Optimal Transport for Generative Models

by Xianfeng Gu* and Shing-Tung Yau†

Abstract. Optimal transport plays a fundamental role in deep learning. Natural datasets have intrinsic patterns, which can be summarized as the manifold distribution principle: a natural class of data can be treated as a probability distribution on a low dimensional manifold, embedded in a high dimensional ambient space. A deep learning system mainly accomplishes two tasks: manifold learning and probability distribution transformation.

Given a manifold X, all the probability measures on X form an infinite dimensional manifold $\mathcal{P}(X)$. Optimal transport assigns a Riemannian metric on $\mathcal{P}(X)$, the so-called Wasserstein metric, and defines Otto's calculus, such that variational optimization can be carried out in $\mathcal{P}(X)$. A deep learning system learns the distribution by optimizing some functional in $\mathcal{P}$, therefore optimal transport lays down the theoretic foundation for deep learning.

This work introduces the theory of optimal transport and the profound relation between Brenier's theorem and Alexandrov's theorem in differential geometry via Monge-Ampère equation. We give a variational proof for Alexandrov's theorem, and convert the proof to a computational algorithm to solve the optimal transport map. The algorithm is based on computational geometry and can be generalized to general manifold setting.

Optimal transport theory and algorithms have been extensively applied in the models of Generative Adversarial Networks (GANs). In a GAN model, the generator computes the OT map, while the discriminator computes the Wasserstein distance between the generated data distribution and the real data distribution. The optimal transport theory shows the competition between the generator and the discriminator is completely unnecessary and should be replaced by collaboration. Furthermore, the regularity theory of optimal transport map explains the intrinsic reason for mode collapsing.

A novel generative model is introduced, which uses an autoencoder (AE) for manifold learning and OT map for probability distribution transformation. This AE-OT model improves the theoretical rigor and transparency, as well as the computational stability and efficiency; in particular, it eliminates the mode collapse.

* Stony Brook University, Stony Brook, NY, U.S.
E-mail: gu@cs.stonybrook.edu
† Harvard University, Cambridge, MA, U.S.
E-mail: yau@math.harvard.edu

1. Introduction

Deep learning is the mainstream technique for many machine learning tasks, including image recognition, machine translation, speech recognition, and so on. Despite its great success, the theoretical understanding on how it works remains primitive. Many fundamental open problems need to be solved, and many profound questions need to be answered.

In this chapter, we focuses on a geometric view of optimal transport (OT) to understand deep learning models, such as generative adversarial networks (GANs). Especially, we aim at answering the following basic questions:

1. What does a deep learning system really learn?

The system learns the probability distributions on manifolds. Each natural class of data set can be

treated as a point cloud in the high dimensional ambient space, and the point cloud approximates a special probability measure defined on a low dimensional manifold. The system learns two things: one is the manifold structure, the other is the distribution on the manifold. The manifold structure is represented by the encoding and decoding maps, which map between the manifold and the latent space. In generative models, such as GANs, the probability distributions are represented by the transport mappings from a predefined white noise (such as a Gaussian distribution, which can be easily generated from a uniform distribution) to the data distribution either in the latent space or on the data manifold.

2. How does a deep learning system really learn?

All the probability distributions on a manifold Σ form an infinite dimensional space $\mathcal{P}(\Sigma)$. A deep learning system performs optimization in the space of $\mathcal{P}(\Sigma)$. For example, the principle of maximum entropy searches for a distribution in $\mathcal{P}(\Sigma)$ by optimizing the entropy functional with some constraints obtained by observations. The optimal transport theory defines a Riemannian metric on the probability distribution space $\mathcal{P}(\Sigma)$ and Otto's calculus, such that the Wasserstein distance between measures can be computed explicitly, the variational optimizations can be carried out by these theoretic tools. For example, the discriminator in the WGAN model computes the Wasserstein distance between the real data distribution and the generated data distribution, the training process follows the Wasserstein gradient flow on $\mathcal{P}(\Sigma)$.

3. How well does a deep learning system really learn?

Current deep learning system designs have fundamental flaws, most generative models suffer from mode collapsing. Namely, they keep forgetting some knowledge already learned at the intermediate stage, or they generate unrealistic samples. This can be explained by the regularity theory of optimal transport maps, basically the transport maps are discontinuous, whereas the deep neural networks can only represent continuous maps, therefore either the map misses some connected components of the support of data distribution or covers all the components but also the gaps among them.

