have $\gamma \in \Pi(\boldsymbol{p}, \boldsymbol{q}; M^{\boldsymbol{pq}})$. The triangle inequality then follows from

$$\begin{aligned}
&\mathcal{S}_{krk}(\boldsymbol{p}, \boldsymbol{q}) = \min_{\pi\in\Pi(\boldsymbol{p},\boldsymbol{q};M^{\boldsymbol{pq}})} \sum_{i,j} M^{\boldsymbol{pq}}_{i,j}\pi_{i,j}(\alpha c(x_i, y_j) + (1-\alpha)d(R_i^s, R_j^t)) \\
\leq &\sum_{i,j} \tilde{\gamma}_{i,j}(\alpha c(x_i, y_j) + (1-\alpha)d(R_i^s, R_j^t)) \\
= &\sum_{i,j}(\alpha c(x_i, y_j) + (1-\alpha)d(R_i^s, R_j^t)) \sum_k \frac{(M^{\boldsymbol{pr}} \odot \pi^{\boldsymbol{pr}})_{i,k}(M^{\boldsymbol{rq}} \odot \pi^{\boldsymbol{rq}})_{k,j}}{\tilde{r}_k} \\
\leq &\sum_{i,k,j}(\alpha(c(x_i, z_k) + c(z_k, y_j)) + (1-\alpha)(d(R_i^s, R_k^r) + d(R_k^r, R_j^t))\frac{(M^{\boldsymbol{pr}} \odot \pi^{\boldsymbol{pr}})_{i,k}(M^{\boldsymbol{rq}} \odot \pi^{\boldsymbol{rq}})_{k,j}}{\tilde{r}_k} \\
= &\sum_{i,k,j}(\alpha c(x_i, z_k) + (1-\alpha)d(R_i^s, R_k^r))\frac{(M^{\boldsymbol{pr}} \odot \pi^{\boldsymbol{pr}})_{i,k}(M^{\boldsymbol{rq}} \odot \pi^{\boldsymbol{rq}})_{k,j}}{\tilde{r}_k} \\
+ &\sum_{i,k,j}(\alpha c(z_k, y_j) + (1-\alpha)d(R_k^r, R_j^t))\frac{(M^{\boldsymbol{pr}} \odot \pi^{\boldsymbol{pr}})_{i,k}(M^{\boldsymbol{rq}} \odot \pi^{\boldsymbol{rq}})_{k,j}}{\tilde{r}_k} \\
= &\sum_{i,k}(\alpha c(x_i, z_k) + (1-\alpha)d(R_i^s, R_k^r))(M^{\boldsymbol{pr}} \odot \pi^{\boldsymbol{pr}})_{i,k} \\
+ &\sum_{k,j}(\alpha c(z_k, y_j) + (1-\alpha)d(R_k^r, R_j^t))(M^{\boldsymbol{rq}} \odot \pi^{\boldsymbol{rq}})_{k,j} \\
= &\mathcal{S}_{krk}(\boldsymbol{p}, \boldsymbol{r}) + \mathcal{S}_{krk}(\boldsymbol{r}, \boldsymbol{q}),
\end{aligned} \tag{57}$$

where z_k is the support point of $\boldsymbol{r}$ and R_k^r is the relation of z_k to the keypoints of $\boldsymbol{r}$.

■

C.2 Proof of Theorem 6

(a) If $\boldsymbol{p}$ and $\boldsymbol{q}$ are isomorphic, for any $i \in [m]$ and any $i_u \in \mathcal{I}$, we have $c(x_i, x_{i_u}) = c'(y_{\sigma(i)}, y_{\sigma(i_u)}) = c'(y_{\sigma(i)}, y_{i_u})$, implying that $R_i^s = R_{\sigma(i)}^t$. We define π as $\pi_{i,j} = p_i$ if $j = \sigma(i)$,

otherwise 0. We then have

$$\begin{aligned}
&\sum_{i,j}\alpha\Big(\sum_{k,l}(M^{\boldsymbol{pq}}\odot\pi)_{i,j}(M^{\boldsymbol{pq}}\odot\pi)_{k,l}|C^s_{i,k}-C^t_{j,l}|^2\Big)+(1-\alpha)(M^{\boldsymbol{pq}}\odot\pi)_{i,j}G_{i,j}\\
=&\sum_i\Big[\alpha\Big(\sum_k(M^{\boldsymbol{pq}}\odot\pi)_{i,\sigma(i)}(M^{\boldsymbol{pq}}\odot\pi)_{k,\sigma(k)}|C^s_{i,k}-C^t_{\sigma(i),\sigma(k)}|^2\Big)\\
&\qquad+(1-\alpha)(M^{\boldsymbol{pq}}\odot\pi)_{i,\sigma(i)}G_{i,\sigma(i)}\Big]\\
=&\sum_i\Big[\alpha\Big(\sum_k(M^{\boldsymbol{pq}}\odot\pi)_{i,\sigma(i)}(M^{\boldsymbol{pq}}\odot\pi)_{k,\sigma(k)}|c(x_i,x_k)-c'(x_{\sigma(i)},x_{\sigma(k)})|^2\Big)\\
&\qquad+(1-\alpha)(M^{\boldsymbol{pq}}\odot\pi)_{i,\sigma(i)}d(R^s_i,R^t_{\sigma(i)})\Big]\\
=&0.
\end{aligned}\tag{58}$$

This implies $\mathcal{S}_{krg}(\boldsymbol{p},\boldsymbol{q})=0$.

(b) Let $(M^{\boldsymbol{pq}})\odot\pi^*$ be the optimal transport plan corresponding to $\mathcal{S}_{krg}(\boldsymbol{p},\boldsymbol{q})$. If $\mathcal{S}_{krg}(\boldsymbol{p},\boldsymbol{q})=0$, we have

$$\sum_{i,j,k,l}(M^{\boldsymbol{pq}}\odot\pi^*)_{i,j}(M^{\boldsymbol{pq}}\odot\pi^*)_{k,l}|C^s_{i,k}-C^t_{j,l}|^2=0.\tag{59}$$

This indicates that the Gromov-Wasserstein distance

$$\min_{\pi\in\Pi(\boldsymbol{p},\boldsymbol{q})}\sum_{i,j,k,l}\pi_{i,j}\pi_{k,l}|C^s_{i,k}-C^t_{j,l}|^2=0.\tag{60}$$

By virtue to Gromov-Wasserstein properties in Mémoli (2011), there exists a bijection $\sigma:[m]\longmapsto[m]$ such that $c(x_i,x_k)=c'(y_{\sigma(i)},y_{\sigma(k)})$, and $p_i=q_{\sigma(i)}$.

■

Appendix D. Proof and Generalization of Theorem 7

For convenience of understanding, Theorem 7 in the paper is for the case that $p_i=q_j$ for all $(i,j)\in\mathcal{K}$. In this appendix, we provide and prove Theorem 11, a generalization of Theorem 7 in the paper, for the general case that there could exist some $(i,j)\in\mathcal{K}$ such that $p_i\neq q_j$.

