Hilbert Curve Projection Distance for Distribution Comparison

Tao Li, Cheng Meng, Hongteng Xu, *Member, IEEE*, and Jun Yu

***Abstract*—Distribution comparison plays a central role in many machine learning tasks like data classification and generative modeling. In this study, we propose a novel metric, called *Hilbert curve projection (HCP) distance*, to measure the distance between two probability distributions with low complexity. In particular, we first project two high-dimensional probability distributions using Hilbert curve to obtain a coupling between them, and then calculate the transport distance between these two distributions in the original space, according to the coupling. We show that HCP distance is a proper metric and is well-defined for probability measures with bounded supports. Furthermore, we demonstrate that the modified empirical HCP distance with the L_p cost in the d-dimensional space converges to its population counterpart at a rate of no more than $O(n^{-1/2\max\{d,p\}})$. To suppress the curse-of-dimensionality, we also develop two variants of the HCP distance using (learnable) subspace projections. Experiments on both synthetic and real-world data show that our HCP distance works as an effective surrogate of the Wasserstein distance with low complexity and overcomes the drawbacks of the sliced Wasserstein distance.**

***Index Terms*—Distribution comparison, optimal transport, Hilbert curve, Wasserstein distance, projection robust Wasserstein distance.**

I. Introduction

MEASURING the distance between two probability distributions is significant for many machine learning tasks, e.g., data classification [1], [2], [3], generative modeling [4], [5], among others. Among the commonly-used distance measures for probability distributions, classic f-divergence based metrics, e.g., the Kullback-Leibler (KL) divergence and the total variation (TV) distance, do not work well when the probability distributions have disjoint supports [6], while the kernel-based methods like the maximum mean discrepancy (MMD) [7] require sophisticated kernel selection. Recently, the Wasserstein distance [8] has attracted wide attention in the machine learning community because of its advantages on overcoming these limitations, and it has shown great potential in many challenging learning problems [6], [9].

Given the samples of the two distributions, the computation of Wasserstein distance corresponds to solving either differential equations [10], [11] or linear programming problems [12], [13]. To alleviate the computational burden, the Sinkhorn distance [14] imposes an entropic regularizer on the Wasserstein distance and leverages the Sinkhorn-scaling algorithm accordingly. The work in [6] considers the Kantorovich duality of Wasserstein distance and converts the problem to a "max-min" game. Besides these two approximation methods, more surrogates of the Wasserstein distance have been proposed in recent years, e.g., the sliced Wasserstein (SW) distance [15], the generalized sliced Wasserstein (GSW) distance [16], the tree-structured Wasserstein (TSW) distance [17], and so on. Despite the computational efficiency, these surrogates may fail to provide effective approximations for the Wasserstein distance. Take the two Gaussian mixture distributions in Fig. 1(a) as an example. We keep the source distribution (in purple) unchanged while shifting the central Gaussian component of the target distribution (in orange) vertically with an offset $\alpha \in [0, 1]$. For the various distances defined between the two distributions, Fig. 1(b) shows their changes with respect to α. Existing methods often lead to coarse approximations of the Wasserstein distance, whose tendencies w.r.t. α can even be opposite to the Wasserstein distance. This phenomenon indicates that replacing the Wasserstein distance with these surrogates may lead to sub-optimal, even undesired, results in some learning tasks.

In this study, we propose a novel metric for distribution comparison, called Hilbert curve projection (HCP) distance. In principle, our HCP distance first projects two probability distributions along the Hilbert curve [18] of the sample space and then calculates the coupling based on the projected distributions. Such a Hilbert curve projection works better than linear projections on preserving the structure of the data distribution since the Hilbert curve enjoys the locality-preserving property, i.e., the locality between data points in the high-dimensional space being preserved in the projected one-dimensional space [19], [20]. Our HCP distance provides a new surrogate of the Wasserstein

Manuscript received 10 November 2022; revised 27 November 2023; accepted 29 January 2024. Date of publication 8 February 2024; date of current version 5 June 2024. This work was supported in part by the National Natural Science Foundation of China under Grant 62106271, Grant 12101606, and Grant 12001042, in part by the Renmin University of China research fund program for young scholars, and Beijing Institute of Technology research fund program for young scholars, and in part by Beijing Municipal Natural Science Foundation under Grant 1232019. Recommended for acceptance by B. Hammer. *(All authors contributed equally and the order of authors' names is alphabetical.) (Corresponding author: Hongteng Xu.)*

Tao Li is with the Institute of Statistics and Big Data, Renmin University of China, Beijing 100872, China (e-mail: 2019000153@ruc.edu.cn).

Cheng Meng is with the Center for Applied Statistics, Institute of Statistics and Big Data, Renmin University of China, Beijing 100872, China (e-mail: chengmeng@ruc.edu.cn).

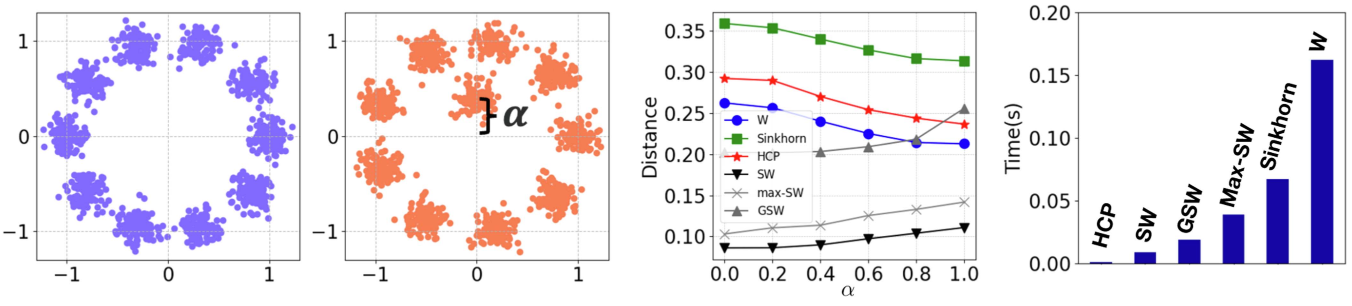
Hongteng Xu is with the Gaoling School of Artificial Intelligence, Beijing Key Laboratory of Big Data Management and Analysis Methods, Renmin University of China, Beijing 100872, China (e-mail: hongtengxu313@gmail.com).

Jun Yu is with the School of Mathematics and Statistics, Beijing Institute of Technology, Beijing 100811, China (e-mail: yujunbeta@bit.edu.cn).

The code of this work is at https://github.com/sherlockLitao/HCP.

This article has supplementary downloadable material available at https://doi.org/10.1109/TPAMI.2024.3363780, provided by the authors.

Digital Object Identifier 10.1109/TPAMI.2024.3363780



(a) (b) (c)

Fig. 1. (a) The samples of source and target distributions. (b) Illustrations of various distances with the increase of α. (c) Comparisons for various distances on their runtime. The proposed HCP distance provides an effective and efficient surrogate of the Wasserstein distance, which performs similarly and has low computational complexity.

distance, with both efficiency and effectiveness — it performs similarly as the Wasserstein distance does and spends less time than other methods, as shown in Fig. 1(b) and (c), respectively.

We provide in-depth analysis of the HCP distance, demonstrating that it is a well-defined metric for probability measures with bounded supports. Given n samples in d-dimensional space, the computational complexity for calculating empirical HCP distance is approximately linear to n. In addition, the modified empirical HCP distance with the L_p cost converges to its population counterpart at a rate of no more than $O(n^{-1/2\max\{d,p\}})$. Furthermore, to mitigate the curse-of-dimensionality, we develop two variants of the HCP distance using (learnable) subspace projections. We test the HCP distance and its variants on various machine learning tasks, including data classification and generative modeling, and compare them with state-of-the-art methods. Empirical results support the superior performance of the proposed metrics in both synthetic and real-data settings.

II. Related Work and Preliminaries

A. Wasserstein Distance and Sliced Wasserstein Distance

Let $\mathscr{P}_p(\mathbb{R}^d)$ be the set of Borel probability measures in $\mathbb{R}^d$ with finite pth moment. Consider two probability measures $\mu, \nu \in \mathscr{P}_p(\mathbb{R}^d)$ with corresponding probability density functions f_μ, f_ν. The p-Wasserstein distance [8] between μ and ν is defined as

$$\begin{aligned}\mathrm{W}_p(\mu,\nu) &= \Big(\inf_{\gamma\in\Gamma(\mu,\nu)} \int_{\mathbb{R}^d\times\mathbb{R}^d} \|x-y\|_p^p \mathrm{d}\gamma(x,y)\Big)^{1/p}\\ &\xrightarrow{1\mathrm{D}\,\mu,\nu} \Big(\int_0^1 \|F_\mu^{-1}(z)-F_\nu^{-1}(z)\|_p^p \mathrm{d}z\Big)^{1/p}, \qquad (1)\end{aligned}$$

where $\|\cdot\|_p$ is the L_p norm and $\Gamma(\mu,\nu)$ is the set of all couplings (or called transportation plans): $\Gamma(\mu,\nu)=\{\gamma\in\mathscr{P}_p(\mathbb{R}^d\times\mathbb{R}^d)\ \text{ s.t. }\forall \text{ Borel set } A,B\subset\mathbb{R}^d, \gamma(A\times\mathbb{R}^d)=\mu(A), \gamma(\mathbb{R}^d\times B)=\nu(B)\}$.

Though it is difficult to calculate Wasserstein distance in general, according to (1), for one-dimensional probability measures μ and ν, the Wasserstein distance has a closed-form, where $F_\mu(x)=\mu((-\infty,x])=\int_{-\infty}^x f_\mu(x)\mathrm{d}x$ is the cumulative distribution function (CDF) for f_μ, and similarly, F_ν is the CDF for f_ν. This fact motivates the design of the sliced Wasserstein (SW) distance [15] (and its variants [21], [22]), which projects d-dimensional probability measures to 1D space and computes the 1D Wasserstein distance accordingly. Let $\mathbb{S}_{d,q}=\{\mathbf{E}\in\mathbb{R}^{d\times q}: \mathbf{E}^\top\mathbf{E}=\mathbf{I}_q\}$ $(q<d)$ be the set of orthogonal matrices and $P_{\mathbf{E}}(\boldsymbol{x})=\mathbf{E}^\top\boldsymbol{x}$ be the linear transformation for $\boldsymbol{x}\in\mathbb{R}^d$. Denote $P_{\mathbf{E}\#}\mu$ as the pushforward of μ by $P_{\mathbf{E}}$, which corresponds to the distribution of the projected samples. For all $\mu,\nu\in\mathscr{P}_p(\mathbb{R}^d)$, the p-sliced Wasserstein distance between them is given by

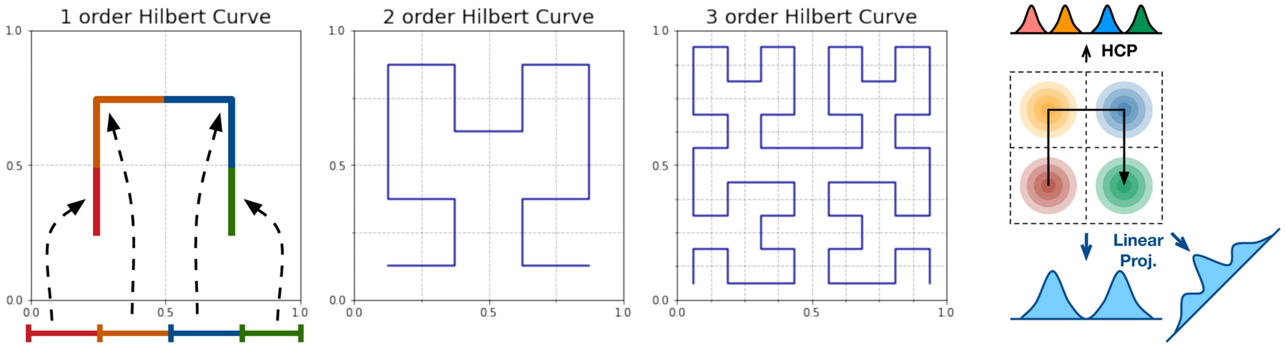
$$\mathrm{SW}_p(\mu,\nu)=\Big(\int_{\mathbf{E}\in\mathbb{S}_{d,1}} \mathrm{W}_p^p\left(P_{\mathbf{E}\#}\mu, P_{\mathbf{E}\#}\nu\right)\mathrm{d}\sigma(\mathbf{E})\Big)^{1/p}, \quad (2)$$

where σ is the uniform distribution on $\mathbb{S}_{d,1}$. However, as aforementioned, the SW distance often fails to approximate the Wasserstein distance because its linear projections break the structure of the original distributions. Additionally, the random projections introduce unnecessary randomness when computing the distance.

B. Other Optimal Transport Distances

Based on the Wasserstein distance and the SW distance mentioned above, many optimal transport-based distances have been proposed in recent years, which can be roughly categorized into two classes. The first class considers approximating the Wasserstein distance by alternative optimization methods. Typically, the Sinkhorn distance in [14] imposes an entropic regularizer on the Wasserstein distance. Following this framework, many variants have been proposed to accelerate the computation [23], [24], [25], [26], [27], [28]. Besides the Sinkhorn-based algorithm, other methods, such as primal-dual method [29], [30], stochastic gradient descent [31], proximal point method [32], Bregman alternating direction method of multipliers (Bregman ADMM) [33], [34], [35], and so on, have drawn great attention. However, the computational cost of these methods is at least $O(n^2)$, which may not be applicable to large-scale data.

The second class follows the strategy of the SW distance, finding surrogates of the Wasserstein distance by various projection methods. To improve the efficiency of the SW distance, Max-sliced Wasserstein (Max-SW) [21], distributional sliced Wasserstein [36] and orthogonal sliced Wasserstein [37]



(a) $\{\widehat{H}_k\}_{k=1}^3$

(b) HCP v.s. Linear proj.