From the above short answers, we can see the importance of the theories of manifold and optimal transport for deep learning. In the following, we will briefly review the most related works in section 2; briefly introduce the theory of optimal transport in section 3; explain the computational algorithms for optimal transport in details in section 4; after the preparation, we explain the manifold distribution principle in deep learning and manifold learning by auto-encoder in section 5 and 6 respectively; then we use optimal transport view to analyze GAN model, explain the reason for mode collapse and the novel design to eliminate mode collapse in section 7; finally, we conclude the work in section 8.

2. Related Works

The literature of optimal transport and generative models is huge. Here, we only review the most directly related works.

2.1 Optimal Transportation Map

Monge-Kantorovich theory has been applied to solve optimal transportation problem via linear programming technique [30, 29]. The method was intuitively applied for image registration and warping in early research works. This approach was proposed in [55], however due to the expensive computational cost, the method can hardly handle the 3D image registration problem efficiently. Optimal transportation map was also applied for texture mapping purposes in [16], where the surface is intially mapped to the unit sphere conformally, then the mapping is optimized by a gradient flow with multiple level of resolutions to accelerate the convergence. Since the exact evaluation of Wasserstein distance is expensive, the heat kernel method was applied to approximate it in [50, 49]. In order to extend the problem into large data set, [12] added an entropic regularizer into the original Linear Programming problem and as a result, the regularized problem can be quickly computed with the Sinkhorn algorithm. Then Solomon et al. [49] improved the computational efficiency by the introduction of fast convolution.

Recent research works are more based on Monge-Brenier theory [9]. Gu et al. used a geometric variational approach to prove Alexandrov theorem in [20], which is equivalent to the discrete Brenier theorem. The method leads to a constructive algorithm for computing optimal transportation maps in general settings. In [15], De Goes et al. proposed to use OT for 2D shape reconstruction and simplification, later on they generalized to use capacity-constrained Voronoi tessellation to deal with blue noise processing problem [14]. [40] proposed a multi-scale approach to accelerate the computation for large scale problems. Most of the early works focus on 2D image registration and processing, recent works generalized them

to deal with 3D surfaces by using computational geometric approaches. By incorporating with conformal mapping methods, optimal transportation maps are applied to obtain area-preserving maps in [53]. The methods in [61] can simultaneously balance the area and the angle distortion. Su et al. generalized the algorithm to three dimensional case and presented a volume-preserving maps in [51] and then in [52] they further gave a volumetric controllable algorithm by OT map.

While most of the research works deal with optimal transport problems with Euclidean metric, [58, 11] focused on the solving the optimal transportation problems in the spherical domain. The method has also been applied for area-preserving brain mapping in [54], which maps the cortical surface onto the unit sphere conformally then onto the extended complex plane by the stereographic projection. The method has been improved in [44] by using conformal welding method.

Recent research works also introduce optimal transportation theory in optical design field. Reflector design problem was summarized as a group of Monge-Ampère equation problem in [57, 21, 58]. The correspondence between Monge-Ampère equations and reflector design problems was listed as one of the open problems in [60], and can further be related to optimal transportation theory. Similar researches in lens design situation were introduced in [23]. Numerical methods and simulation results of these optical design problems were proposed in [42].

2.2 Generative Models

Encoder-Decoder Architecture A breakthrough for image generating comes from the scheme of Variational Autoencoders (VAEs) (e.g. [31]), where the decoders approximate real data distributions from a Gaussian distribution in a variational approach (e.g. [31] and [47]). Latter Yuri Burda et al. [62] lower the requirement of latent distribution and propose the importance weighted autoencoder (IWAE) model through a different lower bound. Bin and David [13] propose that the latent distribution of VAE may not be Gaussian and improve it by firstly training the original model and then generating new latent code through the extended ancestral process. Another improvement of the VAE is the VQ-VAE model [1], which requires the encoder to output discrete latent codes by vector quantisation, then the posterior collapse of VAEs can be overcome. By multi-scale hierarchical organization, this idea is further used to generate high quality images in VQ-VAE-2 [46]. In [24], the authors adopt the Wasserstein distance in the latent space to measure the distance between the distribution of the latent code and the given one and generate images with better quality. Different from the VAEs, the AE-OT model [3] firstly embed the images into the latent space by autoencoder, then an extended semi-discrete OT map is computed to generate new latent code based on the fixed ones. Decoded by the decoder, new images can be generated. Although the encoder-decoder based methods are relatively simple to train, the generated images tend to be blurry.