Theorem 11 *Suppose $A>0$, $\xi>0$, $\sum_{i\in\mathcal{I}}p_i+\max\{q_{\kappa(i)}-p_i,0\}<s$, and $\sum_{j\in\mathcal{J}}q_j+\max\{p_{\kappa'(j)}-q_j,0\}<s$, then the optimal transport plan $M\odot\pi^*$ of the partial KPG-RL model in Eq. (17) is the m-by-n block in the upper left corner of the optimal transport plan $\bar{M}\odot\bar{\pi}^*$ of problem $min_{\bar{\pi}\in\Pi(\bar{\boldsymbol{p}},\bar{\boldsymbol{q}};\bar{M})}\langle\bar{M}\odot\bar{\pi},\bar{G}\rangle_F$.*

The definitions of $\kappa(i)$ and $\kappa'(j)$ are given in Appendix A. The condition $\sum_{i\in\mathcal{I}}p_i+\max\{q_{\kappa(i)}-p_i,0\}<s$ implies that the sum of the mass ($\sum_{i\in\mathcal{I}}p_i$) of the source key points and the mass ($\sum_{i\in\mathcal{I}}\max\{q_{\kappa(i)}-p_i,0\}$) of the other source points apart from the key points that should be transported to the target key points is less than s. The condition $\sum_{j\in\mathcal{J}}q_j+\max\{p_{\kappa'(j)}-q_j,0\}<s$ implies that the sum of the mass ($\sum_{j\in\mathcal{J}}q_j$) of target keypoints and the mass ($\sum_{j\in\mathcal{J}}\max\{p_{\kappa'(j)}-q_j,0\}$) of the other target points apart from keypoints received from source keypoints is less than s. The two conditions are reasonable to guarantee the admissible solutions of problem (17). If $p_i=q_j$ for all $(i,j)\in\mathcal{K}$ (*i.e.*, $p_i=q_{\kappa(i)},\forall i\in\mathcal{I}$ and $q_j=p_{\kappa'(j)},\forall j\in\mathcal{J}$), Theorem 11 degenerates into Theorem 7.

D.1 Proof of Theorem 11

We denote $\ddot{\pi} = \bar{\pi}^*_{1:m,1:n}$ and $t = \bar{\pi}^*_{m+1,n+1}$. To prove Theorem 11, we first make some preparations and then perform the following three steps. In Step 1, we prove that $t = 0$. In Step 2, we prove that $\ddot{\pi} \in \Pi^s(\boldsymbol{p}, \boldsymbol{q}; M)$, which means that $\ddot{\pi}$ is a feasible solution to problem (17). In Step 3, we prove that $M \odot \ddot{\pi}$ is the optimal transport plan of problem (17). We next detail these steps.

Preparations.
Since $\bar{\pi}^* \in \Pi(\bar{\boldsymbol{p}}, \bar{\boldsymbol{q}}; \bar{M})$ and $t = \bar{\pi}^*_{m+1,n+1}$, we have

$$\begin{aligned}\mathbb{1}^\top_{m+1}(\bar{M} \odot \bar{\pi}^*)\mathbb{1}_{n+1} =& \begin{bmatrix}\mathbb{1}^\top_m & 1\end{bmatrix}\left(\begin{bmatrix}M & \boldsymbol{a}\\ \boldsymbol{b} & 1\end{bmatrix} \odot \begin{bmatrix}\ddot{\pi} & \bar{\pi}^*_{1:m,n+1}\\ \bar{\pi}^*_{m+1,1:n} & \bar{\pi}^*_{m+1,n+1}\end{bmatrix}\right)\begin{bmatrix}\mathbb{1}_n\\ 1\end{bmatrix}\\ =& \mathbb{1}^\top_m(M \odot \ddot{\pi})\mathbb{1}_n + \sum_{i=1}^m a_i\bar{\pi}^*_{i,n+1} + \sum_{j=1}^n b_j\bar{\pi}^*_{m+1,j} + t.\end{aligned} \tag{61}$$

Meanwhile, we have

$$\mathbb{1}^\top_{m+1}(\bar{M} \odot \bar{\pi}^*)\mathbb{1}_{n+1} = \mathbb{1}^\top_{m+1}\bar{\boldsymbol{p}} = \|\bar{\boldsymbol{p}}\|_1 = \|\boldsymbol{p}\|_1 + \|\boldsymbol{q}\|_1 - s, \tag{62}$$

$$\sum_{i=1}^m a_i\bar{\pi}^*_{i,n+1} + t = \|\boldsymbol{p}\|_1 - s, \tag{63}$$

and

$$\sum_{j=1}^n b_j\bar{\pi}^*_{m+1,j} + t = \|\boldsymbol{q}\|_1 - s. \tag{64}$$

Combining the above four equations, we have

$$\mathbb{1}^\top_m(M \odot \ddot{\pi})\mathbb{1}_n + \|\boldsymbol{p}\|_1 + \|\boldsymbol{q}\|_1 - 2s - t = \|\boldsymbol{p}\|_1 + \|\boldsymbol{q}\|_1 - s. \tag{65}$$

Therefore,

$$\mathbb{1}^\top_m(M \odot \ddot{\pi})\mathbb{1}_n = s + t. \tag{66}$$

Step 1: prove that $t = \bar{\pi}^*_{m+1,n+1} = 0$.
First, we have

$$\begin{aligned}\langle \bar{M} \odot \bar{\pi}^*, \bar{G}\rangle_F =& \sum_{i=1}^m\sum_{j=1}^n M_{i,j}\bar{\pi}^*_{i,j}G_{i,j} + \xi\sum_{i=1}^m a_i\bar{\pi}^*_{i,n+1}\\ &+\xi\sum_{j=1}^n b_j\bar{\pi}^*_{m+1,j} + (2\xi + A)\bar{\pi}^*_{m+1,n+1}\\ =& \sum_{i=1}^m\sum_{j=1}^n M_{i,j}\bar{\pi}^*_{i,j}G_{i,j} + \xi(\|\boldsymbol{p}\|_1 + \|\boldsymbol{q}\|_1 - 2s - 2t) + (2\xi + A)t\\ =& \sum_{i=1}^m\sum_{j=1}^n M_{i,j}\bar{\pi}^*_{i,j}G_{i,j} + \xi(\|\boldsymbol{p}\|_1 + \|\boldsymbol{q}\|_1 - 2s) + At.\end{aligned} \tag{67}$$