Fig. 2. (a) The k-order Hilbert curve, with $k = 1, 2, 3$, in 2D space. (b) The comparison between Hilbert curve projection (HCP) and linear projections.

have been proposed. Recently, the projection robust and integral projection robust Wasserstein distance consider projecting on the subspaces of higher dimensions [38], [39], [40]. Beyond using linear projections, generalized sliced Wasserstein (GSW) [16], convolutional sliced Wasserstein [41], and amortized sliced Wasserstein [42] have been proposed, which capture the complicated structure of data distributions by nonlinear projections. In addition to above methods, the tree sliced Wasserstein (TSW) [17] generates random tree metrics for data points and then computes Wasserstein distance on tree metrics. TSW computes distance on given tree metrics and thus, it is more suitable for classification task compared with generative model. Note that, these projection-based distances may fail to provide effective surrogates for the Wasserstein distance, as shown in Fig. 1, and searching for effective projections will bring additional computational cost.

C. Applications of Optimal Transport Distances

Optimal transport distances have recently drawn great attention in various machine-learning tasks. Wasserstein distance and its variants serve as the loss functions for generative modeling, such as Wasserstein generative adversarial networks (WGANs) [6], [21], [43], [44] and Wasserstein autoencoders (WAEs) [9], [22], [45]. In classification tasks, optimal transport distances measure the discrepancy between set-level data [46], leading to discriminative models for various data, such as texts [1], [2], point clouds [3], and graphs [47]. Besides generative modeling and classification, optimal transport distances are also applied to other problems, such as data clustering [34], [48], dimension reduction [49], [50], and domain adaptation [51]. These optimal transport-based methods have shown the potential for various practical applications, e.g., graph matching and partitioning [52], [53], color transfer [54], [55], document analysis [56], and so on.

III. Proposed Method

A. Hilbert Curve and its Locality-Preserving Property

Our work is based on the well-known Hilbert curve [18]. Mathematically, for a d-dimensional ($d \geq 2$) unit hyper-cube, i.e., $[0,1]^d$, the k-order Hilbert space-filling curve, denoted as $\widehat{H}_k$, partitions [0,1] and $[0,1]^d$ into $(2^k)^d$ intervals and blocks, respectively, and constructs a bijection between them. Taking the $\{\widehat{H}_k\}_{k=1}^3$ in 2D space as examples, Fig. 2(a) illustrates how the intervals in [0,1] are constructed and mapped to the blocks in $[0,1]^2$. The Hilbert curve is defined as the limit of a sequence of k-order Hilbert space-filling curves, i.e., $H(x) = \lim_{k\rightarrow\infty} \widehat{H}_k(x)$ with $x \in [0,1]$. It provides a well-defined surjection $H : [0,1] \rightarrow [0,1]^d$ and is able to cover the entire hypercube [18]. Note that, although the Hilbert curve H is not a bijection, most of the data points in $[0,1]^d$ are still invertible — it is known that the set $\mathcal{A}$, which includes the points in $[0,1]^d$ such that these points have more than one pre-image in $[0,1]$, has measure zero [57]. Actually, for any point in $\mathcal{A}$, there are finite pre-images in [0,1]. Hence, there is a bijection between $[0,1]^d$ and $\{\min\{H^{-1}(x)\} : x \in [0,1]^d\}$. We denote $\mathcal{N} = [0,1] \backslash \{\min\{H^{-1}(x)\} : x \in [0,1]^d\}$ as the negligible set.

We are interested in the Hilbert curve because it enjoys the so-called *locality-preserving property* [57], [58]: For any $x, y \in [0,1]$, one has

$$\|H(x) - H(y)\|_2 \leq 2\sqrt{d+3}|x-y|^{1/d}.$$

Such an inequality indicates the advantage of the Hilbert curve over linear projections. In particular, if two points are far from each other in a high-dimensional space, their pre-images with respect to the Hilbert curve will also be far from each other. Fig. 2(b) further illustrates this property through a toy example. Specifically, for a 2D distribution with four modals, while linear projections tend to wrongly merge some modals, the projection along the Hilbert curve can distinguish the modals successfully. This property motivates us to propose the Hilbert curve projection distance shown below.

B. Hilbert Curve Projection Distance

Hilbert Curve for Probability Measure: In this study, we focus on probability measures with bounded supports. This condition has been widely used in the optimal transport literature to simplify the theoretical analysis [59], [60], [61], [62]. Let $\mathscr{P}_\infty(\mathbb{R}^d)$ be the set of Borel probability measures in $\mathbb{R}^d$ with bounded supports. Denote the support of a probability measure

$\mu \in \mathscr{P}_\infty(\mathbb{R}^d)$ as Ω_μ. Let $\widetilde{\Omega}_\mu = \prod_{i=1}^d [a_i, b_i]$ be the smallest hyper-rectangle covering Ω_μ. For each μ, we can define a Hilbert curve $H_\mu : [0,1] \to \widetilde{\Omega}_\mu$ as $H_\mu(t) = (b-a) \odot H(t) + a$ where $\odot$ is the Hadamard product and a, b are vectors with ith dimension being a_i, b_i respectively.[1]

Denote $\mathcal{K} = \{\frac{m_1}{2^{m_2}} : m_1, m_2 \in \mathbb{N}, m_1 \leq 2^{m_2}\}$ as a dense set in [0,1]. According to [57], [58], $H_\mu([0,t])$ is a Borel measurable set for any $t \in \mathcal{K}$ and $H_\mu([0,t])$ is a Lebesgue measurable set for any $t \in [0,1]$. This motivates us to define a cumulative distribution function along the Hilbert curve (denoted as $g_\mu : [0,1] \to [0,1]$) and the corresponding inverse cumulative distribution function (g_μ^{-1}), respectively

$$\begin{aligned} g_\mu(t) &= \inf\nolimits_{s \in \mathcal{K},\, s \geq t} \mu\Big(H_\mu([0,s])\Big), \\ g_\mu^{-1}(t) &= \inf\nolimits_{s \in [0,1],\, g_\mu(s) > t} s. \end{aligned} \tag{3}$$

Accordingly, the formal definition of our Hilbert curve projection distance is as follows.

Definition 1 (Hilbert Curve Projection Distance): Let $\mathscr{P}_\infty(\mathbb{R}^d)$ be the set of Borel probability measures in $\mathbb{R}^d$ with bounded supports. Denote the supports of two probability measures $\mu, \nu \in \mathscr{P}_\infty(\mathbb{R}^d)$ as Ω_μ and Ω_ν, respectively. Denote $\mathcal{K} = \{\frac{m_1}{2^{m_2}} : m_1, m_2 \in \mathbb{N}, m_1 \leq 2^{m_2}\}$ as a dense set in [0,1]. Let $H_\mu : [0,1] \to \widetilde{\Omega}_\mu$, where $\widetilde{\Omega}_\mu$ is the smallest hyper-rectangle that covers Ω_μ, $g_\mu(t) = \inf_{s \in \mathcal{K},\, s \geq t} \mu\Big(H_\mu([0,s])\Big)$, and $g_\mu^{-1}(t) = \inf_{s \in [0,1],\, g_\mu(s) > t} s$ (with H_ν, g_ν and g_ν^{-1} defined in the same way). For $p \in \mathbb{Z}_+$, the p-order Hilbert curve projection distance is defined as

$$\mathrm{HCP}_p(\mu, \nu) = \left(\int_0^1 \| H_\mu(g_\mu^{-1}(t)) - H_\nu(g_\nu^{-1}(t)) \|_p^p \mathrm{d}t \right)^{\frac{1}{p}}. \tag{4}$$

Remark 1: The assumption for bounded support is commonly used in optimal transport literature [59], [60], [61], [62] and is essential for technical proof. For unbounded cases, one possible remedy is to use a bounded measurable bijective mapping f, such as element-wise $\tan^{-1}(\cdot)$, to transform the original measures μ and ν. We then could get the transport plan between $f_{\#}\mu$ and $f_{\#}\nu$ based on the Hilbert curve projections where $f_{\#}\mu$ is the pushforward of μ by f, and compute the distance in the original unbounded space.

According to the definition, the principle of our HCP distance is projecting high-dimensional distributions along their Hilbert curves to obtain an efficient and effective coupling between them, and then calculating the corresponding HCP distance between two distributions in the original space according to the coupling. The following theoretical results show that our HCP distance is a proper metric, and it is an upper bound of the p-Wasserstein distance.

Theorem 1: HCP_p is a well-defined metric in $\mathscr{P}_\infty(\mathbb{R}^d)$, and $\mathrm{W}_p(\mu, \nu) \leq \mathrm{HCP}_p(\mu, \nu), \forall \mu, \nu \in \mathscr{P}_\infty(\mathbb{R}^d)$.

Given two random variables, i.e., $Z_1 \sim \mu$ and $Z_2 \sim \nu$, we denote $\mathrm{HCP}(\mu, \nu)$ as $\mathrm{HCP}(Z_1, Z_2)$. Clearly, HCP distance has the following properties which are also valid for Wasserstein distance [63].

1) For any $z \in \mathbb{R}^d$, $\mathrm{HCP}_p(Z_1 + z, Z_1) = \|z\|_p$.
2) For any $a \in \mathbb{R}$, $\mathrm{HCP}_p(aZ_1, aZ_2) = |a| \mathrm{HCP}_p(Z_1, Z_2)$.
3) For any $z \in \mathbb{R}^d$, $\mathrm{HCP}_p(Z_1 + z, Z_2 + z) = \mathrm{HCP}_p(Z_1, Z_2)$.
4) For any $z \in \mathbb{R}^d$, $\mathrm{HCP}_2^2(Z_1 + z, Z_2) = \mathrm{HCP}_2^2(Z_1, Z_2) + \|z + \mathbb{E}Z_1 - \mathbb{E}Z_2\|_2^2 - \|\mathbb{E}Z_1 - \mathbb{E}Z_2\|_2^2$.

Here, "$Z_1 + z$" means impose a translation z on the random variable Z_1, and "aZ_1" means scaling the random variable Z_1.

C. Topological Properties of the HCP Distance

As shown in Theorem 1, HCP distance induces a stronger topology compared to Wasserstein distance because $\mathrm{W}_p(\mu, \nu) \leq \mathrm{HCP}_p(\mu, \nu)$. This means that the sequence of probability measures, i.e., $\{\mu_n\}$, always converges in Wasserstein distance when $n \to \infty$ if it converges in HCP distance, i.e., $\mathrm{HCP}(\mu_n, \mu) \to 0 \Rightarrow \mathrm{W}(\mu_n, \mu) \to 0$.

Additionally, we compare our HCP distance with the total variation (TV) distance on their induced topology and propose the following Theorem:

Theorem 2: Let $\widetilde{\Omega}_\mu$ be the smallest hyper-rectangle that covers the support of the probability measure $\mu \in \mathscr{P}_\infty(\mathbb{R}^d)$. When $\{\mu_n\}$ converges to μ in the total variation distance and $\widetilde{\Omega}_{\mu_n} = \widetilde{\Omega}_\mu$ for all n's, we have $\mathrm{TV}(\mu_n, \mu) \to 0 \Rightarrow \mathrm{HCP}(\mu_n, \mu) \to 0$.

Note that, our HCP distance is not equivalent to the Wasserstein distance or the TV distance because

$$\begin{aligned} &\mathrm{W}(\mu_n, \mu) \to 0 \nRightarrow \mathrm{HCP}(\mu_n, \mu) \to 0, \\ &\mathrm{HCP}(\mu_n, \mu) \to 0 \nRightarrow \mathrm{TV}(\mu_n, \mu) \to 0. \end{aligned}$$

The following two examples verify the above claims, respectively.

Example 1: Consider two probability distribution $\mu_\theta = \frac{1}{4}(\delta_{(0,0)} + \delta_{(1,1)} + \delta_{(\frac{1}{2}-\theta, \frac{1}{4})} + \delta_{(\frac{1}{2}-\theta, \frac{3}{4})})$ and $\nu_\theta = \frac{1}{4}(\delta_{(0,0)} + \delta_{(1,1)} + \delta_{(\frac{1}{2}+\theta, \frac{1}{4})} + \delta_{(\frac{1}{2}+\theta, \frac{3}{4})})$ where δ is the Dirac measure. Then, when $0 < \theta < 0.5$, we have $\mathrm{W}_2(\mu_\theta, \nu_\theta) = |\sqrt{2}\theta|$. However, when $\theta \neq 0$, $\mathrm{HCP}_2(\mu_\theta, \nu_\theta) = \sqrt{2\theta^2 + 1/8}$.

Example 2: Let $Z \sim \mathrm{Unif}[0,1]$ be samples of the uniform distribution on the unit interval. Let μ_0 be the probability distribution of $(0, Z) \in \mathbb{R}^2$. Let μ_θ be the family of probability distributions parametrized with θ corresponding to $(\theta, Z) \in \mathbb{R}^2$. Then $\mathrm{HCP}_p(\mu_0, \mu_\theta) = \mathrm{W}_p(\mu_0, \mu_\theta) = |\theta|$. However, when $\theta \neq 0$, $\mathrm{TV}(\mu_0, \mu_\theta) = 1$.

In summary, we can find that HCP metricizes a topology stronger than the weak topology induced by the Wasserstein distance. Additionally, as shown in Example 2 (which is also used in [6]), our HCP distance can perform as well as the Wasserstein distance does when comparing the probability measures with disjoint supports.

D. Numerical Implementation

Let Δ^n be the n-Simplex. Given the samples of two probability measures, i.e., $X = \{x_i\}_{i=1}^n \sim \mu$ and $Y = \{y_j\}_{j=1}^m \sim \nu$,

[1] In the case when $a_i = b_i$, one can utilize the following two strategies without affecting the theoretical properties. The first strategy is to let $b_i = a_i + 1$. This may cause redundant computational costs in this dimension. The second strategy is removing this dimension, performing Hilbert curve in the $\mathbb{R}^{d-1}$ and complementing this dimension for the final Hilbert curve.