Generative Adversarial Networks The GAN model [19] tries to alternatively update the generator, which maps the noise sampled from a given distribution to real images, and the discriminator differentiates the difference between the generated images and the real ones. If the generated images successfully fool the discriminator, we say the model is well trained. Later, [45] proposes a deep convolutional neural network (DCGAN) to generate images with better quality. While being a powerful tool in generating realistic samples, GANs can be hard to train and suffer from mode collapse problem [18]. After delicate analysis, [6] points out that it is the KL divergence the original GAN used causes these problems. Then the authors introduce the celebrated WGAN, which makes the whole framework easy to converge. To satisfy the Lipschitz continuity required by WGAN, a lot of methods are proposed, including clipping [6], gradient penalty [22], spectral normalization [43] and so on. Later, Wu et al. [28] use the Wasserstein divergence objective, which get rid of the Lipschitz approximation problem and gets a better result. Instead L_1 cost adopted by WGAN, Liu et al. [37] propose the WGAN-QC by taking the L_2 cost into consideration. Though various GANs can generate sharp images, they will theoretically encounter the mode collapse or mode mixture problem [18, 3].

Hybrid Models To solve the blurry image problem of encoder-decoder architecture and the mode collapse/mixture problems of GANs, a natural idea is to compose them together. Larsen et al. [32] propose to combine the variational autoencoder with a generative adversarial network, and thus generate images better than VAEs. [39] matches the aggregated posterior of the hidden code vector of the autoencoder with an arbitrary prior distribution by a discriminator and then applies the model into tasks like semi-supervised classification and dimensionality reduction. BiGAN [27] uses the discriminator to differentiate both the generated images and the generated latent code. Further, by utilizing the BigGAN generator [4], the BigBiGAN [17] extends this method to generate much better results. Here we also treat the BourGAN [59] as a hybrid model, because it firstly embeds the images into latent space by Bourgain theorem, then trains the GAN model by sampling from the latent space using the GMM model.

Conditional GANs are another kind of hybrid models that can also be treated as image-to-image transformation. For example, using an encoder-decoder architecture to build the connection between paired images and then differentiating the decoded images with the real ones by a discriminator, [25] is able to transform images of different styles. Further, SRGAN [33] uses similar architecture to get super resolution images from their low resolution versions. The SRGAN model utilizes the content loss and adversarial loss. It uses the paired data and the visually meaningful features used by SRGAN are extracted from the pre-trained VGG19 network [48], which makes it not so reasonable under the scenes where the datasets are not included in those used to train the VGG.

Optimal Transport Based Generative Model In [35] Lei et al. first gave a geometric interpretation to the Generative Adversarial Networks (GANs). By using the optimal transportation view of GAN model, they showed that the discriminator computes the Wasserstein distance via the Kantorovich potential, the generator calculates the transportation map. For a large class of transportation costs, the Kantorovich potential can give the optimal transportation map by a close-form formula. Therefore, it is sufficient to solely optimize the discriminator. This shows the adversarial competition can be avoided, and the computational architecture can be simplified. In [34] the authors pointed out that GANs mainly accomplish two tasks: manifold learning and probability distribution transformation. The latter can be carried out using the classical OT method. Then in [3] a new generative model based on extended semi-discrete optimal transport was proposed, which avoids representing discontinuous maps by DNNs, therefore effectively prevents mode collapse and mode mixture.

Numerical Method In this work, we show that the reason that causes the mode collapse in deep learning is indeed the discontinuity of optimal transport map in general. It is very similar to the situation when using the classic numerical method to solve OT map. For instance, the Brenier potential in OT satisfies the Hamiltonian- Jacobi equation which could be continuous. However, its velocity (corresponding to the OT map) satisfying the conservation law is generally discontinuous. For examples, the Benamou-Brenier method [7] and Haker-Tannenbaum-Angent method [5] compute the optimal transport maps based on fluid dynamics.