Suppose $\bar{\pi}^*_{m+1,n+1} > 0$, we next construct a solution γ such that $\gamma_{m+1,n+1} = 0$ and leads to conflict. We randomly select a set $S = \{(i,j)|\bar{\pi}^*_{i,j} > 0, i \le m, j \le n, i \notin \mathcal{I}, j \notin \mathcal{J}\}$ and a index pair (i_0, j_0) satisfying the constraints of elements in S, such that $\sum_{(i,j)\in S} \bar{\pi}^*_{i,j} \le t$ and $\sum_{(i,j)\in S} \bar{\pi}^*_{i,j} + \bar{\pi}^*_{i_0,j_0} > t$. In the rest part of this section, the involved i, j satisfy $i \le m$ and $j \le n$. Such non-empty S and (i_0, j_0) always exist, because

$$\begin{aligned}
\mathbb{1}_m^\top (M \odot \ddot{\pi}) \mathbb{1}_n &= \sum_{i=1}^{m}\sum_{j=1}^{n} M_{i,j}\bar{\pi}^*_{i,j} = \sum_{i\in\mathcal{I},j} \bar{\pi}^*_{i,j} + \sum_{i\notin\mathcal{I},j\in\mathcal{J}} M_{i,j}\bar{\pi}^*_{i,j} + \sum_{i\notin\mathcal{I},j\notin\mathcal{J}} \bar{\pi}^*_{i,j} \\
&= \sum_{i\in\mathcal{I},j} \bar{\pi}^*_{i,j} + \sum_{i\notin\mathcal{I},j\in\mathcal{J}} M_{i,j}\bar{\pi}^*_{i,j} + \sum_{i\notin\mathcal{I},j\notin\mathcal{J}} \bar{\pi}^*_{i,j} \\
&= \sum_{i\in\mathcal{I},j} \bar{\pi}^*_{i,j} + \sum_{i\notin\mathcal{I},i'\in\mathcal{I}} M_{i,\kappa(i')}\bar{\pi}^*_{i,\kappa(i')} + \sum_{i\notin\mathcal{I},j\notin\mathcal{J}} \bar{\pi}^*_{i,j} \\
&\le \sum_{i\in I} p_i + \sum_{i'\in\mathcal{I},i\neq i'} M_{i,\kappa(i')}\bar{\pi}^*_{i,\kappa(i')} + \sum_{i\notin\mathcal{I},j\notin\mathcal{J}} \bar{\pi}^*_{i,j} \\
&= \sum_{i\in I} p_i + \sum_{i'\in\mathcal{I}} \max\{q_{\kappa(i')} - p_{i'}, 0\} + \sum_{i\notin\mathcal{I},j\notin\mathcal{J}} \bar{\pi}^*_{i,j},
\end{aligned} \tag{68}$$

$\mathbb{1}_m^\top (M \odot \ddot{\pi}) \mathbb{1}_n = s + t$, and $\sum_{i\in\mathcal{I}} p_i + \max\{q_{\kappa(i)} - p_i, 0\} < s$, we have $\sum_{i\notin\mathcal{I},j\notin\mathcal{J}} \bar{\pi}^*_{ij} > t$. We now move the mass of index pairs in S and (i_0, j_0) to their marginal such that a total mass of t is moved. Specifically, for $(i,j) \in S$, we set $\gamma_{i,j} = 0, \gamma_{i,n+1} = \bar{\pi}^*_{i,n+1} + \bar{\pi}^*_{i,j}, \gamma_{m+1,j} = \bar{\pi}^*_{m+1,j} + \bar{\pi}^*_{i,j}$. For (i_0, j_0), we set $\gamma_{i_0,j_0} = \bar{\pi}^*_{i_0,j_0} - (t - \sum_{i\notin\mathcal{I},j\notin\mathcal{J}} \bar{\pi}^*_{i,j}), \gamma_{i_0,n+1} = \bar{\pi}^*_{i_0,n+1} + (t - \sum_{i\notin\mathcal{I},j\notin\mathcal{J}} \bar{\pi}^*_{i,j}), \gamma_{m+1,j_0} = \bar{\pi}^*_{m+1,j_0} - (t - \sum_{i\notin\mathcal{I},j\notin\mathcal{J}} \bar{\pi}^*_{i,j})$. For $(i,j) \notin S$, we set $\gamma_{i,j} = \bar{\pi}^*_{i,j}, \gamma_{i,n+1} = \bar{\pi}^*_{i,n+1}, \gamma_{m+1,j} = \bar{\pi}^*_{m+1,j}$. It is easy to verify that $\gamma \in \Pi(\bar{\boldsymbol{p}}, \bar{\boldsymbol{q}}; \bar{M})$. Similar to Eq. (67), we have

$$\langle \bar{M} \odot \gamma, \bar{G} \rangle_F = \sum_{i=1}^{m}\sum_{j=1}^{n} M_{i,j}\gamma_{i,j}G_{i,j} + \xi(\|\boldsymbol{p}\|_1 + \|\boldsymbol{q}\|_1 - 2s). \tag{69}$$

Using the optimality of $\bar{M} \odot \bar{\pi}^*$, we have

$$\langle \bar{M} \odot \gamma, \bar{G} \rangle_F - \langle \bar{M} \odot \bar{\pi}^*, \bar{G} \rangle_F = \sum_{i=1}^{m}\sum_{j=1}^{n} M_{i,j}(\gamma_{i,j} - \bar{\pi}^*_{i,j})G_{i,j} - At > 0. \tag{70}$$

From the definition of γ, we can see that $\gamma_{i,j} \le \bar{\pi}^*_{i,j}$, and thus $\sum_{i=1}^{m}\sum_{j=1}^{n} M_{i,j}(\gamma_{i,j} - \bar{\pi}^*_{i,j})G_{i,j} \le 0$. Hence, from Eq. (70), we have $A < 0$, which contradicts the assumption that $A > 0$. Therefore, $t = \bar{\pi}^*_{m+1,n+1} = 0$ holds.

Step 2: prove that $\ddot{\pi}$ is a feasible solution of problem in Eq. (17).

We verify the constraints as follows.

(1) Since $\bar{\pi}^* \ge 0$, we have $\ddot{\pi} \ge 0$.

(2) $(\bar{M} \odot \bar{\pi}^*)\mathbb{1}_{n+1} = \begin{bmatrix} M \odot \ddot{\pi} & \boldsymbol{a} \odot \bar{\pi}^*_{1:m,n+1} \\ \boldsymbol{b} \odot \bar{\pi}^*_{m+1,1:n} & 0 \end{bmatrix} \begin{bmatrix} \mathbb{1}_n \\ 1 \end{bmatrix} = \begin{bmatrix} \boldsymbol{p} \\ \|\boldsymbol{q}\|_1 - s \end{bmatrix}$, then $(M \odot \ddot{\pi})\mathbb{1}_n + \boldsymbol{a} \odot \bar{\pi}^*_{1:m,n+1} = \boldsymbol{p}$, and $(M \odot \ddot{\pi})\mathbb{1}_n \le \boldsymbol{p}$.

(3) Similarly, from $\mathbb{1}_{m+1}^\top (\bar{M} \odot \bar{\pi}^*) = (\boldsymbol{q}, \|\boldsymbol{q}\|_1 - s)^\top$, we have $\mathbb{1}_m^\top (M \odot \ddot{\pi}) \le \boldsymbol{q}$.

(4) $\mathbb{1}_m^\top(M \odot \ddot{\pi})\mathbb{1}_n = s$ holds because $t = 0$ as in Step 1.