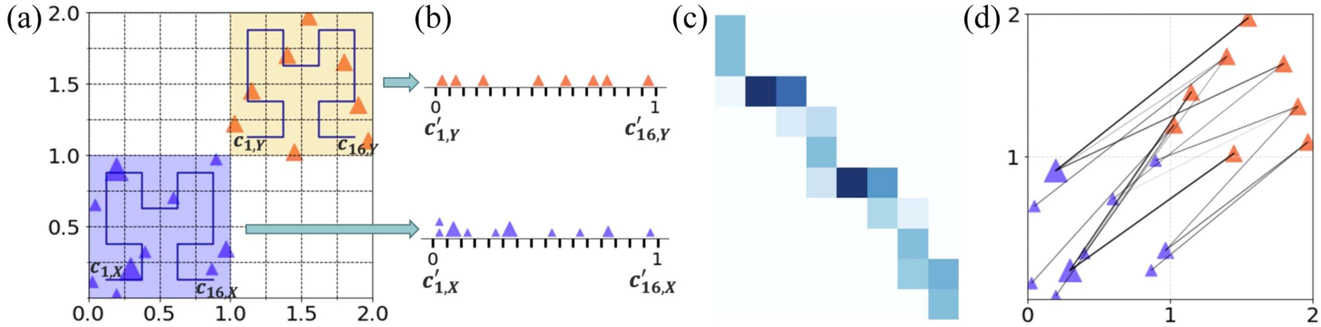


Fig. 3. An illustration of Algorithm 1 when $d = k = 2$. (a) The source (purple) and target (orange) data points, with corresponding hyper-rectangles and k-order Hilbert curves. (b) The projected points along the Hilbert curves. (c) The coupling matrix calculated by the projected points. (d) The HCP distance calculates the distance between the original samples based on the coupling matrix.

Algorithm 1: Computation of HCP Distance.

```
1: Input: $(\{x_i\}_{i=1}^n, \boldsymbol{a})$, $(\{y_j\}_{j=1}^m, \boldsymbol{b})$, $k$
2: Map $\{x_i\}_{i=1}^n$ to $\{x'_i\}_{i=1}^n$, $\{y_j\}_{j=1}^m$ to $\{y'_j\}_{j=1}^m$, through
   $(\widehat{H}_k^X)^{-1}$ and $(\widehat{H}_k^Y)^{-1}$. $O((n+m)dk)$
3: Calculate the optimal transport plan $\mathbf{P}$ between
   $(\{x'_i\}_{i=1}^n, \boldsymbol{a})$ and $(\{y'_j\}_{j=1}^m, \boldsymbol{b})$ using sorting and the
   North-West corner rule. Let $\mathcal{S} := \{(i,j) | P_{ij} \neq 0\}$.
   $O(n\log(n) + m\log(m))$
4: Output: $\mathbf{P}$, $\mathrm{HCP}_p = (\sum_{(i,j)\in\mathcal{S}} \|x_i - y_j\|_p^p P_{ij})^{1/p}$
```

whose empirical distributions are $\boldsymbol{a} \in \Delta^{n-1}$ and $\boldsymbol{b} \in \Delta^{m-1}$, respectively, we use a k-order Hilbert curve to calculate the empirical HCP distance between the two sample sets. Let $\widetilde{\Omega}_X$ and $\widetilde{\Omega}_Y$ be the smallest hyper-rectangles that cover these two sample sets, respectively. We define two k-order Hilbert curves, i.e., $\widehat{H}_k^X : [0,1] \rightarrow \widetilde{\Omega}_X$ and $\widehat{H}_k^Y : [0,1] \rightarrow \widetilde{\Omega}_Y$. Here, $\widehat{H}_k^X$ partitions both [0,1] and $\widetilde{\Omega}_X$ into 2^{kd} blocks, denoted by $\{c'_{j,X}\}_{j=1}^{2^{dk}}$ and $\{c_{j,X}\}_{j=1}^{2^{dk}}$, respectively, and construct a bijection between these blocks. For any data point $x \in \widetilde{\Omega}_X$, we assign x to its corresponding block $c_{j,X}$ in $\widetilde{\Omega}_X$, $j \in \{1, \ldots, 2^{kd}\}$, then map x to the center of the block $c'_{j,X} = (\widehat{H}_k^X)^{-1}(c_{j,X})$. Therefore, all the samples belonging to the same block are mapped to the same point in [0,1]. Based on $\widehat{H}_k^Y$, we map $\{y_j\}_{j=1}^m$ to [0,1] in the same way. The mapped points along with their probability densities are then used to calculate the optimal coupling matrix $\mathbf{P} \in \mathbb{R}^{n\times m}$ using the closed-form formulation of the 1D optimal transport problem. In particular, we first sort the mapped points, then calculate $\mathbf{P}$ using the North-West corner rule with $O(n+m)$ operations [64]. Note that there are at most $m+n$ nonzero elements in $\mathbf{P}$. Let $\mathcal{S} := \{(i,j) | P_{ij} \neq 0\}$ be the index set. Finally, the empirical HCP distance can then be calculated by $(\sum_{(i,j)\in\mathcal{S}} \|x_i - y_j\|_p^p P_{ij})^{1/p}$.

The above pipeline is illustrated in Fig. 3 and summarized in Algorithm 1, respectively. As suggested by [18], we select k that of the order $O(\log(n))$ in practice. Empirical results in the following experimental section show that the performance of Algorithm 1 is not sensitive to k.

Essentially, the empirical HCP distance is to compute the distance between two Hilbert rank-based sorted samples.[2] Note that, there are two main routines for Hilbert sort. The first gets Hilbert indices by projecting points in high dimension to the Hilbert curve and then sorts these indices based on the Hilbert rank [18], [58], [65], [66], [67]. The second idea is recursively sorting points without using Hilbert indices, e.g., the work in [18], [68] and the C++ library CGAL [69]. Though we take the first routine here, codes based on these two algorithms are both provided.

Computational Cost: The complexity of computing the k-order Hilbert index for n points in d-dimensional space is $O(ndk)$, [68], [70]. As shown in Algorithm 1, solving the optimal transport problem in Step 3 requires $O(n\log n + m\log m)$ time. When $m = O(n)$ and $k = O(\log(n))$, the overall computational complexity of HCP distance is at the order of $O(n\log(n)d)$.

Comparison With Existing Methods: The proposed HCP distance enjoys several critical advantages over the Wasserstein and SW distance.

- First, HCP can provide a decent transport plan between the input probability measures as a byproduct while SW could not. The key reason is that Hilbert curve is invertible almost everywhere. Linear projections in SW and nonlinear projections in GSW do not satisfy this property. Such a coupling matrix is essential for effective generative modeling, as will be seen in Section V.
- Second, we compute the distance in the original space rather than in the projected one-dimensional space. Hilbert curve only plays a role in achieving a transport plan. We don't apply any transformation on data points when computing HCP distance. However, SW involves transforming data points by linear projections and then computing Wasserstein distance using these transformed data points. Fig. 1 provides an intuitive example to show the difference between these two strategies. The reason why SW and its variants lead to an opposite trend compared with the Wasserstein distance is that SW computes Wasserstein

[2] The Hilbert rank is defined as follows: We say x_1 ranks in front of x_2, that is to say, $\min\{H^{-1}(x_1)\} < \min\{H^{-1}(x_2)\}$.

distance using linear transformed data points, and such linear transformation may break the structure of the original distributions. We refer to the Experiment Section for a more intuitive discussion.

- Last but not least, HCP computes faster than SW distance in practice. This is because calculating SW distance requires projection and sorting multiple times, while calculating HCP distance requires only once. Additionally, beyond the Hilbert curve-based discrepancy in [71], our HCP distance can deal with the samples with different sizes and weights with theoretical guarantees.

In summary, compared to Wasserstein distance, HCP has an approximately linear computational complexity, and thus is applicable to large-scale datasets. Compared to SW distances, HCP distance performs more similarly to the Wasserstein distance.

E. Statistical Convergence of Empirical HCP Distance

Let $\{x_i\}_{i=1}^n \sim \mu$, whose empirical measure is defined by $\mu_n = \frac{1}{n}\sum_{i=1}^n \delta_{x_i}$. Directly studying the statistical convergence of $\text{HCP}_p(\mu, \mu_n)$ is challenging because of the randomness of the bounded supports — the smallest hyper-rectangle covering the support of the probability measure μ_n, i.e., $\widetilde{\Omega}_{\mu_n}$ can be various w.r.t. sample size n, which leads to different Hilbert curves, and accordingly, we could not easily analyze the convergence rate without any other strict conditions on the support's boundary.

To eliminate the influence of the randomness, we consider an indirect strategy, studying a modified empirical Hilbert curve projection distance instead. Specifically, following the definitions in (3), we first define the cumulative distribution function and its inverse for the empirical measure μ_n, whose Hilbert curve, however, is based on the original probability measure μ

$$\begin{aligned}\hat{g}_{\mu_n}(t) &= \inf\nolimits_{s\in\mathcal{K},\, s\geq t}\, \mu_n\Big(H_\mu([0,s])\Big),\\ \hat{g}^{-1}_{\mu_n}(t) &= \inf\nolimits_{s\in[0,1],\, \hat{g}_{\mu_n}(s)>t}\, s. \end{aligned} \tag{5}$$

Accordingly, we define the modified empirical Hilbert curve projection distance as

$$\overline{\text{HCP}}_p(\mu,\mu_n) = \left(\int_0^1 \|H_\mu(g_\mu^{-1}(t)) - H_\mu(\hat{g}^{-1}_{\mu_n}(t))\|_p^p \mathrm{d}t\right)^{\frac{1}{p}}. \tag{6}$$

The only difference between $\text{HCP}_p(\mu,\mu_n)$ and $\overline{\text{HCP}}_p(\mu,\mu_n)$ is that the latter replaces the H_{μ_n} defined on $\widetilde{\Omega}_{\mu_n}$ with the H_μ defined on $\widetilde{\Omega}_\mu$. Note that, such a modified HCP distance is hard to implement in practice because both $\widetilde{\Omega}_\mu$ and H_μ are unknown in general. However, compared to the original HCP distance, the modified HCP distance is much easier to analyze because the Hilbert curve H_μ it used is deterministic and irrelevant to the sample. We demonstrate that the modified empirical HCP distance converges to its population counterpart almost surely. The following theorem provides an upper bound for the convergence rate.

Theorem 3: Let $\{x_i\}_{i=1}^n$ be an i.i.d. sample that is generated from the probability measure $\mu \in \mathscr{P}_\infty(\mathbb{R}^d)$. The empirical measure is defined by $\mu_n = \frac{1}{n}\sum_{i=1}^n \delta_{x_i}$. Then, we have almost surely

$$\overline{\text{HCP}}_p(\mu,\mu_n) \to 0, \text{ and } \mathbb{E}\overline{\text{HCP}}_p(\mu,\mu_n) \lesssim O(n^{-\frac{1}{2\max\{p,d\}}}).$$

Directly from Theorem 3, we can conclude the following theoretical results.

Corollary 3.1: Assume that probability measures $\mu, \nu \in \mathscr{P}_\infty(\mathbb{R}^d)$. Let $\{x_i\}_{i=1}^n$ and $\{y_i\}_{i=1}^n$ be two i.i.d. samples, which are generated from probability measures μ and ν, respectively. Let $\{x_{(i)^*}\}_{i=1}^n$ and $\{y_{(i)^*}\}_{i=1}^n$ be the sorted samples along the Hilbert curves H_μ and H_ν, respectively. Then, we have almost surely

$$\overline{\text{HCP}}_p(\mu_n,\nu_n) = \left(\frac{1}{n}\sum_{i=1}^n \|x_{(i)^*} - y_{(i)^*}\|_p^p\right)^{\frac{1}{p}} \to \text{HCP}_p(\mu,\nu),$$

where μ_n and ν_n are the empirical version of μ and ν, respectively. Furthermore, we have

$$|\mathbb{E}\overline{\text{HCP}}_p(\mu_n,\nu_n) - \text{HCP}_p(\mu,\nu)| \lesssim O(n^{-\frac{1}{2\max\{p,d\}}}).$$

Corollary 3.1 tells us the modified empirical Hilbert curve distance is to compute the distance between two Hilbert rank-based sorted samples. Moreover, when the samples are with different numbers, we have

Corollary 3.2: Assume that probability measures $\mu, \nu \in \mathscr{P}_\infty(\mathbb{R}^d)$. Let $\{x_i\}_{i=1}^n$ and $\{y_j\}_{j=1}^m$ be two i.i.d. samples, which are generated from probability measures μ and ν, respectively. Let $\{x_{(i)^*}\}_{i=1}^n$ and $\{y_{(j)^*}\}_{j=1}^m$ be the sorted samples along the Hilbert curves H_μ and H_ν, respectively. Then, we have

$$\overline{\text{HCP}}_p(\mu_n,\nu_m) = \left(\pi_{ij}\sum_{i=1}^n\sum_{j=1}^m \|x_{(i)^*} - y_{(j)^*}\|_p^p\right)^{\frac{1}{p}},$$

where μ_n, ν_m are the empirical version of μ, ν, respectively and, π_{ij} is the optimal transport plan between $\sum_{i=1}^n \delta_i/n$ and $\sum_{j=1}^m \delta_j/m$ with Euclidean distance cost. Furthermore, we have

$$|\mathbb{E}\overline{\text{HCP}}_p(\mu_n,\nu_m) - \text{HCP}_p(\mu,\nu)| \lesssim O(\min\{n,m\}^{-\frac{1}{2\max\{p,d\}}}).$$

Additionally, from Theorem 3, we know that convergence rate of modified empirical HCP distance has an upper bound $O(n^{-1/2p} + n^{-1/2d})$, which is slightly slower than the convergence rate of Wasserstein distance (i.e., $O(n^{-1/2p} + n^{-1/d})$ provided by [63]). In particular, given a probability measure μ and its empirical version μ_n, we have

$$W_p(\mu,\mu_n) \leq \overline{\text{HCP}}_p(\mu,\mu_n).$$

Furthermore, the following corollary indicates that under some mild conditions, the modified HCP distance can have the same convergence rate as Wasserstein distance does.