3. Optimal Transport Theory

In this subsection, we will introduce basic concepts and theorems in classic optimal transport theory, focusing on Brenier's approach, and their generalization to the discrete setting. Details can be found in Villani's book [56].

3.1 Monge Problem

Suppose $X \subset \mathbb{R}^d$, $Y \subset \mathbb{R}^d$ are two measurable subsets of d-dimensional Euclidean space $\mathbb{R}^d$, μ, ν are two probability measures defined on X and Y respectively, with density functions

$$\mu(x) = f(x)dx, \quad \nu(y) = g(y)dy.$$

Suppose their total measures are equal, $\mu(X) = \nu(Y)$, namely

$$\int_X f(x)dx = \int_Y g(y)dy. \tag{1}$$

We only consider maps which preserve the measure.

Definition 1 (Measure-Preserving Map)**.** *A map $T : X \to Y$ is* measure preserving *if for any measurable set $B \subset Y$, the set $T^{-1}(B)$ is μ-measurable and $\mu(T^{-1}(B)) = \nu(B)$, i.e.*

$$\int_{T^{-1}(B)} f(x)dx = \int_B g(y)dy. \tag{2}$$

Measure-preserving condition is denoted as $T_\#\mu = \nu$, where $T_\#\mu$ is the push forward measure induced by T. Suppose $T : X \to Y$ is differentiable, $T \in C^1(X)$, then the measure-preserving map satisfies the Jacobian equation:

$$\det DT(x) = \frac{f(x)}{g \circ T(x)}. \tag{3}$$

Definition 2 (Transport Cost)**.** *Given a* cost function $c(x,y) : X \times Y \to \mathbb{R}_{\geq 0}$*, which indicates the cost of moving each unit mass from the source to the target, the total* transport cost *of the map $T : X \to Y$ is defined to be*

$$\mathcal{C}(T) := \int_X c(x, T(x))d\mu(x). \tag{4}$$

The Monge's problem of optimal transport arises from finding the measure-preserving map that minimizes the total transport cost.

Problem 1 (Monge's Optimal Transport [8] (MP))**.** *Given a transport cost function $c : X \times Y \to \mathbb{R}$, find the measure preserving map $T : X \to Y$ that minimizes the total transport cost*

$$(MP) \qquad \min_{T_\#\mu = \nu} \int_X c(x, T(x))d\mu(x). \tag{5}$$

Definition 3 (Optimal Transport Map)**.** *The solutions to the Monge's problem is called the* optimal transport map*, whose total transport cost defines the* Wasserstein distance *between μ and ν.*

If $c(x,y)=\frac{1}{2}\|x-y\|^2$, the Wasserstein distance is denoted as $\mathcal{W}_2(\mu,\nu)$, then

$$\mathcal{W}_2^2(\mu,\nu)=\min_{T_\#\mu=\nu}\frac{1}{2}\int_X|x-T(x)|^2d\mu(x). \tag{6}$$

3.2 Kantorovich's Approach

Depending on the cost function and the measures, the optimal transport map between (X,μ) and (Y,ν) may not exist. For example, suppose μ is atomic $\mu=\delta(x-x_0)$, and $\nu=\sum_{i=1}^k\nu_i\delta(y-y_i)$ with $\sum_{i=1}^k\nu_i=1$, then the mass concentrated on x_0 has to be split and sent to different y_i's. Kantorovich relaxed transport maps to *transport plans* or *transport schemes*. A transport plan is represented by a joint probability measure $\rho:X\times Y\to\mathbb{R}_{\geq 0}$, such that the marginal probability of ρ equals to μ and ν respectively. Formally, let the projection maps be $\pi_x(x,y)=x$, $\pi_y(x,y)=y$, then define joint measure class

$$\Pi(\mu,\nu):=\{\rho:X\times Y\to\mathbb{R}:(\pi_x)_\#\rho=\mu,(\pi_y)_\#\rho=\nu\} \tag{7}$$