(5) $\forall i \in \mathcal{I}, (\bar{M} \odot \bar{\pi}^*)_{i,:}\mathbb{1}_{n+1} = (M \odot \ddot{\pi})_{i,:}\mathbb{1}_n + a_i\bar{\pi}^*_{i,n+1} = p_i$. Since $a_i = 0$, we have $(M \odot \ddot{\pi})_{1,:}\mathbb{1}_n = p_i$.

(6) $\forall j \in \mathcal{J}, \mathbb{1}_{m+1}^\top(\bar{M} \odot \bar{\pi}^*)_{:,j} = \mathbb{1}_m^\top(M \odot \ddot{\pi})_{:,j} + b_j\bar{\pi}^*_{m+1,j} = q_j$. Since $b_j = 0$, we have $\mathbb{1}_m^\top(M \odot \ddot{\pi})_{:,j} = q_j$.

Therefore, we have $\ddot{\pi} \in \Pi^s(\boldsymbol{p}, \boldsymbol{q}; M)$, and $\ddot{\pi}$ is a feasible solution to the problem in Eq. (17).

Step 3: prove that $M \odot \ddot{\pi}$ is the optimal transport plan of problem in Eq. (17).

Suppose there exist a transport plan $M \odot \gamma$ with $\gamma \in \Pi^s(\boldsymbol{p}, \boldsymbol{q}; M)$ such that

$$\sum_{i=1}^{m}\sum_{j=1}^{n} M_{i,j}\gamma_{i,j}G_{i,j} < \sum_{i=1}^{m}\sum_{j=1}^{n} M_{i,j}\ddot{\pi}_{i,j}G_{i,j}.$$

We construct $\bar{\gamma}$ as follows. For $i \le m, j \le n$, $\bar{\gamma}_{i,j} = \gamma_{i,j}$. $\bar{\gamma}_{i,n+1} = p_i - \sum_{j=1}^{n}\gamma_{i,j}, \forall i \le m$. $\bar{\gamma}_{m+1,j} = q_j - \sum_{i=1}^{n}\gamma_{i,j}, \forall j \le n$. $\bar{\gamma}_{m+1,n+1} = 0$. Easily, we can verify that $\bar{\gamma}$ is in $\Pi(\bar{\boldsymbol{p}}, \bar{\boldsymbol{q}}; \bar{M})$. Meanwhile,

$$\begin{aligned}\langle \bar{M} \odot \bar{\gamma}, \bar{G}\rangle_F =& \sum_{i=1}^{m}\sum_{j=1}^{n} M_{i,j}\gamma_{i,j}C_{i,j} + \xi(\|\boldsymbol{p}\|_1 + \|\boldsymbol{q}\|_1 - 2s)\\ <& \sum_{i=1}^{m}\sum_{j=1}^{n} M_{i,j}\ddot{\pi}_{i,j}G_{i,j} + \xi(\|\boldsymbol{p}\|_1 + \|\boldsymbol{q}\|_1 - 2s)\\ =& \langle \bar{M} \odot \bar{\pi}^*, \bar{G}\rangle_F.\end{aligned} \tag{71}$$

This contradicts the fact that $\bar{M} \odot \bar{\pi}^*$ is the optimal transport plan of problem $\min_{\bar{\pi} \in \Pi(\bar{\boldsymbol{p}}, \bar{\boldsymbol{q}}; \bar{M})}\langle \bar{M} \odot \bar{\pi}, \bar{G}\rangle_F$. Therefore, $M \odot \ddot{\pi}$ is the optimal transport plan of problem in Eq. (17).

■

Appendix E. Proof of Theorem 8

We rewrite the χ^2-regularized model as

$$\begin{aligned}&\min_{\pi} \sum_{i,j} M_{i,j}\pi_{i,j}G_{i,j} + \epsilon\sum_{i,j}\frac{(M_{i,j}\pi_{i,j})^2}{p_i q_j}\\ &\text{s.t. } \sum_j M_{i,j}\pi_{i,j} = p_i, \sum_i M_{i,j}\pi_{i,j} = q_j, \pi_{i,j} \ge 0.\end{aligned} \tag{72}$$

The Lagrange function is

$$\begin{aligned}L(\pi, \phi, \psi) =& \sum_{i,j} M_{i,j}(G_{i,j} - \phi(x_i) - \psi(y_j))\pi_{i,j} + \epsilon\sum_{i,j}\frac{M_{i,j}\pi_{i,j}^2}{p_i q_j}\\ &+ \sum_i \phi(x_i)p_i + \sum_j \psi(y_j)q_j,\end{aligned} \tag{73}$$

where we utilize $M_{i,j}^2 = M_{i,j}$. Minimizing $L(\pi, \phi, \psi)$ *w.r.t.* π ($\pi_{i,j} \geq 0$) is equivalent to

$$\sum_{i,j} \min_{\pi_{i,j} \geq 0} \left\{ M_{i,j}(G_{i,j} - \phi(x_i) - \psi(y_j))\pi_{i,j} + \epsilon \frac{M_{i,j}\pi_{i,j}^2}{p_i q_j} \right\}. \tag{74}$$

If $M_{i,j} = 0$, the minimizer $\pi_{i,j}$ could be arbitrary non-negative value. If $M_{i,j} = 1$, the minimizer is $\pi_{i,j} = \frac{1}{2\epsilon}(-G_{i,j} + \phi(x_i) + \psi(y_j))_+ p_i q_j$, where $a_+ = \max\{a, 0\}$. Hence, $M_{i,j}\pi_{i,j} = \frac{1}{2\epsilon} M_{i,j}(-G_{i,j} + \phi(x_i) + \psi(y_j))_+ p_i q_j$. The dual problem is

$$\begin{aligned}
&\max_{\phi,\psi} \min_{\pi} L(\pi, \phi, \psi) \\
=&\max_{\phi,\psi} \sum_i \phi(x_i)p_i + \sum_j \psi(y_j)q_j - \frac{1}{2\epsilon} \sum_{i,j} M_{i,j}[(-G_{i,j} + \phi(x_i) + \psi(y_j))_+]^2 p_i q_j \\
&+ \frac{1}{4\epsilon} \sum_{i,j} M_{i,j}(-G_{i,j} + \phi(x_i) + \psi(y_j))_+^2 p_i q_j \\
=&\max_{\phi,\psi} \sum_i \phi(x_i)p_i + \sum_j \psi(y_j)q_j - \frac{1}{4\epsilon} \sum_{i,j} M_{i,j}(-G_{i,j} + \phi(x_i) + \psi(y_j))_+^2 p_i q_j.
\end{aligned} \tag{75}$$

By the strong duality, the optimal solution ϕ^*, ψ^* satisfies

$$(M \odot \pi^*)_{i,j} = \frac{1}{2\epsilon} M_{i,j}(-G_{i,j} + \phi^*(x_i) + \psi^*(y_j))_+ p_i q_j. \tag{76}$$

■

Appendix F. Implementation Details for I2I Translation Experiment

The experiment consists of two steps. In the first step, we learn the potentials ϕ and ψ based on Eq. (23) and compute the transport plan as Eq. (24). In the second step, we learn the manifold barycentric projection T_{MBP} by Eq. (26) or the manifold sampling map T_{MSP} by Eq. (28). During training, at each step, we randomly sample a small set of unpaired images and merge them with all paired images to construct a mini-batch. We assign uniform mass to all samples within the mini-batch.