Corollary 3.3: Assume that probability measure $\mu \in \mathcal{P}_\infty(\mathbb{R}^d)$. If there exist two Borel measurable sets $A, B \subset \mathbb{R}^d$ such that $\mu(A) > 0, \mu(B) > 0, \mu(A\cup B) = 1$ and $dist(A,B) = \inf_{x\in A, y\in B}\|x-y\|_2 > 0$, then when $p \geq d$, we have

$$\mathbb{E}\overline{\text{HCP}}_p(\mu,\mu_n) = O(n^{-\frac{1}{2p}}), \text{ and } \mathbb{E}W_p(\mu_n,\mu) = O(n^{-\frac{1}{2p}}),$$

where μ_n is the empirical version of μ.

The above theoretical results of the modified HCP distance provide us with important insights into the convergence of our HCP distance — with the increase of the sample size n, the difference between $\widetilde{\Omega}_{\mu_n}$ and $\widetilde{\Omega}_{\mu}$ may not be too large in probability. Accordingly, the convergence of our HCP distance should be similar to that of the modified HCP distance in probability as well. Under some special cases, we can easily analyze the convergence of our HCP distance. For example, for nondegenerate discrete measures with finite supports, $\mathbb{E}\mathrm{HCP}_p(\mu, \mu_n)$ is of the order $O(n^{-1/2p})$, independently of the dimension, which is the same as the Wasserstein distance [63].

Corollary 3.4: Assume that probability measure μ is a non-degenerate discrete probability measure with K supports $\{s_i\}_{i=1}^K$, that is, $\mu = \sum_{i=1}^K p_i \delta_{s_i}$ and $\boldsymbol{p} = \{p_i\}_{i=1}^K \in \Delta^{K-1}$. Then, we have

$$\mathbb{E}\mathrm{HCP}_p(\mu, \mu_n) = O(n^{-\frac{1}{2p}}),$$

where μ_n is the empirical version of μ.

F. Other Space-Filling Curves

The proposed distance can be implemented based on other space-filling curves as well. For example, the Peano and Sierpinski space-filling curves also satisfy the Hölder inequality with exponent $1/d$. The Z-order space-filling curve, which is differentiable almost everywhere, also satisfies the Hölder inequality but with exponent $1/(d \log_2 3)$ [57], [58]. However, compared to the Hilbert curve, the Peano curve and Sierpinski curve are difficult to implement through algorithms. The convergence rate of the distance based on the Z-order curve is $O(n^{-\frac{1}{2\max\{d \log_2 3, p\}}})$, which is slower than that based on the Hilbert curve. In sum, we mainly focus on the Hilbert curve in this study.

IV. Variants of the Hilbert Curve Projection Distance

The theoretical results in the previous section indicate that analogous to the Wasserstein distance, our HCP distance may suffer from the curse-of-dimensionality as well. Motivated by the projection-robust Wasserstein distance [39], [40], we propose two variants of the HCP distance to alleviate this limitation.

A. Integral Projection Robust Hilbert Curve Projection Distance

We first propose the integral projection robust Hilbert curve projection (IPRHCP) distance that combines the idea of HCP distance and random projections.

Definition 2: Suppose that probability measures $\mu, \nu \in \mathscr{P}_\infty(\mathbb{R}^d)$. The p-order q-dimensional integral projection robust Hilbert curve projection distance is defined as

$$\begin{aligned}&\mathrm{IPRHCP}_{p,q}(\mu, \nu)\\ &= \left(\int_{\mathbf{E} \in \mathbb{S}_{d,q}} \mathrm{HCP}_p^p \left(P_{\mathbf{E}\#}\mu, P_{\mathbf{E}\#}\nu \right) \mathrm{d}\sigma(\mathbf{E}) \right)^{\frac{1}{p}}, \qquad (7)\end{aligned}$$

where σ is the uniform distribution on $\mathbb{S}_{d,q}$.

Next, we demonstrate that IPRHCP distance is a valid distance metric and reveal the relations between IPRHCP distance and other metrics, including the p-order SW distance [15] and the p-order q-dimensional integral projection robust Wasserstein distance [40], denoted as $\mathrm{IPRW}_{p,q}$.

Theorem 4: $\mathrm{IPRHCP}_{p,q}$ is a well-defined metric in $\mathscr{P}_\infty(\mathbb{R}^d)$, and we have $\mathrm{IPRW}_{p,q}(\mu, \nu) \leq \mathrm{IPRHCP}_{p,q}(\mu, \nu), \ \forall \mu, \nu \in \mathscr{P}_\infty(\mathbb{R}^d)$.

In practice, the expectation in (7) can be approximated using a Monte Carlo scheme: We first randomly and uniformly draw several matrices from the set of orthogonal matrices $\mathbb{S}_{d,q}$. We then project the distributions to subspace $\mathbf{E}$ and compute the HCP distance between the projected samples. Finally, we replace the expectation on the right hand side of (7) with a finite-sample average.

Theorem 5: Given two probability measures $\mu, \nu \in \mathscr{P}_\infty(\mathbb{R}^d)$, we have $\mathrm{SW}_p^p(\mu, \nu) \leq \alpha_{q,p} \mathrm{IPRHCP}_{p,q}^p(\mu, \nu)$, where $\alpha_{q,p} = \int_{\mathbb{S}_{q,1}} \|\theta\|_p^p d\theta / q \leq 1$. As a special case, when $p = 2$, one has $\alpha_{q,2} = 1/q$ and $\mathrm{SW}_2(\mu, \nu) \leq \mathrm{IPRHCP}_{2,q}(\mu, \nu)/\sqrt{q}$.

Corollary 5.1: If we replace $\mathbb{S}_{d,q}$ in (7) with matrix set $\{\mathbf{E} \in \mathbb{R}^{d \times q} : \mathbf{E}^\top \mathbf{E} = \mathbf{J}_q\}$ where $\mathbf{J}_q$ is a $q \times q$ all-ones matrix, we have $\mathrm{IPRHCP}_{p,q}(\mu, \nu) = q^{1/p} \mathrm{SW}_p(\mu, \nu), \forall \mu, \nu \in \mathscr{P}_\infty(\mathbb{R}^d)$.

IPRHCP shares a similar sense to SW. As shown in Theorem 5 and Corollary 5.1, we provided some inequalities and equalities between IPRHCP and SW to illustrate their relationship. We provide the following theorem to show IPRHCP overcomes curse-of-dimensionality.

Theorem 6: Suppose that probability measures $\mu, \nu \in \mathscr{P}_\infty(\mathbb{R}^d)$. Let $\{x_i\}_{i=1}^n$ and $\{y_i\}_{i=1}^n$ be two i.i.d. samples, which are generated from probability measures μ and ν, respectively. Let $\{x_{\mathbf{E},(i)^*}\}_{i=1}^n$ and $\{y_{\mathbf{E},(i)^*}\}_{i=1}^n$ be the sorted samples of $\{\mathbf{E}^T x_i\}_{i=1}^n$ and $\{\mathbf{E}^T y_i\}_{i=1}^n$ along the Hilbert curves $H_{P_{\mathbf{E}\#}\mu}$ and $H_{P_{\mathbf{E}\#}\nu}$, respectively. Based on the definition of $\overline{\mathrm{HCP}}(\mu_n, \nu_n)$, we can define

$$\begin{aligned}&\overline{\mathrm{IPRHCP}}_{p,q}(\mu_n, \nu_n)\\ &\quad= \left(\int_{\mathbf{E} \in \mathbb{S}_{d,q}} \overline{\mathrm{HCP}}_p^p \left(P_{\mathbf{E}\#}\mu_n, P_{\mathbf{E}\#}\nu_n \right) \mathrm{d}\sigma(\mathbf{E}) \right)^{\frac{1}{p}}\\ &\quad= \left(\int_{\mathbf{E} \in \mathbb{S}_{d,q}} \frac{1}{n} \sum_{i=1}^n \|x_{\mathbf{E},(i)^*} - y_{\mathbf{E},(i)^*})\|_p^p \mathrm{d}\sigma(\mathbf{E}) \right)^{\frac{1}{p}},\end{aligned}$$

where μ_n and ν_n are the empirical version of μ and ν, respectively. Then, we have

$$|\mathbb{E}\overline{\mathrm{IPRHCP}}_{p,q}(\mu_n, \nu_n) - \mathrm{IPRHCP}_{p,q}(\mu, \nu)| \lesssim O(n^{-\frac{1}{2\max\{p,q\}}}).$$

B. Projection Robust Hilbert Curve Projection Distance

The IPRHCP distance considers the integration of the HCP distances defined in all q-dimensional subspaces. When assuming the two distributions differ only on one low-dimensional subspace, as the projection robust Wasserstein (PRW) distance [39] does, we can avoid the integration and just consider the maximal possible HCP distance among all projections, which leads to the

Algorithm 2: Computation of PRHCP Distance:

```
1: Input: $(\mathbf{X}=\{x_i\}_{i=1}^n, \boldsymbol{a})$, $(\mathbf{Y}=\{y_j\}_{j=1}^m, \boldsymbol{b})$, $k$, $q$
2: Initialize $\mathbf{U}=\mathbf{\Omega}=\mathbf{I}_d$, $t=0$, $\tau=1$
3: While not converge
  a) $\mathbf{P} \leftarrow$ Algorithm 1 $[(\{\mathbf{U}^\top x_i\}_{i=1}^n, \boldsymbol{a}), (\{\mathbf{U}^\top y_j\}_{j=1}^m, \boldsymbol{b}), k]$. $O((n\log(n)+m\log(m))d)$
  b) $\mathbf{U}\in\mathbb{R}^{d\times q} \leftarrow$ top $q$ singular vectors of the matrix $(\mathbf{X}-\mathrm{diag}(\boldsymbol{a}^{-1})\mathbf{PY})$ with weight $\boldsymbol{a}$. $O((n+m)d^2)$
  c) $\mathbf{\Omega}\leftarrow(1-\tau)\mathbf{\Omega}+\tau\mathbf{UU}^\top$, and then $\mathbf{U}\leftarrow$ top $q$ eigenvectors of $\mathbf{\Omega}$. $O(d^2q+d^3)$
  d) $t\leftarrow t+1$, $\tau\leftarrow 2/(2+t)$
4: Output: The coupling $\mathbf{P}$, and $\mathrm{PRHCP}_{p,q} = (\sum_{(i,j)\in\{(i,j)|P_{ij}\neq 0\}} \|\mathbf{U}^\top x_i - \mathbf{U}^\top y_j\|_p^p P_{ij})^{1/p}$
```

proposed projection robust Hilbert curve projection (PRHCP) distance.

Definition 3: Suppose that probability measures $\mu,\nu\in\mathscr{P}_\infty(\mathbb{R}^d)$. The p-order q-dimensional projection robust Hilbert curve projection distance is defined as

$$\mathrm{PRHCP}_{p,q}(\mu,\nu)=\sup_{\mathbf{E}\in\mathbb{S}_{d,q}}\mathrm{HCP}_p\left(P_{\mathbf{E}\#}\mu, P_{\mathbf{E}\#}\nu\right). \quad (8)$$

The PRHCP distance is also a valid distance.

Theorem 7: $\mathrm{PRHCP}_{p,q}(\mu,\nu)$ is a well-defined metric in $\mathscr{P}_\infty(\mathbb{R}^d)$, and we have $\mathrm{PRW}_{p,q}(\mu,\nu)\leq\mathrm{PRHCP}_{p,q}(\mu,\nu)$, $\forall\mu,\nu\in\mathscr{P}_\infty(\mathbb{R}^d)$.

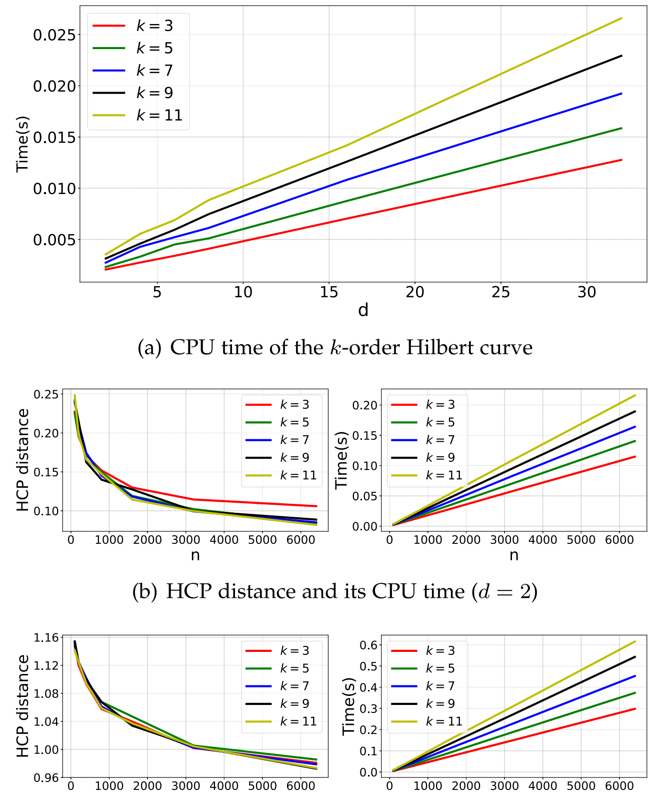
In practice, given the samples of the probability measures, i.e., the sample matrices $\mathbf{X}=[x_i^\top]\in\mathbb{R}^{n\times d}$ and $\mathbf{Y}=[y_j^\top]\in\mathbb{R}^{m\times d}$, we consider an EM-like optimization scheme to calculate the empirically PRHCP distance, i.e., we optimize the transport plan $\mathbf{P}$ and the $d\times q$ orthogonal matrix $\mathbf{E}$ alternately and iteratively. Details for calculating PRHCP distance are summarized in Algorithm 2. This algorithm is similar to the one for calculating the subspace robust Wasserstein distance in [38], except that the transport plan is calculated by the HCP distance. As we observed in numerical experiments, Algorithm 2 performs well for high-dimensional cases and is robust to noise. Theoretical justification for these observations is left for future work.