Problem 2 (Kantorovich). *Given a transport cost function $c:X\times Y\to\mathbb{R}$, find the joint probability measure $\rho:X\to Y$ with marginals μ and ν that minimizes the total transport cost*

$$(KP)\qquad \min_{\rho\in\Pi(\mu,\nu)}\int_{X\times Y}c(x,y)d\rho(x,y). \tag{8}$$

Kantorovich's problem can be solved using linear programming method. Due to the duality of lineary programming, the (KP) Eqn. 8 can be reformulated as the duality problem (DP) as follows:

Problem 3 (Kantorovich Dual). *Given a transport cost function $c:X\times Y\to\mathbb{R}$, find the function $\varphi\in L^1(X)$ and $\psi\in L^1(Y)$, such that*

$$(DP)\quad \max_{\varphi,\psi}\left\{\int_X\varphi(x)d\mu+\int_Y\psi(y)d\nu\ :\ \varphi(x)+\psi(y)\leq c(x,y)\right\} \tag{9}$$

The maximum value of Eqn. 9 gives the Wasserstein distance. Most existing Wasserstein GAN models are based on the duality formulation under the L^1 cost function.

Definition 4 (c-transform). *The c-transform of $\varphi:X\to\mathbb{R}$ is defined as $\varphi^c:Y\to\mathbb{R}$:*

$$\varphi^c(y)=\inf_{x\in X}(c(x,y)-\varphi(x)). \tag{10}$$

Assume $c(x,y)$ and φ are with C^1 continuity, then the necessary condition for c-transform is given by

$$\nabla_x c(x,y(x))-\nabla\varphi(x)=0. \tag{11}$$

Then the Kantorovich dual problem can be rewritten as

$$(DP)\qquad \mathcal{W}_c(\mu,\nu)=\max_\varphi\int_X\varphi(x)d\mu+\int_Y\varphi^c(y)d\nu, \tag{12}$$

where φ is called the *Kantorovich's potential*.

3.3 Brenier's Approach

Given a strictly C^1 convex function $h:\Omega\to\mathbb{R}$, where Ω is a convex domain in $\mathbb{R}^n$, the gradient mapping $x\mapsto\nabla h(x)$ is invertible. The inverse mapping is denoted as $(\nabla h)^{-1}$.

Suppose the cost function $c(x,y)=h(x-y)$ where h is a strictly C^1 convex function, then the solution to Kantorovich's dual problem Eqn. 12 satisfies the c-transform condition Eqn. 11, hence we obtain the formula for the optimal transport map T,

$$T(x)=x-(\nabla h)^{-1}(\nabla\varphi(x)). \tag{13}$$

This leads to the following theorem,

Theorem 1 (Villani [56]). *Given μ and ν on a compact domain $\Omega\subset\mathbb{R}^n$ there exists an optimal transport plan ρ for the cost $c(x,y)=h(x-y)$ with h strictly convex. It is unique and of the form $(id,T_\#)\mu$, provided μ is absolutely continuous and $\partial\Omega$ is negligible. More over, there exists a Kantorovich potential φ, and T can be represented as*

$$T(x)=x-(\nabla h)^{-1}(\nabla\varphi(x)).$$

For quadratic Euclidean distance cost, $h(x)=\frac{1}{2}\langle x,x\rangle$, $(\nabla h)^{-1}(x)=x$, then Eqn. 13 becomes

$$T(x)=x-\nabla\varphi(x)=\nabla\left(\frac{1}{2}\langle x,x\rangle-\varphi(x)\right)=\nabla u, \tag{14}$$

where the function $u:X\to\mathbb{R}$ is called the *Brenier potential*. In this case, the Brenier's potential u and the Kantorovich's potential φ is related by Eqn. 14. Assume the Brenier potential is C^2 convex, by Jacobian equation Eqn. 3, it satisfies the following *Monge-Ampère equation*:

$$\det\left(\frac{\partial^2u}{\partial x_i\partial x_j}\right)(x)=\frac{f(x)}{g\circ\nabla u(x)} \tag{15}$$

The existence, uniqueness and the intrinsic structure of the optimal transport map were proven by Brenier [9].