F.1 Details for the First Step

Architectures of the potentials $\boldsymbol{\phi}$ and $\boldsymbol{\psi}$ in Eq. (23). Both ϕ and ψ consist of a feature extractor, followed by an output head. The output head is constructed as follows: FC(1024,1024) → ReLU → FC(1024,1024) → ReLU → FC(1024,1024) → ReLU → FC(1024,1024) → ReLU → FC(1024,1), where "FC(a, b)" is the Fully-Connected layer with input/output channel of a/b. The feature extractor is pretrained and fixed. For digits, we train autoencoders for source and target images respectively, and take the encoder part of the autoencoder as the feature extractor. The architecture of the encoder part is as follows: Conv(1,32,3,1) → GN(4) → ReLU → Conv(32,64,3,2) → GN(32) → ReLU → Conv(64,128,3,2) → GN(32) → ReLU → Conv(128,256,3,2) → GN(32) → L2-Norm. The architecture of the decoder part of the autoencoder is as follows: Tconv(256,128,3,2) → GN(32) → ReLU → Tconv(128,64,3,2)

$\rightarrow$ GN(32) $\rightarrow$ ReLU $\rightarrow$ Tconv(64,32,3,2) $\rightarrow$ GN(32) $\rightarrow$ ReLU $\rightarrow$ Tconv(32,1,3,1) $\rightarrow$ Sigmoid. "Conv(a, b, k, s)" and "Tconv(a,b,k,s)" are respectively the Convolutional layer and the Transposed-Convolutional layer, where a and b are the input and output channel respectively, the kernel size is $k \times k$, and the stride is s. "GN(m)" is the Group Normalization layer with m groups. "L2-Norm" is the L_2-normalization. For the natural animal images, the feature extractor is taken as the image encoder ("ViT-B/32") of CLIP (Radford et al., 2021).

Training details for learning the potentials $\boldsymbol{\phi}$ and $\boldsymbol{\psi}$ in Eq. (23). When training with Eq. (23), the optimization algorithm is Adam, the learning rate is 1e-5, and the batch size is 64. ϵ in Eq. (22) is set to 0.005.

F.2 Details for the Second Step

Architectures of $\boldsymbol{T'}$ in Eq. (26) *and $\boldsymbol{D}$ in Eq.* (27). For digits, the architectures of T' is as follows: Conv(1,64,4,2) $\rightarrow$ BN $\rightarrow$ ReLU $\rightarrow$ Conv(64,128,4,2) $\rightarrow$ BN $\rightarrow$ ReLU $\rightarrow$ Conv(128, 128,3,1) $\rightarrow$ BN $\rightarrow$ ReLU $\rightarrow$ Conv(128,128,3,1) $\rightarrow$ BN $\rightarrow$ ReLU $\rightarrow$ Tconv(128,64,4,2) $\rightarrow$ BN $\rightarrow$ ReLU $\rightarrow$ Tconv(64,1,4,2) $\rightarrow$ Sigmoid, where "BN" is the Batch Normalization layer. The architectures of D is as follows: Conv(1,64,4,2) $\rightarrow$ ReLU $\rightarrow$ Conv(64,128,4,2) $\rightarrow$ BN $\rightarrow$ ReLU $\rightarrow$ Conv(128,256,4,2) $\rightarrow$ BN $\rightarrow$ ReLU $\rightarrow$ Conv(256,1,3,0).

For natural animal images, to reduce the training burden, we map the images to the lantent feature space using the encoder of Stable Diffusion and then perform our MBP and MSP in the latent space. In the latent space, we construct T' and D using linear layers. Specifically, the architecture of T' is as follows: Linear(4096,2048) $\rightarrow$ ReLU $\rightarrow$ Linear(2048, 512) $\rightarrow$ ReLU $\rightarrow$ Linear(512, 128) $\rightarrow$ ReLU $\rightarrow$ Linear(128,512) $\rightarrow$ ReLU $\rightarrow$ Linear(512,2048) $\rightarrow$ ReLU $\rightarrow$ Linear(2048,4096). For MSP, the noise z is 128-dimensional which is added to the features after layer Linear(512, 128). The architecture of D for MBP is as follows: Linear(4096,2048) $\rightarrow$ ReLU $\rightarrow$ Linear(2048, 512) $\rightarrow$ ReLU $\rightarrow$ Linear(512, 1). For MSP, we concatenate source features and generated target features as the input of D, for which the input dimension of D is changed to 8192.

Training details for learning $\boldsymbol{T'}$. When training T' with Eq. (26), the optimization algorithm is Adam, the learning rate is 1e-4, and the batch size is 64. $\bar{d}$ is set to cosine distance in feature space.

Appendix G. Runtime and Peak Memory Scaling Comparison

We evaluate the scalability of our neural transport (based on dual formulation) approach against the primal formulation (solved via Sinkhorn), utilizing the same data setup as the "Toy experiments for evaluating KPG-RL-KP and KPG-RL-GW" in Section 6.1. We vary the sample size m (setting $m = n$) from $5,000$ to $80,000$, with ϵ fixed at 0.001 for both methods. The neural transport model is trained for 100 epochs. All experiments are conducted on a Tesla-V100 GPU (32GB). Table 13 reports the runtime and peak memory usage. We observe that as the sample size increases, Sinkhorn's algorithm seems to exhibit nearly exponential growth in both runtime and memory, encountering out-of-memory (OOM) errors when $m \geq 50,000$. In contrast, the neural transport approach maintains constant peak memory usage (≈ 0.02 GB) regardless of sample size, as it relies on mini-batch optimization rather than storing full coupling matrices. Furthermore, its runtime seems to scale linearly

Sample size		5000	10000	20000	30000	50000	80000
Time (s)	Sinkhorn	0.8	1.8	12.0	28.4	OOM	OOM
	Neural	21.5	41.6	87.6	132.8	217.3	346.9
Memory (GB)	Sinkhorn	0.68	2.25	8.94	20.12	OOM	OOM
	Neural	0.02	0.02	0.02	0.02	0.02	0.02

Table 13: Runtime (in seconds) and peak memory (in GB) scaling comparison between Sinkhorn's algorithm and neural transport method across different sample sizes (m). "OOM" indicates Out-of-Memory.

with sample size. These results indicate that while Sinkhorn's algorithm is computationally efficient for smaller sample sizes, the neural transport approach offers superior scalability for large-scale transport problems.

References

Michael S Albergo, Nicholas M. Boffi, and Eric Vanden-Eijnden. Stochastic interpolants: A unifying framework for flows and diffusions. *arXiv preprint arXiv:2303.08797*, abs/2303.08797, 2023.

David Alvarez-Melis and Nicolo Fusi. Geometric dataset distances via optimal transport. In *NeurIPS*, 2020.