Computational Cost: For brevity, we consider the case that $n=m>d>q$. Step 3(a) requires $O(n\log(n)d)$ time, as discussed in the last section. Recall that there are at most $(n+m)$ nonzero elements in $\mathbf{P}$, and thus Step 3(b) requires only $O(n+m)d^2$ time. The cost for Step 3(c) involves $O(d^2q)$ for $\mathbf{UU}^\top$ and $O(d^3)$ for solving the eigen-decomposition problem, respectively. Thus, the overall complexity of Algorithm 2 is $O(n\log(n)dL+nd^2L)$, where L is the number of iterations.

The proofs of above Theorems and their corollaries are given in Appendix, available online.

V. Experiments

To demonstrate the feasibility and efficiency of our HCP distance and its variants, we conducted extensive numerical experiments and compared them with the main-stream competitors, including maximum mean discrepancy (MMD), Wasserstein distance, Sinkhorn distance [14], SW distance [15], max-SW distance [21], GSW distance [16], TSW distance [17], and PRW distance [39]. For all the distances, we considered the Euclidean cost, i.e., $p=2$. We use k-order Hilbert curves with $k=5\log(n)$. We set the dimension for the intrinsic space as $q=2$ for PRW, IPRHCP, and PRHCP. All experiments are implemented by an AMD 3600 CPU and an RTX 1080Ti GPU. For each experiment, we replicate it 100 times and record the average performance.



(a) CPU time of the k-order Hilbert curve

(b) HCP distance and its CPU time ($d=2$)

(c) HCP distance and its CPU time ($d=10$)

Fig. 4. (a) CPU time for generating the k-order Hilbert curve versus d when $n=100$. (b) Left: HCP distance versus n when $d=2$. Right: CPU time for generating the k-order Hilbert curve versus n when $d=2$. (c) Left: HCP distance versus n when $d=10$. Right: CPU time for generating the k-order Hilbert curve versus n when $d=10$.

A. Analytic Experiments on Synthetic Data

1) Robustness and Efficiency Analysis: The performance of our HCP distance is mainly determined by three factors: (1) the order of the Hilbert curve; (2) the dimension of sample; and (3) the number of samples. To demonstrate the robustness and efficiency of our HCP distance, we test it on synthetic data and analyze the influences of the above three factors.

Specifically, we generate two sample sets of size n from the uniform distribution on the unit hypercube $[0,1]^d$ and we calculate the HCP distance between these two sample sets. When calculating the HCP distance, the k-order Hilbert curve is applied. The results in Fig. 4(a) indicate the computational cost for generating the k-order Hilbert curve is linear to d. Fig. 4(b) and (c) show the average HCP distances and the average CPU time for generating the k-order Hilbert curve versus different

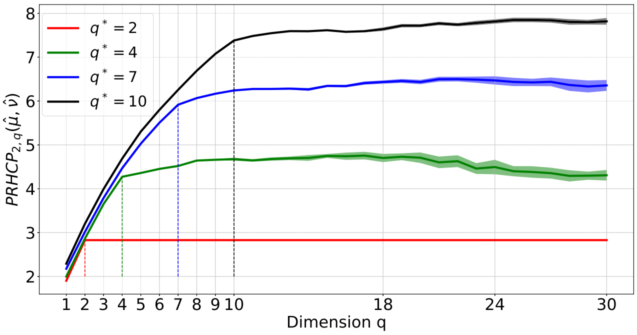


Fig. 5. $\mathrm{PRHCP}_{2,q}(\mu_n,\nu_n)$ versus the dimension q for $q^*=2,4,7,10$.

n's when $d=2$ or 10, respectively. From these two figures, we observe that the HCP distance is not sensitive to the choice of k, as long as k is not too small (i.e., $k>3$). We also observe that the computational cost for generating the k-order Hilbert curve is linear to n.

For the variant of our HCP distance, i.e., the PRHCP distance, one more factor should be considered — the dimension of subspace. Ideally, this distance should be robust to the setting of q as long as q is equal to or larger than the dimension of the effective subspaces.

To demonstrate their robustness to q, we follow the settings in [38], [39], considering a uniform distribution $\mu=\mathcal{U}([-1,1])^d$ and its pushforward under a map T, i.e., $\nu=T_{\#}\mu$. Here, the map $T(x)=x+2\,\mathrm{sign}(x)\odot(\sum_{i=1}^{q^*}e_i)$, where sign is taken elementwise, $q^*=2,4,7,10$, and $(e_1,\ldots,e_d)$ is the canonical basis of $\mathbb{R}^d$. Obviously, the map T splits the hypercube into four different hyper-rectangles, and the dimension of the effective subspace equals to q^*.

Setting $d=50$ and $n=100$, we calculate the PRHCP distance under different q's. Fig. 5 shows the PRHCP distance increases rapidly when $q<q^*$ and tends to be stable and consistent when $q\geq q^*$. Such an observation indicates PRHCP can dig out useful subspace information effectively.

2) Comparisons in High-Dimensional Scenarios: As shown in Fig. 1, our HCP distance provides an effective and efficient surrogate of Wasserstein distance for 2D data. Here, we further compare various metrics on approximating Wasserstein distance for high-dimensional data. In particular, let $\{x_i\}_{i=1}^n$ and $\{y_i\}_{i=1}^n$ be i.i.d. samples generated from two Gaussian distributions, i.e., $\mathcal{N}_d(\mathbf{0}_d,\mathbf{\Sigma}_X)$ and $\mathcal{N}_d(\mathbf{0}_d,\mathbf{\Sigma}_Y)$, respectively. We consider three different settings as follows.

1) $\boldsymbol{\mu}_X=\mathbf{0}_d,\quad \boldsymbol{\mu}_Y=(\theta,\theta,0,\ldots,0)^\top,\quad \mathbf{\Sigma}_X=\mathbf{\Sigma}_Y=\boldsymbol{I}_d$.
2) $\boldsymbol{\mu}_X=\mathrm{diag}(3\boldsymbol{I}_2,\boldsymbol{I}_{d-2}),\boldsymbol{\mu}_Y=\mathrm{diag}(\theta\boldsymbol{I}_2,\boldsymbol{I}_{d-2})$.
3) $\boldsymbol{\mu}_X=\mathrm{diag}(3\boldsymbol{I}_2,\boldsymbol{I}_{d-2}),\boldsymbol{\mu}_Y=\mathrm{diag}(\theta\boldsymbol{I}_2+3\theta\boldsymbol{B}_2,\boldsymbol{I}_{d-2})$.

where $\boldsymbol{I}_d$ and $\boldsymbol{B}_d$ are identity and backward identity matrices with size $(d\times d)$, respectively. In each of the three settings, the distance between the two distributions is controlled by a hyperparameter θ.

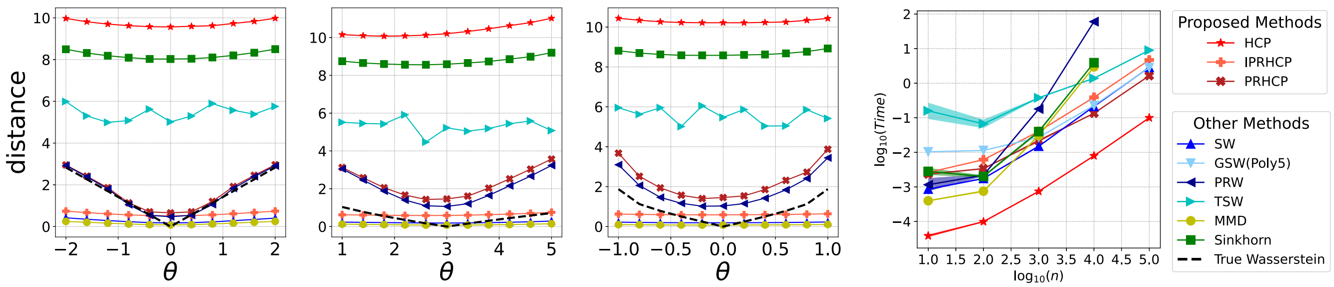
We set $n=200$, $d=50$. Given different θ's, we generate different samples and calculate the distance between the two sample sets under different metrics. Fig. 6(a) shows the averaged distance in 100 trials. Taking the true Wasserstein distance between the two Gaussian densities as a benchmark, $\mathrm{W}_2(\mathcal{N}_d(\mathbf{0}_d,\mathbf{\Sigma}_X),\mathcal{N}_d(\mathbf{0}_d,\mathbf{\Sigma}_Y))=\mathrm{tr}(\mathbf{\Sigma}_X+\mathbf{\Sigma}_Y-2(\mathbf{\Sigma}_X^{\frac{1}{2}}\mathbf{\Sigma}_Y\mathbf{\Sigma}_X^{\frac{1}{2}})^{\frac{1}{2}})^{\frac{1}{2}}$, we observe that most of the metrics, including our HCP distance, suffer from the curse-of-dimensionality or lack of robustness to noise, i.e., their distances are not sensitive to the parameter θ. Among these metrics, the PRW distance and our PRHCP distance are the only two that provide reasonable distances — they perform similarly as the true Wasserstein distance. In other words, although the HCP distance suffers from the curse-of-dimensionality, this problem can be mitigated by combining the HCP distance with the subspace projection strategy, leading to the PRHCP distance. Besides the comparison on the effectiveness, we also compare the CPU time for different metrics. Fig. 6(b) shows the CPU time (in seconds) versus different n's. The time for our methods, including the HCP distance and its variants, is approximately linear to n. Compared to other metrics, our HCP requires significantly less time than all the competitors, and its two variants are at least comparable to other distances in runtime. Especially, our PRHCP distance works as well as the PRW distance does in high-dimensional scenarios, but its runtime is much less than the PRW distance's runtime, which demonstrates its superiority on both effectiveness and efficiency.

Additionally, we consider a synthetic example to demonstrate the empirical sample complexity of the proposed distances. We generate two samples of size n from the standard d-dimensional Gaussian distributions and we calculate the distances between these two samples w.r.t. different distance metrics. Fig. 7 shows the average distances versus n for $d=2$ and 20, respectively. We observe that when $d=20$, the Wasserstein distance and the HCP distance converge slowly as expected, while SW, IPRHCP, and PRHCP converge much faster. In the aspect of the empirical sample complexity, the slope of the curves indicates that our HCP distance is comparable to the Wasserstein distance, and our IPRHCP and PRHCP distances are comparable to the SW distance.

B. *Approximation of Wasserstein Flow*

1) Comparison on Synthetic Data: Following the experiment in [16], we consider the problem $\min_\mu \mathrm{W}_2(\mu,\nu)$, where ν is a fixed target distribution, and μ is the source distribution initialized as $\mu_0=\mathcal{N}(0,1)$ and updated iteratively via $\partial_t\mu_t=-\nabla\mathrm{W}_2(\mu_t,\nu)$. We consider four different distributions for the target ν, i.e., *Circle*, *Swiss Roll*, *25-Gaussian*, and *Puma*, and approximate the Wasserstein distance W_2 by SW, max-SW, GSW, max-GSW, and HCP. Each method applies one projection per iteration and sets the learning rate to be 0.01. The experiments are replicated one hundred times, and we record the averaged 2-Wasserstein distance between μ_t and ν at each iteration. The comparison for the methods on their convergence curves and the snapshots of their learning results when $t=150$ are shown in Fig. 8(a). We can find that applying HCP helps to accelerate the learning process and leads to better results.

Fig. 8(a) shows that using SW or its variants as the loss function may lead to slow convergence. Taking the *25-Gaussian* case as an example, we provide an intuitive explanation for



(a)

(b)

Fig. 6. Comparison for various metrics. (a) Distances versus different θ. (b) CPU time versus different n.

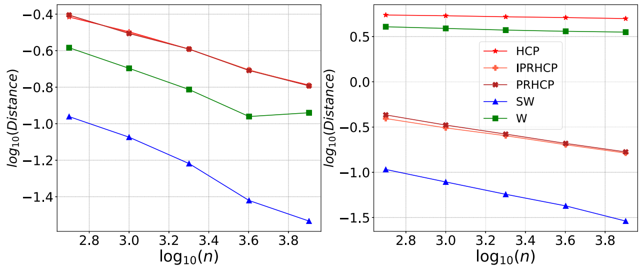


Fig. 7. Comparison for sample complexity. Left: $d = 2$. Right: $d = 20$. Each curve represents a distance versus n.

this phenomenon. In particular, we illustrate the iterations of SW, Max-SW, GSW, Max-GSW and HCP in Fig. 8(b). We observe that the flow w.r.t. SW and its variants go through 2 processes: first, red points spread out without covering the central Gaussian; second, they cover the central Gaussian slowly. Such an observation indicates linear projection fails to preserve high-dimensional data structure, especially when the data are multi-modal, and thus resulting in slow convergence. The proposed HCP distance, on the contrary, utilizes a Hilbert curve to preserve the structure of high-dimensional data and thus leads to faster convergence.

2) Color Transfer for Images: Besides testing on synthetic data, we consider the real-world color transfer task. As shown in Fig. 9(a), we transfer the color of a *Spring Forest* image to an *Autumn Forest* image. Each image is represented as nearly two million pixels in the RGB space ($d = 3$). Considering the large sample size, we use SW distance and HCP distance to approximate the Wasserstein flow, with the same learning hyperparameters. The comparison of the methods on their color transfer results and iterations are shown in Fig. 9. We can find that applying the HCP distance helps to accelerate the learning process. Quantitatively, it takes 496.7 seconds for the SW-based method and 57.3 seconds for our HCP-based method.