Theorem 2 (Brenier [9]). *Suppose X and Y are measurable subsets of the Euclidean space $\mathbb{R}^d$ and the transport cost is the quadratic Euclidean distance $c(x,y)=1/2\|x-y\|^2$. Furthermore μ is absolutely continuous with respect to Lebesgue measure and μ and ν have finite second order moments,*

$$\int_X|x|^2d\mu(x)+\int_Y|y|^2d\nu(y)<\infty, \tag{16}$$

then there exists a convex function $u : X \to \mathbb{R}$, the so-called Brenier potential, its gradient map ∇u gives the solution to the Monge's problem,

$$(\nabla u)_{\#}\mu = \nu. \tag{17}$$

The Brenier potential is unique upto a constant, hence the optimal mass transport map is unique.

Therefore, finding the optimal transport map is reduced to solving the Monge-Ampère equation.

Problem 4 (Brenier). *Suppose X and Y are subsets of the Euclidean space $\mathbb{R}^d$ and the transport cost is the quadratic Euclidean distance. Furthermore μ is absolutely continuous with respect to Lebesgue measure and μ and ν have finite second order moments, find a convex function $u : X \to \mathbb{R}$, satisfies the Monge-Ampère equation Eqn. 15.*

For quadratic Euclidean distance cost $c(x,y) = 1/2\|x-y\|^2$ in $\mathbb{R}^n$, the c-transform and the classical Legendre transform have special relations.

Definition 5 (Legendre Transform). *Given a function $\varphi : \mathbb{R}^n \to \mathbb{R}$, its Legendre transform is defined as*

$$\varphi^*(y) := \sup_x \left(\langle x,y\rangle - \varphi(x)\right). \tag{18}$$

We can show the following relation holds for quadratic Euclidean cost,

$$\frac{1}{2}|y|^2 - \varphi^c(y) = \left(\frac{1}{2}|x|^2 - \varphi(x)\right)^*. \tag{19}$$

3.4 McCann's Displacement

We consider all the probability measures μ defined on X with finite second order moment, μ is absolutely continuous with respect to Lebesgue measure,

$$\mathcal{P}(X) := \left\{\mu : \int_X |x|^2 d\mu(x) < \infty, \mu \text{ a.c.}\right\} \tag{20}$$

Then according to Brenier's theorem, for any pair $\mu,\nu \in \mathcal{P}(X)$, there exists a unique optimal transport map $T : X \to X$, $T_{\#}\mu = \nu$, furthermore $T = \nabla u$ for some Brenier potential u, which satisfies the Monge-Ampère equation 15. The transportation cost gives the Wasserstein distance between μ and ν in Eqn. 6.

Definition 6. *Given a path $\rho : [0,1] \to \mathcal{P}(X)$ in the $(\mathcal{P}(X),\mathcal{W}_2)$, if it satisfies the condition*

$$\mathcal{W}_2(\rho(s),\rho(t)) = |t-s|\mathcal{W}_2(\rho(0),\rho(1)) \quad \forall s,t \in [0,1], \tag{21}$$

then we say ρ is a geodesic.

McCann gives the geodesic formula in the distance space $(\mathcal{P}(X),\mathcal{W}_2)$.

Theorem 3 (McCann). *Given $\mu,\nu \in (\mathcal{P}(X),\mathcal{W}_2)$ and u is the corresponding Brenier potential, then the geodesic connecting μ and ν is given by*

$$\rho(t) := ((1-t)Id + t\nabla u)_{\#}\mu \quad t \in [0,1],$$

which is called McCann's displacement.

3.5 Benamou-Brenier Dynamic Fluid

Brenier-Benamou gives another formulation of geodesic using fluid dynamic. Let $X = \mathbb{R}^n$, consider a flow field in X, represented by the density field $\rho(t,x)$ and the flow velocity field $\mathbf{v}(t,x)$. We denote $\rho(t,\cdot)$ as ρ_t, $\mathbf{v}(t,\cdot)$ as $\mathbf{v}_t$. We define $\Sigma(\mu,\nu)$ as set of flows $(\rho,\mathbf{v}) = (\rho_t,\mathbf{v}_t)$, $0 \le t \le 1$, satisfying the following conditions:

1. ρ_t is continuous with respect to t and $\rho_t(x)$ is absolutely continuous with respect to the Lebesgue measure in X;
2. $\mathbf{v}(t,x)$ is L^2 integrable with respect tot he measure $d\rho_t(x)dt$,
$$\int_0^1 \int_X |\mathbf{v}(t,x)|^2 d\rho_t(x)dt < \infty.$$
3. The union of the support of ρ_t is bounded,
$$\bigcup_{0\le t\le 1} \text{Supp}(\rho_t) \text{ bounded.}$$
4. By mass conservation law, the pair $(\rho,\mathbf{v})$ satisfy the *continuity equation*,
$$\frac{\partial \rho_t}{\partial t} + \nabla\cdot(\rho_t\mathbf{v}_t) = 0 \tag{22}$$
in the distributional sense.
5. Furthermore, the flow satisfies the boundary condition $\rho_0 = \mu$ and $\rho_1 = \nu$.

Problem 5 (Benamous-Brenier). *Find the flow $(\rho,\mathbf{v}) \in \Sigma(\mu,\nu)$ that minimizes the total kinetic energy,*

$$A[\rho,\mathbf{v}] = \int_0^1 \left(\int_X \rho_t(x)|\mathbf{v}_t(x)|^2 dx\right) dt. \tag{23}$$

Benamou-Brenier proves kinetic energy of the solution to Eqn. 23 equals to the square of Wasserstein distance in Eqn. 6, namely Benamou-Brenier problem is equivalent to Brenier problem, furthermore the geodesic is given by the solution to the Benamou-Brenier problem,

$$\min\left\{\frac{1}{2}\int_0^1 \int_X |\mathbf{v}(x,t)|^2 d\rho(x,t)dt : (\rho_t,\mathbf{v}_t) \in \Sigma(\mu,\nu)\right\}.$$

3.6 Otto's Calculus

Suppose $\mathbf{v}$ is the optimal flow, given any divergence field $\nabla\cdot\mathbf{w}=0$,

$$-\nabla\cdot\rho\left(\mathbf{v}+\varepsilon\frac{\mathbf{w}}{\rho}\right)=-\nabla\cdot\rho\mathbf{v}=\frac{\partial\rho}{\partial t},$$

therefore $\mathbf{v}+\varepsilon\mathbf{w}/\rho\in\Sigma(\mu,\nu)$. By the optimality of $\mathbf{v}$, we have

$$\int\rho|\mathbf{v}|^2\le\int\rho\left|\mathbf{v}+\varepsilon\frac{\mathbf{w}}{\rho}\right|^2,$$

therefore we have

$$\int\langle\mathbf{v},\mathbf{w}\rangle=0.$$

Because $\mathbf{w}$ is an arbitrary divergence free vector field, by Hodge decomposition theorem, we have $\mathbf{v}$ is the gradient field of some function φ, $\mathbf{v}=\nabla\varphi$. Benamou-Brenier problem is reduced to

$$\mathcal{W}_2^2(\mu,\nu)=\min_{(\rho_t,u)}\left\{\int_0^1\int_X|\nabla u|^2d\rho_t dt,\rho_0=\mu,\rho_1=\nu,\right.$$
$$\left.-\nabla\cdot(\rho_t\nabla u)=\frac{\partial\rho_t}{\partial t}\right\}.$$

Given two geodesics $\rho_1(t),\rho_2(t)\subset\mathcal{P}(X)$, $\rho_1(0)=\rho_2(0)=\rho$, their tangent vectors at $\rho\in\mathcal{P}(X)$ are

$$\frac{\partial\rho_1}{\partial t}=-\nabla\cdot(\rho_1\nabla\varphi_1),\quad\frac{\partial\rho_2}{\partial t}=-\nabla\cdot(\rho_2\nabla\varphi_2),$$

the Riemannian metric is defined as

$$\left\langle\frac{\partial\rho_1}{\partial t},\frac{\partial\rho_2}{\partial t}\right\rangle_\rho=\int_X\langle\nabla\varphi_1,\nabla\varphi_2\rangle\rho(x)dx.$$