David Alvarez-Melis and Tommi S Jaakkola. Gromov-wasserstein alignment of word embedding spaces. In *EMNLP*, 2018.

Martin Arjovsky, Soumith Chintala, and Léon Bottou. Wasserstein generative adversarial networks. In *ICML*, 2017.

Yogesh Balaji, Rama Chellappa, and Soheil Feizi. Robust optimal transport with applications in generative modeling and domain adaptation. In *NeurIPS*, 2020.

Aayush Bansal, Yaser Sheikh, and Deva Ramanan. Pixelnn: Example-based image synthesis. In *ICLR*, 2018.

Jessa Bekker and Jesse Davis. Learning from positive and unlabeled data: A survey. *Mach. Learn.*, 109(4):719–760, 2020.

Mathieu Blondel, Vivien Seguy, and Antoine Rolet. Smooth and sparse optimal transport. In *AISTATS*, 2018.

Nicolas Bonneel, Michiel Van De Panne, Sylvain Paris, and Wolfgang Heidrich. Displacement interpolation using lagrangian mass transport. In *SIGGRAPH Asia*, 2011.

Valentin De Bortoli, James Thornton, Jeremy Heng, and Arnaud Doucet. Diffusion schrödinger bridge with applications to score-based generative modeling. In *NeurIPS*, 2021.

Silvia Bucci, Mohammad Reza Loghmani, and Tatiana Tommasi. On the effectiveness of image rotation for open set domain adaptation. In *ECCV*, pages 422–438, 2020.

Luis A Caffarelli and Robert J McCann. Free boundaries in optimal transport and monge-ampere obstacle problems. *Ann. of Math.*, 171:673–730, 2010.

Kai Cao, Xiangqi Bai, Yiguang Hong, and Lin Wan. Unsupervised topological alignment for single-cell multi-omics integration. *Bioinformatics*, 36(Supplement_1):i48–i56, 2020.

Kai Cao, Yiguang Hong, and Lin Wan. Manifold alignment for heterogeneous single-cell multi-omics data integration using Pamona. *Bioinformatics*, 38(1):211–219, 2021.

Lawrence Cayton. Algorithms for manifold learning. *Univ. of California at San Diego Tech. Rep*, 12(1-17):1, 2005.

Laetitia Chapel, Mokhtar Z Alaya, and Gilles Gasso. Partial optimal transport with applications on positive-unlabeled learning. In *NeurIPS*, 2020.

Song Chen, Blue B Lake, and Kun Zhang. High-throughput sequencing of the transcriptome and chromatin accessibility in the same cell. *Nat. Biotechnol.*, 37(12):1452–1457, 2019.

Tianrong Chen, Guan-Horng Liu, and Evangelos Theodorou. Likelihood training of schrödinger bridge using forward-backward SDEs theory. In *ICLR*, 2022.

Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *ICML*, pages 1597–1607, 2020.

Lih Feng Cheow, Elise T Courtois, Yuliana Tan, Ramya Viswanathan, Qiaorui Xing, Rui Zhen Tan, Daniel SW Tan, Paul Robson, Yuin-Han Loh, Stephen R Quake, et al. Single-cell multimodal profiling reveals cellular epigenetic heterogeneity. *Nat. Methods*, 13(10):833–836, 2016.

Lenaic Chizat, Gabriel Peyré, Bernhard Schmitzer, and François-Xavier Vialard. An interpolating distance between optimal transport and fisher–rao metrics. *Found. Comput. Math.*, 18(1):1–44, 2018.

Jooyoung Choi, Sungwon Kim, Yonghyun Jeong, Youngjune Gwon, and Sungroh Yoon. Ilvr: Conditioning method for denoising diffusion probabilistic models. In *ICCV*, 2021.

Yunjey Choi, Youngjung Uh, Jaejun Yoo, and Jung-Woo Ha. Stargan v2: Diverse image synthesis for multiple domains. In *CVPR*, 2020.

Nicolas Courty, Rémi Flamary, Devis Tuia, and Alain Rakotomamonjy. Optimal transport for domain adaptation. *IEEE Trans. Pattern Anal. Mach. Intell.*, 39(9):1853–1865, 2017.

William R Crum, Thomas Hartkens, and DLG Hill. Non-rigid image registration: theory and practice. *Br. J. Radiol.*, 77(suppl_2):S140–S153, 2004.

Marco Cuturi. Sinkhorn distances: Lightspeed computation of optimal transport. In *NeurIPS*, 2013.

Pinar Demetci, Rebecca Santorella, Björn Sandstede, William Stafford Noble, and Ritambhara Singh. Scot: single-cell multi-omics alignment with optimal transport. *J. Comput. Biol.*, 29(1):3–18, 2022.

Arnaud Dessein, Nicolas Papadakis, and Jean-Luc Rouas. Regularized optimal transport and the rot mover's distance. *J. Mach. Learn. Res.*, 19(1):590–642, 2018.

Jeff Donahue, Yangqing Jia, Oriol Vinyals, Judy Hoffman, Ning Zhang, Eric Tzeng, and Trevor Darrell. Decaf: A deep convolutional activation feature for generic visual recognition. In *ICML*, 2014.

Zhen Fang, Jie Lu, Feng Liu, and Guangquan Zhang. Semi-supervised heterogeneous domain adaptation: Theory and algorithms. *IEEE Trans. Pattern Anal. Mach. Intell.*, 45(1): 1087–1105, 2023. doi: 10.1109/TPAMI.2022.3146234.

Charles Fefferman, Sanjoy Mitter, and Hariharan Narayanan. Testing the manifold hypothesis. *J. Amer. Math. Soc.*, 29(4):983–1049, 2016.

Alessio Figalli. The optimal partial transport problem. *Arch. Ration. Mech. Anal.*, 195(2): 533–560, 2010.

Charlie Frogner, Chiyuan Zhang, Hossein Mobahi, Mauricio Araya, and Tomaso A Poggio. Learning with a wasserstein loss. In *NeurIPS*, 2015.

Aude Genevay, Gabriel Peyré, and Marco Cuturi. Learning generative models with sinkhorn divergences. In *AISTATS*, 2018.

Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron C. Courville, and Yoshua Bengio. Generative adversarial nets. In *NeurIPS*, 2014.

Xiang Gu, Yucheng Yang, Wei Zeng, Jian Sun, and Zongben Xu. Keypoint-guided optimal transport with applications in heterogeneous domain adaptation. In *NeurIPS*, 2022.

Kevin Guittet. *Extended Kantorovich norms: a tool for optimization.* PhD thesis, INRIA, 2002.

Ishaan Gulrajani, Faruk Ahmed, Martin Arjovsky, Vincent Dumoulin, and Aaron C Courville. Improved training of wasserstein gans. In *NeurIPS*, 2017.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016.

Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. In *NeurIPS*, 2017.

Nhat Ho, XuanLong Nguyen, Mikhail Yurochkin, Hung Hai Bui, Viet Huynh, and Dinh Phung. Multilevel clustering via wasserstein means. In *ICML*, 2017.