C. Data Classification

1) 3D Point Cloud Classification: For low-dimensional data like 3D points, our HCP distance is superior to other distances in their classification tasks. We consider the ModelNet10 dataset [72] that contains around 5,000 CAD objects from 10 categories. For each category, we randomly sample 50 objects for training and 30 object objects for testing. Following the work in [73], we randomly sample $n = 100, 200, 500, 1000, 2000$ points per object to get 3D point cloud data. We calculate the pairwise distance between the point clouds w.r.t. different distance metrics and then use the K-NN algorithm ($n_{neighbors} = 5$) to evaluate the classification accuracy on the testing set. We used the RBF kernel for MMD, and we set the number of slices $n_s = 10$ for SW, $n_s = 10, T = 7, \kappa = 4$ for TSW. Here, T is the predefined deepest level of the tree, n_s is the number of slices and κ is the number of clusters. Table I summarizes the averaged performance of each metric in 10 trials. Our HCP outperforms other distances on accuracy and requires the least amount of time.

2) Document Classification: As a typical high-dimensional data classification problem, document classification can be achieved by comparing the Wasserstein distance between two documents' word embedding sets, as the Word Mover distance [1] does. Our PRHCP distance provides an efficient surrogate of the Wasserstein distance in this problem, which is demonstrated by the following experiment. Following the preprocessing used in [1], we obtain 3,000 documents belonging to three categories from the TWITTER dataset, in which each document is represented as a set of 300-dimensional word embeddings derived by the pre-trained *word2vec* model [74]. We randomly split the dataset into 80% for training and 20% for testing. Similar to the above point cloud classification experiment, we use the K-NN algorithm ($n_{neighbors} = 10$) based on different metrics and evaluate the averaged learning results in 10 trials. In this experiment, we set the number of slices $n_s = 20$ for SW, $n_s = 10, T = 7, \kappa = 4$ for TSW. For PRHCP, we first find the 10-dimensional subspace based on the training data by Algorithm 2 and project testing data to the subspace. Table II shows that our PRHCP distance outperforms other distances on classification accuracy, and its runtime is comparable to TSW.

D. Generative Modeling

The proposed distances help us to design new members of Wasserstein autoencoder (WAE) [9]. In particular, when training autoencoders, we leverage HCP, IPRHCP, and PRHCP to penalize the distance between the latent prior distribution

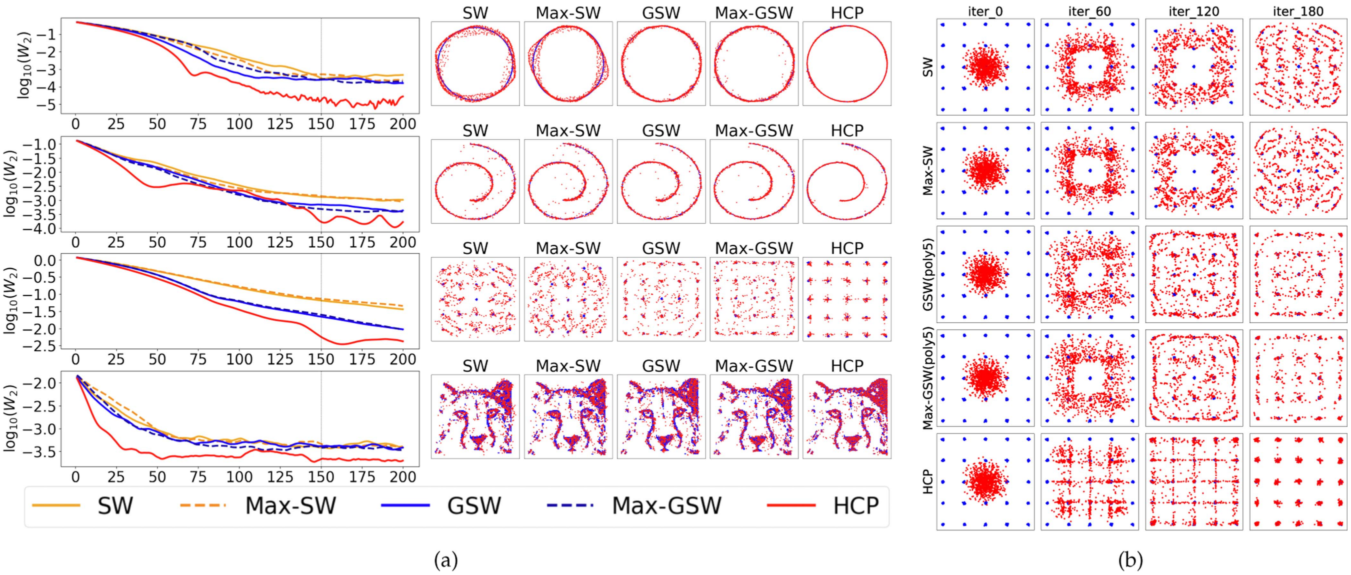


Fig. 8. (a) Left: Log 2-Wasserstein distance between the source and target distributions versus the number of iterations t. Right: A snapshot when $t = 150$. (b) Iterations of different distances based flow.

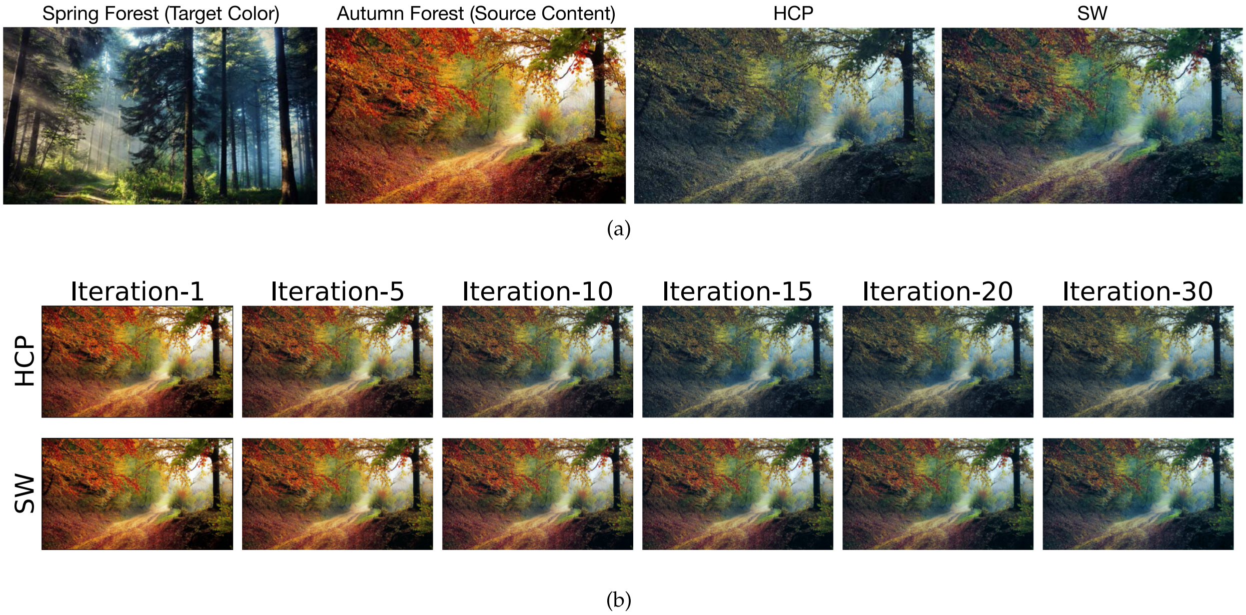


Fig. 9. (a) The images from left to right are the image with the target color, the image with the source content, and the color transfer results achieved based on our HCP distance and the SW distance, respectively. (b) The first row is iterations of color transfer based on our HCP distance. The second row is iterations of color transfer based on the SW distance.

and the expected posterior distribution, which leads to three different generative models, denoted as HCP-AE, IPRHCP-AE, and PRHCP-AE. We test these three models in image generation tasks and compare them with the original Wasserstein autoencoder (WAE) [9] and the well-known sliced Wasserstein autoencoder (SWAE) [22].

1) HCP-Based Autoencoders: We first test the capability of HCP-AE in shaping the low-dimensional latent space of the encoder. We train an HCP-AE to encode the MNIST dataset [75] to a two-dimensional latent space (for the sake of visualization), in which both the autoencoding architecture and the hyperparameter setting are the same as those in [22]. A simple autoencoder with mirrored classic deep convolutional neural networks with 2D average poolings, Leaky-ReLu activation functions, and upsampling layers in the decoder is used. The batch size is 500 and the number of projections for SWAE is 40.

To evaluate the performance, we randomly selected a sample of size 1,000 from the encoded test data points (blue points in

TABLE I
COMPARISONS ON 3D POINT CLOUD CLASSIFICATION

Method	Accuracy(%)					CPU time(s)				
	n=100	n=200	n=500	n=1000	n=2000	n=100	n=200	n=500	n=1000	n=2000
HCP	$73.3_{\pm 1.3}$	$79.2_{\pm 2.8}$	$\mathbf{81.8}_{\pm 0.2}$	$\mathbf{81.0}_{\pm 0.1}$	$\mathbf{82.3}_{\pm 0.7}$	**17.9**	**34.3**	**90.4**	**186.7**	374.5
SW	$71.5_{\pm 2.2}$	$77.0_{\pm 1.0}$	$79.2_{\pm 0.5}$	$79.3_{\pm 0.7}$	$80.5_{\pm 1.0}$	123.9	202.1	410.5	830.8	1808.7
TSW	$72.7_{\pm 3.2}$	$75.0_{\pm 2.5}$	$77.0_{\pm 2.2}$	$77.7_{\pm 1.5}$	$78.0_{\pm 1.0}$	120.7	137.4	165.8	217.2	**282.1**
GSW(Poly5)	$68.3_{\pm 2.0}$	$73.2_{\pm 0.5}$	$76.0_{\pm 0.7}$	$76.8_{\pm 1.2}$	$77.7_{\pm 0.3}$	839.7	940.6	1228.8	1782.4	2970.2
MMD	$66.8_{\pm 4.2}$	$71.8_{\pm 0.2}$	$74.0_{\pm 0.7}$	/	/	153.6	385.7	1785.2	/	/
Sinkhorn	$\mathbf{77.0}_{\pm 2.7}$	$\mathbf{80.8}_{\pm 0.5}$	$81.5_{\pm 0.8}$	/	/	768.7	2305.0	6184.6	/	/

* "/" means that we fail to get a result in 10,000 seconds.

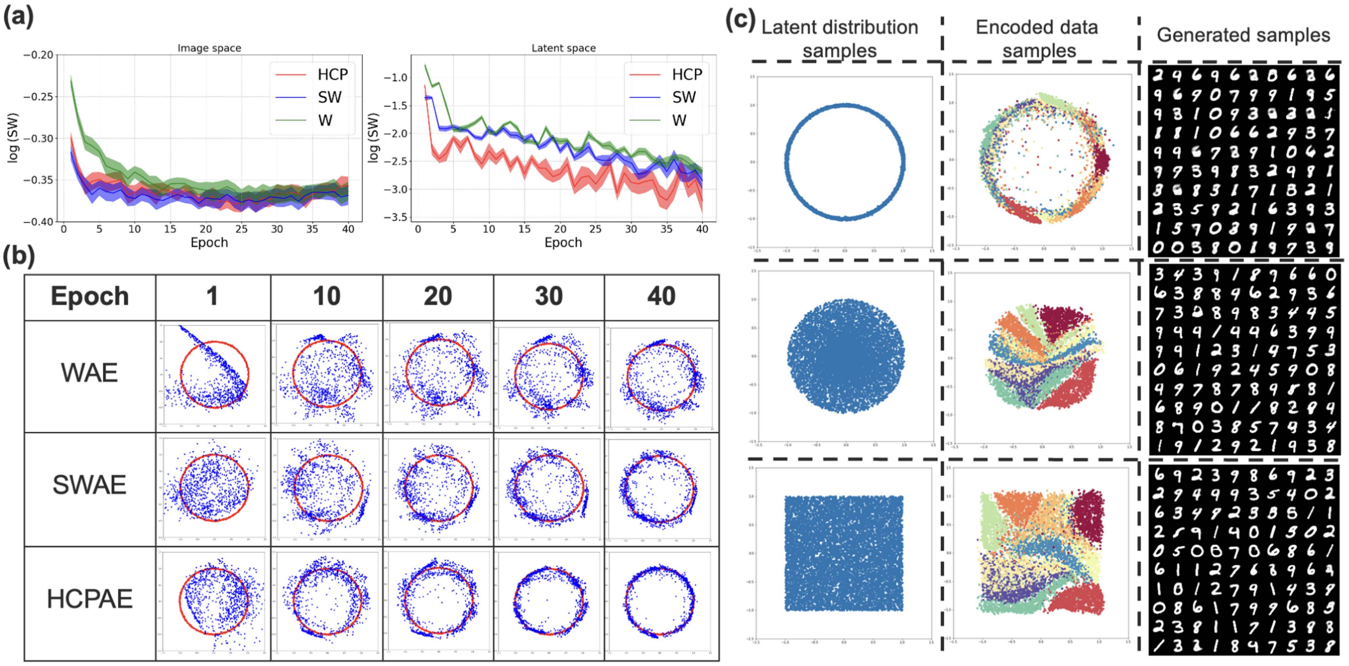


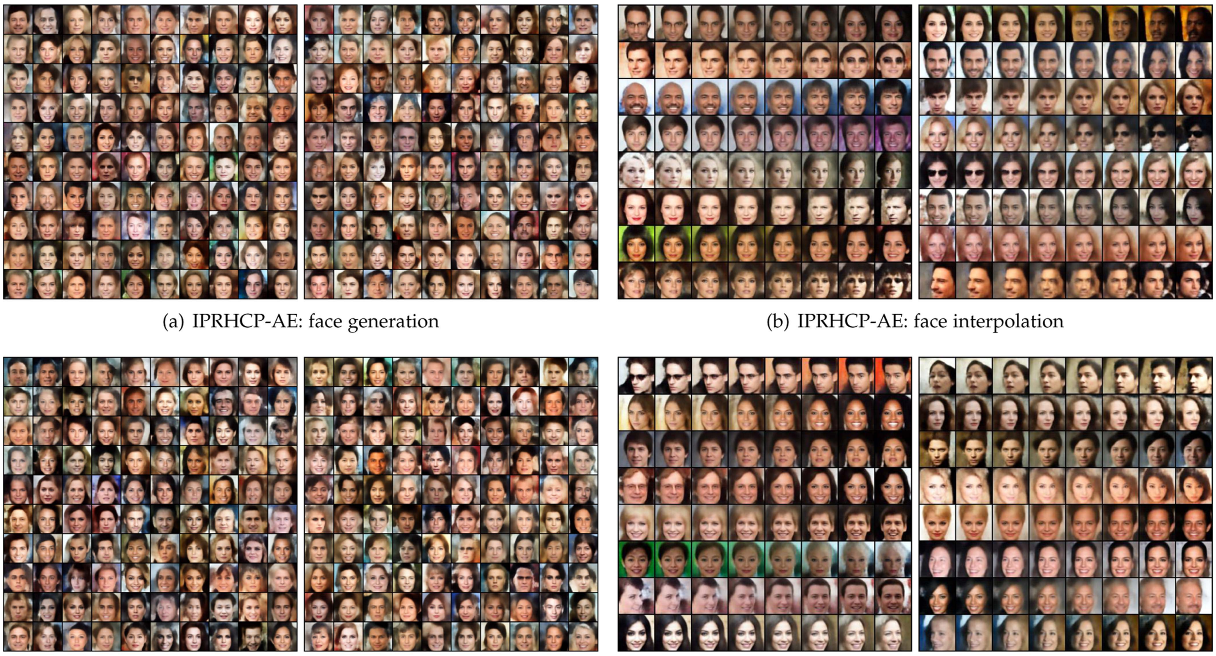
Fig. 10. (a): SW distances between the target sample and the encoded testing sample w.r.t. the image space (left) and the latent space (right); (b) Visualization of these two sample in the latent space during training; (c) Visualization of the encoded samples and the generated images.

TABLE II
COMPARISONS ON DOCUMENT CLASSIFICATION

Method	Accuracy(%)	CPU time(s)
PRHCP	$\mathbf{74.6}_{\pm 0.9}$	669.7
TSW	$71.2_{\pm 0.6}$	**287.7**
Sinkhorn	$70.0_{\pm 0.4}$	10106.7
SW	$68.6_{\pm 1.1}$	2789.7

Fig. 10(b)) and a random sample from the target prior distribution in the latent space (red points in Fig. 10(b)). We observed that our HCP-AE convergences much faster than other methods in the latent space. Moreover, the SW distances versus the number of epochs w.r.t. the image space and the latent space are shown in Fig. 10(a). We observed that though these three methods perform similarly in the image space, our HCP-AE converges much faster in the latent space. Fig. 10(c) visualizes the samples from two different prior distributions in the latent space, the encoded data samples via HCP-AE, and their generated images. The latent codes indeed obey the prior distributions, which reflects the clustering structure of the digits. Accordingly, the learned models are able to generate high quality digit images.

2) IPRHCP and PRHCP-Based Autoencoders: Second, we test the feasibility of IPRHCP-AE and PRHCP-AE in the cases with high-dimensional latent space. For fairness, all the autoencoders have the same DCGAN-style architecture [76] and hyperparameters: the learning rate is 0.001; the optimizer is Adam [77] with $\beta_1 = 0.9$ and $\beta_2 = 0.999$; the number of epochs is 50; the batch size is 100; the weight of regularizer γ is 1; the dimension of latent code is 8 for MNIST and 64 for CelebA; the number of random projections is 50. All the autoencoders use Euclidean distance as the distance between samples, which means the reconstruction loss is the mean-square error (MSE). We compare the proposed methods with the baselines on i) the reconstruction loss on testing samples; ii) the Fréchet Inception Distance (FID) [78] between 10,000 testing samples and 10,000



(a) IPRHCP-AE: face generation

(b) IPRHCP-AE: face interpolation

(c) PRHCP-AE: face generation

(d) PRHCP-AE: face interpolation

Fig. 11. Performance of IPRHCP-AE and PRHCP-AE on face generation and interpolation.

TABLE III
COMPARISONS FOR VARIOUS METHODS ON LEARNING IMAGE GENERATORS

Method	MNIST		CelebA	
	Rec. loss	FID	Rec. loss	FID
WAE	11.30	$54.61_{\pm 0.16}$	68.94	$58.12_{\pm 0.73}$
SWAE	13.68	$42.96_{\pm 0.53}$	68.57	$84.52_{\pm 0.44}$
IPRHCP-AE	11.72	$\mathbf{40.03}_{\pm 0.13}$	69.40	$\mathbf{56.00}_{\pm 0.08}$
PRHCP-AE	**10.07**	$42.87_{\pm 0.46}$	**66.65**	$67.82_{\pm 0.21}$

randomly generated samples. Table III lists the main differences between IPRHCP-AE, PRHCP-AE and these baselines. Among these autoencoders, our IPRHCP-AE and PRHCP-AE are comparable to the considered alternatives on both testing reconstruction loss and FID score. Some image generation and interpolation results achieved by our methods are shown in Fig. 11.

VI. CONCLUSION

In this work, we proposed a novel metric for distribution comparison, named Hilbert curve projection (HCP) distance. Thanks to the locality-preserving property of the Hilbert curve projection, the HCP distance enjoys several advantages over the Wasserstein and SW distance. Furthermore, we develop two variants of the HCP distance using (learnable) subspace projections to mitigate the curse-of-dimensionality.

Limitations and Future Work: Currently, HCP distance still suffers from some limitations. Like the Wasserstein distance, the HCP distance may not be robust to outliers. To address this problem, we could follow the methods in [79], [80], [81] by relaxing marginal constraints through penalty functions such as Kullback-Leibler divergence, total variation distance, and χ^2 divergence. Another possible solution is to consider partial OT methods instead of sorting in Step 3 of Algorithm 1. Besides, HCP distance could not quantify the discrepancy between two measures with different masses. We could follow the idea of (sliced) unbalanced optimal transport [82], [83], [84] by considering unbalanced OT methods instead of sorting in Step 3 of Algorithm 1. We left these directions for our future work. In addition, we plan to apply these new metrics to more learning problems and extend them to Gromov-Wasserstein distance [85], multi-marginal optimal transport [86], [87], and barycenter problems [88], [89]. Additionally, we will explore the theoretical results for other formulations of the Hilbert curve, such as the adaptive Hilbert curve, which works well in practice. And there is much literature on Hilbert sort, such as parallel Hilbert sort [90] and online Hilbert sort [68], [91], which may be extended.

REFERENCES

[1] M. Kusner, Y. Sun, N. Kolkin, and K. Weinberger, “From word embeddings to document distances,” in *Proc. Int. Conf. Mach. Learn.*, PMLR, 2015, pp. 957–966.

[2] G. Huang, C. Guo, M. J. Kusner, Y. Sun, F. Sha, and K. Q. Weinberger, “Supervised word mover’s distance,” in *Proc. Adv. Neural Inf. Process. Syst.*, 2016, pp. 4869–4877.

[3] A. Rakotomamonjy, A. Traoré, M. Berar, R. Flamary, and N. Courty, “Distance measure machines,” 2018, *arXiv:1803.00250*.

[4] I. Goodfellow et al., “Generative adversarial nets,” in *Proc. Adv. Neural Inf. Process. Syst.*, 2014, pp. 2672–2680.

[5] D. P. Kingma and M. Welling, “Auto-encoding variational bayes,” in *Proc. Int. Conf. Learn. Representations*, 2014, pp. 1–14.

[6] M. Arjovsky, S. Chintala, and L. Bottou, “Wasserstein generative adversarial networks,” in *Proc. Int. Conf. Mach. Learn.*, 2017, pp. 214–223.

[7] A. Gretton, K. M. Borgwardt, M. J. Rasch, B. Schölkopf, and A. Smola, "A kernel two-sample test," *J. Mach. Learn. Res.*, vol. 13, no. 1, pp. 723–773, 2012.

[8] C. Villani, *Optimal Transport: Old and New*, vol. 338. Berlin, Germany: Springer, 2009.

[9] I. Tolstikhin, O. Bousquet, S. Gelly, and B. Schölkopf, "Wasserstein autoencoders," in *Proc. 6th Int. Conf. Learn. Representations*, 2018, pp. 1–16.

[10] Y. Brenier, "A homogenized model for vortex sheets," *Arch. Rational Mechanics Anal.*, vol. 138, no. 4, pp. 319–353, 1997.

[11] J.-D. Benamou, Y. Brenier, and K. Guittet, "The Monge–Kantorovitch mass transfer and its computational fluid mechanics formulation," *Int. J. Numer. Methods Fluids*, vol. 40, no. 1/2, pp. 21–30, 2002.

[12] Y. Rubner, L. J. Guibas, and C. Tomasi, "The earth mover's distance, multi-dimensional scaling, and color-based image retrieval," in *Proc. ARPA Image Understanding Workshop*, 1997, pp. 661–668.

[13] O. Pele and M. Werman, "Fast and robust earth mover's distances," in *Proc. IEEE 12th Int. Conf. Comput. Vis.*, 2009, pp. 460–467.

[14] M. Cuturi, "Sinkhorn distances: Lightspeed computation of optimal transport," in *Proc. Adv. Neural Inf. Process. Syst.*, 2013, pp. 2292–2300.

[15] N. Bonneel, J. Rabin, G. Peyré, and H. Pfister, "Sliced and Radon Wasserstein barycenters of measures," *J. Math. Imag. Vis.*, vol. 51, no. 1, pp. 22–45, 2015.

[16] S. Kolouri, K. Nadjahi, U. Simsekli, R. Badeau, and K. Gustavo, "Generalized sliced Wasserstein distances," in *Proc. Adv. Neural Inf. Process. Syst.*, 2019, Art. no. 24.

[17] T. Le, M. Yamada, K. Fukumizu, and M. Cuturi, "Tree-sliced variants of Wasserstein distances," in *Proc. Adv. Neural Inf. Process. Syst.*, 2019, Art. no. 1102.

[18] M. Bader, *Space-Filling Curves: An Introduction With Applications in Scientific Computing*, Berlin, Germany: Springer, 2012.

[19] D. J. Abel and D. M. Mark, "A comparative analysis of some two-dimensional orderings," *Int. J. Geographical Inf. Syst.*, vol. 4, no. 1, pp. 21–31, 1990.

[20] B. Moon, H. V. Jagadish, C. Faloutsos, and J. H. Saltz, "Analysis of the clustering properties of the Hilbert space-filling curve," *IEEE Trans. Knowl. Data Eng.*, vol. 13, no. 1, pp. 124–141, Jan./Feb. 2001.

[21] I. Deshpande et al., "Max-sliced Wasserstein distance and its use for GANs," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 10648–10656.

[22] S. Kolouri, P. E. Pope, C. E. Martin, and G. K. Rohde, "Sliced Wasserstein auto-encoders," in *Proc. Int. Conf. Learn. Representations*, 2018, pp. 1–19.

[23] J. Altschuler, J. Niles-Weed, and P. Rigollet, "Near-linear time approximation algorithms for optimal transport via Sinkhorn iteration," in *Proc. Adv. Neural Inf. Process. Syst.*, 2017, pp. 1961–1971.

[24] J. Altschuler, F. Bach, A. Rudi, and J. Niles-Weed, "Massively scalable sinkhorn distances via the Nyström method," in *Proc. Adv. Neural Inf. Process. Syst.*, 2019, pp. 4429–4439.

[25] P. Dvurechensky, A. Gasnikov, S. Omelchenko, and A. Tiurin, "Adaptive similar triangles method: A stable alternative to sinkhorn's algorithm for regularized optimal transport," 2017, *arXiv:1706.07622*.

[26] A. Thibault, L. Chizat, C. Dossal, and N. Papadakis, "Overrelaxed Sinkhorn–Knopp algorithm for regularized optimal transport," *Algorithms*, vol. 14, no. 5, pp. 143–158, 2021.

[27] M. Li, J. Yu, T. Li, and C. Meng, "Importance sparsification for sinkhorn algorithm," *J. Mach. Learn. Res.*, vol. 24, pp. 1–44, 2023.

[28] Q. Liao, J. Chen, Z. Wang, B. Bai, J. Shi, and H. Wu, "Fast sinkhorn I: An $O(N)$ algorithm for the Wasserstein-1 metric," *Commun. Math. Sci.*, vol. 20, no. 7, pp. 2053–2067, 2022.

[29] W. Guo, N. Ho, and M. Jordan, "Fast algorithms for computational optimal transport and Wasserstein barycenter," in *Proc. Int. Conf. Artif. Intell. Statist.*, PMLR, 2020, pp. 2088–2097.

[30] P. Dvurechensky, A. Gasnikov, and A. Kroshnin, "Computational optimal transport: Complexity by accelerated gradient descent is better than by Sinkhorn's algorithm," in *Proc. Int. Conf. Mach. Learn.*, PMLR, 2018, pp. 1367–1376.

[31] A. Genevay, M. Cuturi, G. Peyré, and F. Bach, "Stochastic optimization for large-scale optimal transport," in *Proc. Adv. Neural Inf. Process. Syst.*, 2016, pp. 3440–3448.

[32] Y. Xie, X. Wang, R. Wang, and H. Zha, "A fast proximal point method for computing exact Wasserstein distance," in *Proc. Int. Conf. Uncertainty Artif. Intell.*, PMLR, 2020, pp. 433–453.

[33] H. Wang and A. Banerjee, "Bregman alternating direction method of multipliers," in *Proc. Adv. Neural Inf. Process. Syst.*, 2014, pp. 2816–2824.

[34] J. Ye, P. Wu, J. Z. Wang, and J. Li, "Fast discrete distribution clustering using Wasserstein barycenter with sparse support," *IEEE Trans. Signal Process.*, vol. 65, no. 9, pp. 2317–2332, May 2017.