Gao Huang, Chuan Guo, Matt J Kusner, Yu Sun, Fei Sha, and Kilian Q Weinberger. Supervised word mover's distance. In *NeurIPS*, 2016.

Viet Huynh, Nhat Ho, Nhan Dam, XuanLong Nguyen, Mikhail Yurochkin, Hung Bui, and Dinh Phung. On efficient multilevel clustering via wasserstein distances. *J. Mach. Learn. Res.*, 22(1):6421–6463, 2021.

Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros. Image-to-image translation with conditional adversarial networks. In *CVPR*, 2017.

Leonid V Kantorovich. On the translocation of masses. *Dokl. Akad. Nauk SSSR*, 37:199–201, 1942.

Prannay Khosla, Piotr Teterwak, Chen Wang, Aaron Sarna, Yonglong Tian, Phillip Isola, Aaron Maschinot, Ce Liu, and Dilip Krishnan. Supervised contrastive learning. In *NeurIPS*, 2020.

Taeksoo Kim, Moonsu Cha, Hyunsoo Kim, Jung Kwon Lee, and Jiwon Kim. Learning to discover cross-domain relations with generative adversarial networks. In *ICML*, 2017.

Alexander Korotin, Vage Egiazarian, Arip Asadulaev, Alexander Safin, and Evgeny Burnaev. Wasserstein-2 generative networks. In *ICLR*, 2021.

Alexander Korotin, Daniil Selikhanovych, and Evgeny Burnaev. Kernel neural optimal transport. In *ICLR*, 2023.

Matt Kusner, Yu Sun, Nicholas Kolkin, and Kilian Weinberger. From word embeddings to document distances. In *ICML*, 2015.

Khang Le, Huy Nguyen, Quang M Nguyen, Tung Pham, Hung Bui, and Nhat Ho. On robust optimal transport: Computational complexity and barycenter computation. In *NeurIPS*, 2021a.

Tam Le, Nhat Ho, and Makoto Yamada. Flow-based alignment approaches for probability measures in different spaces. In *AISTATS*, 2021b.

Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proc. IEEE*, 86(11):2278–2324, 1998.

John Lee, Max Dabagia, Eva Dyer, and Christopher Rozell. Hierarchical optimal transport for multimodal distribution alignment. In *NeurIPS*, 2019.

Shuang Li, Binhui Xie, Jiashu Wu, Ying Zhao, Chi Harold Liu, and Zhengming Ding. Simultaneous semantic alignment network for heterogeneous domain adaptation. In *ACM MM*, pages 3866–3874, 2020.

Matthias Liero, Alexander Mielke, and Giuseppe Savaré. Optimal entropy-transport problems and a new hellinger–kantorovich distance between positive measures. *Invent. Math.*, 211 (3):969–1117, 2018.

Chi-Heng Lin, Mehdi Azabou, and Eva L Dyer. Making transport more robust and interpretable by moving data through a small number of anchor points. In *ICML*, 2021.

Tianyi Lin, Nhat Ho, and Michael I Jordan. On the efficiency of entropic regularized algorithms for optimal transport. *J. Mach. Learn. Res.*, 23(137):1–42, 2022.

Xingchao Liu, Chengyue Gong, and qiang liu. Flow straight and fast: Learning to generate and transfer data with rectified flow. In *ICLR*, 2023.

Facundo Mémoli. Gromov–wasserstein distances and the metric approach to object matching. *Found. Comput. Math.*, 11(4):417–487, 2011.

Chenlin Meng, Yutong He, Yang Song, Jiaming Song, Jiajun Wu, Jun-Yan Zhu, and Stefano Ermon. SDEdit: Guided image synthesis and editing with stochastic differential equations. In *ICLR*, 2022.

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

Azadeh Sadat Mozafari and Mansour Jamzad. A svm-based model-transferring method for heterogeneous domain adaptation. *Pattern Recognit.*, 56:142–158, 2016.

Debarghya Mukherjee, Aritra Guha, Justin M Solomon, Yuekai Sun, and Mikhail Yurochkin. Outlier-robust optimal transport. In *ICML*, 2021.

Aamir Mustafa and Rafał K Mantiuk. Transformation consistency regularization–a semi-supervised paradigm for image-to-image translation. In *ECCV*, 2020.

Andriy Myronenko and Xubo Song. Point set registration: Coherent point drift. *IEEE Trans. Pattern Anal. Mach. Intell.*, 32(12):2262–2275, 2010.

Prashant Pandey, Mrigank Raman, Sumanth Varambally, and Prathosh Ap. Generalization on unseen domains via inference-time label-preserving target projections. In *ICCV*, 2021.

Matteo Pariset, Ya-Ping Hsieh, Charlotte Bunne, Andreas Krause, and Valentin De Bortoli. Unbalanced diffusion schrödinger bridge, 2024. URL `https://openreview.net/forum?id=CWoIj2XJuT`.

Gabriel Peyré, Marco Cuturi, et al. Computational optimal transport: With applications to data science. *Found. Trends Mach. Learn.*, 11(5-6):355–607, 2019.

Aram-Alexandre Pooladian, Heli Ben-Hamu, Carles Domingo-Enrich, Brandon Amos, Yaron Lipman, and Ricky T. Q. Chen. Multisample flow matching: Straightening flows with minibatch couplings. In *ICML*, 2023.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *ICML*, 2021.

Sebastian Reich. A nonparametric ensemble transform method for bayesian inference. *SIAM J. Sci. Comput.*, 35(4):A2013–A2024, 2013.

Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *Int. J. Comput. Vis.*, 115(3):211–252, 2015.

Kate Saenko, Brian Kulis, Mario Fritz, and Trevor Darrell. Adapting visual category models to new domains. In *ECCV*, 2010.

Kuniaki Saito, Donghyun Kim, Stan Sclaroff, Trevor Darrell, and Kate Saenko. Semi-supervised domain adaptation via minimax entropy. In *ICCV*, 2019.

Ryoma Sato, Marco Cuturi, Makoto Yamada, and Hisashi Kashima. Fast and robust comparison of probability measures in heterogeneous spaces. *arXiv preprint arXiv:2002.01615*, 2020.

Morgan A Schmitz, Matthieu Heitz, Nicolas Bonneel, Fred Ngole, David Coeurjolly, Marco Cuturi, Gabriel Peyré, and Jean-Luc Starck. Wasserstein dictionary learning: Optimal transport-based unsupervised nonlinear dictionary learning. *SIAM J. Imaging Sci.*, 11(1): 643–678, 2018.

Vivien Seguy, Bharath Bhushan Damodaran, Rémi Flamary, Nicolas Courty, Antoine Rolet, and Mathieu Blondel. Large-scale optimal transport and mapping estimation. In *ICLR*, 2018.

Thibault Séjourné, François-Xavier Vialard, and Gabriel Peyré. The unbalanced gromov wasserstein distance: Conic formulation and relaxation. In *NeurIPS*, 2021.