[35] H. Xu, J. Liu, D. Luo, and L. Carin, "Representing graphs via Gromov-Wasserstein factorization," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 45, no. 1, pp. 999–1016, Jan. 2023.

[36] K. Nguyen, N. Ho, T. Pham, and H. Bui, "Distributional sliced-Wasserstein and applications to generative modeling," in *Proc. Int. Conf. Learn. Representations*, 2020, pp. 1–12.

[37] M. Rowland, J. Hron, Y. Tang, K. Choromanski, T. Sarlos, and A. Weller, "Orthogonal estimation of Wasserstein distances," in *Proc. 22nd Int. Conf. Artif. Intell. Statist.*, PMLR, 2019, pp. 186–195.

[38] F. P. Paty and M. Cuturi, "Subspace robust Wasserstein distances," in *Proc. Int. Conf. Mach. Learn.*, PMLR, 2019, pp. 5072–5081.

[39] T. Lin, C. Fan, N. Ho, M. Cuturi, and M. Jordan, "Projection robust Wasserstein distance and Riemannian optimization," in *Proc. Adv. Neural Inf. Process. Syst.*, 2020, pp. 9383–9397.

[40] T. Lin, Z. Zheng, E. Chen, M. Cuturi, and M. I. Jordan, "On projection robust optimal transport: Sample complexity and model misspecification," in *Proc. Int. Conf. Artif. Intell. Statist.*, PMLR, 2021, pp. 262–270.

[41] K. Nguyen and N. Ho, "Revisiting sliced Wasserstein on images: From vectorization to convolution," 2022, *arXiv:2204.01188*.

[42] K. Nguyen and N. Ho, "Amortized projection optimization for sliced Wasserstein generative models," 2022, *arXiv:2203.13417*.

[43] A. Genevay, G. Peyré, and M. Cuturi, "Learning generative models with Sinkhorn divergences," in *Proc. Int. Conf. Artif. Intell. Statist.*, PMLR, 2018, pp. 1608–1617.

[44] J. Wu et al., "Sliced Wasserstein generative models," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 3713–3722.

[45] H. Xu, D. Luo, R. Henao, S. Shah, and L. Carin, "Learning autoencoders with relational regularization," in *Proc. Int. Conf. Mach. Learn.*, PMLR, 2020, pp. 10576–10586.

[46] C. Frogner, C. Zhang, H. Mobahi, M. Araya-Polo, and T. Poggio, "Learning with a Wasserstein loss," in *Proc. Adv. Neural Inf. Process. Syst.*, 2015, pp. 2053–2061.

[47] M. Togninalli, E. Ghisu, F. Llinares-López, B. Rieck, and K. Borgwardt, "Wasserstein Weisfeiler-Lehman graph kernels," in *Proc. Adv. Neural Inf. Process. Syst.*, 2019, Art. no. 578.

[48] M. Li, J. Yu, H. Xu, and C. Meng, "Efficient approximation of Gromov-Wasserstein distance using importance sparsification," *J. Comput. Graphical Statist.*, vol. 32, pp. 1512–1523, 2023.

[49] C. Meng, J. Yu, J. Zhang, P. Ma, and W. Zhong, "Sufficient dimension reduction for classification using principal optimal transport direction," in *Proc. Adv. Neural Inf. Process. Syst.*, 2020, pp. 4015–4028.

[50] R. Flamary, M. Cuturi, N. Courty, and A. Rakotomamonjy, "Wasserstein discriminant analysis," *Mach. Learn.*, vol. 107, no. 12, pp. 1923–1945, 2018.

[51] N. Courty, R. Flamary, A. Habrard, and A. Rakotomamonjy, "Joint distribution optimal transportation for domain adaptation," in *Proc. Adv. Neural Inf. Process. Syst.*, 2017, pp. 3733–3742.

[52] H. Xu, D. Luo, H. Zha, and L. C. Duke, "Gromov-Wasserstein learning for graph matching and node embedding," in *Proc. Int. Conf. Mach. Learn.*, PMLR, 2019, pp. 6932–6941.

[53] H. Xu, D. Luo, and L. Carin, "Scalable Gromov-Wasserstein learning for graph partitioning and matching," in *Proc. Adv. Neural Inf. Process. Syst.*, 2019, Art. no. 274.

[54] J. Rabin, S. Ferradans, and N. Papadakis, "Adaptive color transfer with relaxed optimal transport," in *Proc. IEEE Int. Conf. Image Process.*, 2014, pp. 4852–4856.

[55] C. Meng, Y. Ke, J. Zhang, M. Zhang, W. Zhong, and P. Ma, "Large-scale optimal transport map estimation using projection pursuit," in *Proc. Adv. Neural Inf. Process. Syst.*, 2019, pp. 8116–8127.

[56] M. Yurochkin, S. Claici, E. Chien, F. Mirzazadeh, and J. M. Solomon, "Hierarchical optimal transport for document representation," in *Proc. Adv. Neural Inf. Process. Syst.*, 2019, Art. no. 143.

[57] Z. He and A. B. Owen, "Extensible grids: Uniform sampling on a space filling curve," *J. Roy. Statist. Society: Ser. B. (Statist. Methodol.)*, vol. 78, no. 4, pp. 917–931, 2016.

[58] G. Zumbusch, *Parallel Multilevel Methods: Adaptive Mesh Refinement and Loadbalancing*, Berlin, Germany: Springer Science & Business Media, 2012.

[59] K. Fatras, Y. Zine, R. Flamary, R. Gribonval, and N. Courty, "Learning with minibatch Wasserstein: Asymptotic and gradient properties," in *Proc. Int. Conf. Artif. Intell. Statist.*, PMLR, 2020, pp. 2131–2141.

[60] B. Piccoli and F. Rossi, "Generalized Wasserstein distance and its application to transport equations with source," *Arch. Rational Mechanics Anal.*, vol. 211, pp. 335–358, 2014.

[61] J. Weed and F. Bach, "Sharp asymptotic and finite-sample rates of convergence of empirical measures in Wasserstein distance," *Bernoulli*, vol. 25, no. 4A, pp. 2620–2648, 2019.

[62] J. Xi and J. Niles-Weed, "Distributional convergence of the sliced Wasserstein process," in *Proc. Adv. Neural Inf. Process. Syst.*, 2022, pp. 13961–13973.

[63] V. M. Panaretos and Y. Zemel, "Statistical aspects of Wasserstein distances," *Annu. Rev. Statist. Appl.*, vol. 6, pp. 405–431, 2019.

[64] G. Peyré and M. Cuturi, "Computational optimal transport: With applications to data science," *Found. Trends Mach. Learn.*, vol. 11, no. 5/6, pp. 355–607, 2019.

[65] A. R. Butz, "Convergence with Hilbert's space-filling curve," *J. Comput. Syst. Sci.*, vol. 3, no. 2, pp. 128–146, 1969.

[66] A. R. Butz, "Alternative algorithm for Hilbert's space-filling curve," *IEEE Trans. Comput.*, vol. 100, no. 4, pp. 424–426, Apr. 1971.

[67] J. Skilling, "Programming the Hilbert curve," in *Proc. AIP Conf.*, American Institute of Physics, 2004, pp. 381–387.

[68] Y. Imamura, T. Shinohara, K. Hirata, and T. Kuboyama, "Fast Hilbert sort algorithm without using Hilbert indices," in *Proc. Int. Conf. Similarity Search Appl.*, Springer, 2016, pp. 259–267.

[69] A. Fabri and S. Pion, "CGAL: The computational geometry algorithms library," in *Proc. 17th ACM SIGSPATIAL Int. Conf. Adv. Geographic Inf. Syst.*, 2009, pp. 538–539.

[70] C. H. Hamilton and A. Rau-Chaplin, "Compact Hilbert indices: Space-filling curves for domains with unequal side lengths," *Inf. Process. Lett.*, vol. 105, no. 5, pp. 155–163, 2008.

[71] E. Bernton, P. E. Jacob, M. Gerber, and C. P. Robert, "Approximate Bayesian computation with the Wasserstein distance," *J. Roy. Statist. Society: Ser. B. (Statist. Methodol.)*, vol. 81, no. 2, pp. 235–269, 2019.

[72] Z. Wu et al., "3D ShapeNets: A deep representation for volumetric shapes," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2015, pp. 1912–1920.

[73] C. R. Qi, H. Su, K. Mo, and L. J. Guibas, "PointNet: Deep learning on point sets for 3D classification and segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 652–660.

[74] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," in *Proc. Adv. Neural Inf. Process. Syst.*, 2013, pp. 3111–3119.

[75] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," in *Proc. IEEE*, vol. 86, no. 11, pp. 2278–2324, Nov. 1998.

[76] A. Radford, L. Metz, and S. Chintala, "Unsupervised representation learning with deep convolutional generative adversarial networks," in *Proc. Int. Conf. Learn. Representations*, 2016, pp. 1–16.

[77] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *Proc. Int. Conf. Learn. Representations*, 2015, pp. 1–15.

[78] M. Heusel, H. Ramsauer, T. Unterthiner, B. Nessler, and S. Hochreiter, "GANs trained by a two time-scale update rule converge to a local Nash equilibrium," in *Proc. Adv. Neural Inf. Process. Syst.*, 2017, pp. 6629–6640.

[79] Y. Balaji, R. Chellappa, and S. Feizi, "Robust optimal transport with applications in generative modeling and domain adaptation," in *Proc. Adv. Neural Inf. Process. Syst.*, 2020, pp. 12934–12944.

[80] D. Mukherjee, A. Guha, J. M. Solomon, Y. Sun, and M. Yurochkin, "Outlier-robust optimal transport," in *Proc. Int. Conf. Mach. Learn.*, PMLR, 2021, pp. 7850–7860.

[81] K. Le, H. Nguyen, Q. M. Nguyen, T. Pham, H. Bui, and N. Ho, "On robust optimal transport: Computational complexity and barycenter computation," in *Proc. Adv. Neural Inf. Process. Syst.*, 2021, pp. 21947–21959.

[82] K. Pham, K. Le, N. Ho, T. Pham, and H. Bui, "On unbalanced optimal transport: An analysis of sinkhorn algorithm," in *Proc. Int. Conf. Mach. Learn.*, PMLR, 2020, pp. 7673–7682.

[83] L. Chizat, G. Peyré, B. Schmitzer, and F.-X. Vialard, "Scaling algorithms for unbalanced optimal transport problems," *Math. Comput.*, vol. 87, no. 314, pp. 2563–2609, 2018.

[84] T. Séjourné, C. Bonet, K. Fatras, K. Nadjahi, and N. Courty, "Unbalanced optimal transport meets sliced-Wasserstein," 2023, *arXiv:2306.07176*.

[85] F. Mémoli, "Gromov–Wasserstein distances and the metric approach to object matching," *Found. Comput. Math.*, vol. 11, no. 4, pp. 417–487, 2011.

[86] B. Pass, "Multi-marginal optimal transport: Theory and applications," *ESAIM: Math. Modelling Numer. Anal.-Modélisation Mathématique et Analyse Numérique*, vol. 49, no. 6, pp. 1771–1790, 2015.

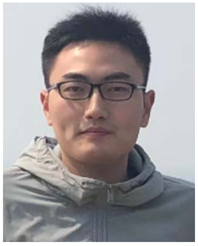
[87] I. Haasler, R. Singh, Q. Zhang, J. Karlsson, and Y. Chen, "Multi-marginal optimal transport and probabilistic graphical models," *IEEE Trans. Inf. Theory*, vol. 67, no. 7, pp. 4647–4668, Jul. 2021.

[88] M. Agueh and G. Carlier, "Barycenters in the Wasserstein space," *SIAM J. Math. Anal.*, vol. 43, no. 2, pp. 904–924, 2011.

[89] G. Peyré, M. Cuturi, and J. Solomon, "Gromov-Wasserstein averaging of kernel and distance matrices," in *Proc. Int. Conf. Mach. Learn.*, PMLR, 2016, pp. 2664–2672.

[90] J. Luitjens, M. Berzins, and T. Henderson, "Parallel space-filling curve generation through sorting," *Concurrency Comput. Pract. Exp.*, vol. 19, no. 10, pp. 1387–1402, 2007.

[91] I. Kamel and C. Faloutsos, "Hilbert R-tree: An improved R-tree using fractals," in *Proc. 20th Int. Conf. Very Large Data Bases*, Sep. 1994, pp. 500–509.



Tao Li received the BS degree in mathematics from Nanjing University, in 2019. He is currently working toward the PhD degree with the Institute of Statistics and Big Data, Renmin University of China. His research interests include optimal transport problems, generative model, sufficient dimension reduction, and variable selection.

Cheng Meng received the PhD degree from the Department of Statistics, University of Georgia, in 2020. He is an assistant professor (tenure-track) with the Institute of Statistics and Big Data, Renmin University of China. His research interests include numerical linear algebra, optimal transport problems, sufficient dimension reduction, nonparametric statistics, and machine learning.

Hongteng Xu (Member, IEEE) received the PhD degree from the School of Electrical and Computer Engineering, Georgia Institute of Technology (Georgia Tech), in 2017. He is an associate professor (tenure-track) with the Gaoling School of Artificial Intelligence, Renmin University of China. From 2018 to 2020, he was a senior research scientist with Infinia ML Inc. In the same time period, he is a visiting faculty member with the Department of Electrical and Computer Engineering, Duke University. His research interests include machine learning and its applications, especially optimal transport theory, sequential data modeling and analysis, deep learning techniques, and their applications in computer vision and data mining.

Jun Yu received the PhD degree in statistics from Peking University, China, in 2019. He is an Assistant Professor with the School of Mathematics and Statistics, Beijing Key Laboratory on MCAACI, and Key Laboratory of Mathematical Theory and Computation in Information Security, Beijing Institute of Technology. His research interests include the design of experiments, statistical sketching and sampling methods for large-scale data, and applied statistics in solving scientific and engineering problems.