Thibault Séjourné, Gabriel Peyré, and François-Xavier Vialard. Unbalanced optimal transport, from theory to numerics. *Handb. Numer. Anal.*, 24:407–471, 2023.

Chen Shen and Yuhong Guo. Unsupervised heterogeneous domain adaptation with sparse feature transformation. In *ACML*, 2018.

Yuyang Shi, Valentin De Bortoli, Andrew Campbell, and Arnaud Doucet. Diffusion schrödinger bridge matching. In *NeurIPS*, 2023.

Akash Singh, Kirti Biharie, Marcel J T Reinders, Ahmed Mahfouz, and Tamim Abdelaal. scTopoGAN: unsupervised manifold alignment of single-cell data. *Bioinform. Adv.*, 3(1): vbad171, 2023.

Ritambhara Singh, Pinar Demetci, Giancarlo Bonora, Vijay Ramani, Choli Lee, He Fang, Zhijun Duan, Xinxian Deng, Jay Shendure, Christine Disteche, et al. Unsupervised manifold alignment for single-cell multi-omics data. In *ACM BCB*, pages 1–10, 2020.

Richard Sinkhorn and Paul Knopp. Concerning nonnegative matrices and doubly stochastic matrices. *Pacific J. Math.*, 21(2):343–348, 1967.

Justin Solomon, Fernando De Goes, Gabriel Peyré, Marco Cuturi, Adrian Butscher, Andy Nguyen, Tao Du, and Leonidas Guibas. Convolutional wasserstein distances: Efficient optimal transportation on geometric domains. *ACM Trans. Graph.*, 34(4):1–11, 2015.

Karl-Theodor Sturm. On the geometry of metric measure spaces. *Acta Math.*, 196(1):65–131, 2006.

Xuan Su, Jiaming Song, Chenlin Meng, and Stefano Ermon. Dual diffusion implicit bridges for image-to-image translation. In *ICLR*, 2023.

Vayer Titouan, Nicolas Courty, Romain Tavenard, and Rémi Flamary. Optimal transport for structured data with application on graphs. In *ICML*, 2019.

Alexander Tong, Guy Wolf, and Smita Krishnaswamy. Fixing bias in reconstruction-based anomaly detection with lipschitz discriminators. *J. Signal Process. Syst.*, 94(2):229–243, 2022.

Alexander Tong, Kilian FATRAS, Nikolay Malkin, Guillaume Huguet, Yanlei Zhang, Jarrid Rector-Brooks, Guy Wolf, and Yoshua Bengio. Improving and generalizing flow-based generative models with minibatch optimal transport. *Trans. Mach. Learn. Res.*, 2024a. ISSN 2835-8856.

Alexander Y. Tong, Nikolay Malkin, Kilian Fatras, Lazar Atanackovic, Yanlei Zhang, Guillaume Huguet, Guy Wolf, and Yoshua Bengio. Simulation-free Schrödinger bridges via score and flow matching. In *AISTATS*, 2024b.

Yao-Hung Hubert Tsai, Yi-Ren Yeh, and Yu-Chiang Frank Wang. Learning cross-domain landmarks for heterogeneous domain adaptation. In *CVPR*, 2016.

Théo Uscidda and Marco Cuturi. The monge gap: A regularizer to learn all transport maps. In *ICML*, 2023.

Cédric Villani. *Optimal transport: old and new*, volume 338. Springer, 2009.

Chang Wang and Sridhar Mahadevan. Heterogeneous domain adaptation using manifold alignment. In *IJCAI*, 2011.

Chao Wang, Haiyong Zheng, Zhibin Yu, Ziqiang Zheng, Zhaorui Gu, and Bing Zheng. Discriminative region proposal adversarial networks for high-quality image-to-image translation. In *ECCV*, 2018.

Qian Wang and Toby P Breckon. Cross-domain structure preserving projection for heterogeneous domain adaptation. *Pattern Recognit.*, 123:108362, 2022.

Hanrui Wu, Hong Zhu, Yuguang Yan, Jiaju Wu, Yifan Zhang, and Michael K Ng. Heterogeneous domain adaptation by information capturing and distribution matching. *IEEE Trans. Image Process.*, 30:6364–6376, 2021.

Hongteng Xu, Dixin Luo, Ricardo Henao, Svati Shah, and Lawrence Carin. Learning autoencoders with relational regularization. In *ICML*, 2020.

Yuguang Yan, Wen Li, Michael KP Ng, Mingkui Tan, Hanrui Wu, Huaqing Min, and Qingyao Wu. Learning discriminative correlation subspace for heterogeneous domain adaptation. In *IJCAI*, 2017.

Yuguang Yan, Wen Li, Hanrui Wu, Huaqing Min, Mingkui Tan, and Qingyao Wu. Semi-supervised optimal transport for heterogeneous domain adaptation. In *IJCAI*, 2018.

Yuan Yao, Yu Zhang, Xutao Li, and Yunming Ye. Heterogeneous domain adaptation via soft transfer network. In *ACM MM*, 2019.

Yuan Yao, Yu Zhang, Xutao Li, and Yunming Ye. Discriminative distribution alignment: A unified framework for heterogeneous domain adaptation. *Pattern Recognit.*, 101:107165, 2020.

Zili Yi, Hao Zhang, Ping Tan, and Minglun Gong. Dualgan: Unsupervised dual learning for image-to-image translation. In *CVPR*, 2017.

Mikhail Yurochkin, Sebastian Claici, Edward Chien, Farzaneh Mirzazadeh, and Justin M Solomon. Hierarchical optimal transport for document representation. In *NeurIPS*, 2019.

Daniel Zeiberg, Shantanu Jain, and Predrag Radivojac. Fast nonparametric estimation of class proportions in the positive-unlabeled classification setting. In *AAAI*, 2020.

Jiying Zhang, Xi Xiao, Long-Kai Huang, Yu Rong, and Yatao Bian. Fine-tuning graph neural networks via graph topology induced optimal transport. In *IJCAI*, 2022.

Pan Zhang, Bo Zhang, Dong Chen, Lu Yuan, and Fang Wen. Cross-domain correspondence learning for exemplar-based image translation. In *CVPR*, 2020.

Min Zhao, Fan Bao, Chongxuan Li, and Jun Zhu. EGSDE: Unpaired image-to-image translation via energy-guided stochastic differential equations. In *NeurIPS*, 2022.

Joey Tianyi Zhou, Sinno Jialin Pan, and Ivor W Tsang. A deep learning framework for hybrid heterogeneous transfer learning. *Artif. Intell.*, 275:310–328, 2019a.

Joey Tianyi Zhou, Ivor W. Tsang, Sinno Jialin Pan, and Mingkui Tan. Multi-class heterogeneous domain adaptation. *J. Mach. Learn. Res.*, 20(57):1–31, 2019b.

Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *ICCV*, 2017a.

Jun-Yan Zhu, Richard Zhang, Deepak Pathak, Trevor Darrell, Alexei A Efros, Oliver Wang, and Eli Shechtman. Toward multimodal image-to-image translation. In *NeurIPS*, 2017b.