Otto's calculus provides a theoretic tool for optimization in $(\mathcal{P}(X),\mathcal{W}_2)$. For example, we can show the Wasserstein gradient flow of entropy is equivalent to the classical heat flow. Given a domain $X\subset\mathbb{R}^d$ with smooth boundary ∂X, and a measure $\rho\in\mathcal{P}(X)$, its entropy is defined as

$$\mathrm{Ent}(\rho):=\int_X\rho\log\rho\ dx.$$

Given a path $\rho(t)\subset\mathcal{P}(X)$,

$$\frac{d}{dt}\mathrm{Ent}(\rho(t))=\int_X\left(\dot\rho\log\rho+\rho\frac{\dot\rho}{\rho}\right)\ dx=\int_X(1+\log\rho)\dot\rho\ dx.$$

By continuity equation $\dot\rho=-\nabla\cdot(\mathbf{v}\rho)$,

$$\int_X\dot\rho\ dx=-\int_X\nabla\cdot(\mathbf{v}\rho)\ dx=-\int_{\partial X}\mathbf{v}\rho\ dx=0.$$

and

$$\nabla\cdot(\rho\log\rho\mathbf{v})=\log\rho\nabla(\rho\mathbf{v})+\langle\nabla\log\rho,\rho\mathbf{v}\rangle.$$

we obtain

$$\frac{d}{dt}\mathrm{Ent}(\rho(t))=\int_X\langle\nabla\log\rho,\mathbf{v}\rangle\rho\ dx$$

This shows the Wasserstein gradient of entropy equals to $\nabla\log\rho$. We plug it into the continuity equation and obtain

$$\frac{\partial\rho_t}{\partial t}+\nabla\cdot\left(-\frac{\nabla\rho_t}{\rho_t}\rho_t\right)=\frac{\partial\rho_t}{\partial t}-\Delta\rho_t=0.$$

This shows that the Wasserstein gradient flow of the entropy is equivalent to the classical heat flow.

3.7 Regularity of Optimal Transportation Maps

Let Ω and Λ be two bounded smooth open sets in $\mathbb{R}^d$, let $\mu=f dx$ and $\nu=g dy$ be two probability measures on $\mathbb{R}^d$ such that $f|_{\mathbb{R}^d\setminus\Omega}=0$ and $g|_{\mathbb{R}^d\setminus\Lambda}=0$. Assume that f and g are bounded away from zero and infinity on Ω and Λ, respectively.

3.7.1 Convex Target Domain

Definition 7 (Hölder continuous). *A real or complex-valued function f on d-dimensional Euclidean space satisfies a Hölder condition, or is Hölder continuous, when there are non-negative real constants C, $\alpha>1$, such that*

$$|f(x)-f(y)|\le C|x-y|^\alpha$$

for all x and y in the domain of f.

Definition 8 (Hölder Space). *The Hölder space $C^{k,\alpha}(\Omega)$, where Ω is an open subset of some Euclidean space and $k\ge0$ an integer, consists of those functions on Ω having continuous derivatives up to order k and such that the k-th partial derivatives are Hölder continuous with exponent α, where $0<\alpha\le1$.*

Consider the optimal transport map $\nabla u:(\Omega,f(x)dx)\to(\Lambda,g(y)dy)$, the following theorems give the regularity of the Brenier potential u. Caffarelli's theorem addresses the cases with the cost function $c(x,y)=1/2|x-y|^2$.

Theorem 4 (Caffarelli [10]). *If Λ is convex, then the Brenier potential u is strictly convex, furthermore*

1. *If $\lambda\le f$, $g\le1/\lambda$ for some $\lambda>0$, then $u\in C^{1,\alpha}_{loc}(\Omega)$.*
2. *If $f\in C^{k,\alpha}_{loc}(\Omega)$ and $g\in C^{k,\alpha}_{loc}(\Lambda)$, with $f,g>0$, then $u\in C^{k+2,\alpha}_{loc}(\Omega)$, $(k\ge0,\alpha\in(0,1))$*

Ma-Trudinger-Wang's theorem [38] handles general cost functions $c(x,y)$. In the following theorem,

$$c_{p,q}:=\frac{\partial^2c(x,y)}{\partial x_p\partial y_q},c_{ij,p}:=\frac{\partial^3c(x,y)}{\partial x_i\partial x_j\partial y_p},c_{ij,pq}:=\frac{\partial^4c(x,y)}{\partial x_i\partial x_j\partial y_p\partial y_q},$$

and $(c^{p,q})$ is the inverse matrix of $c_{p,q}$.

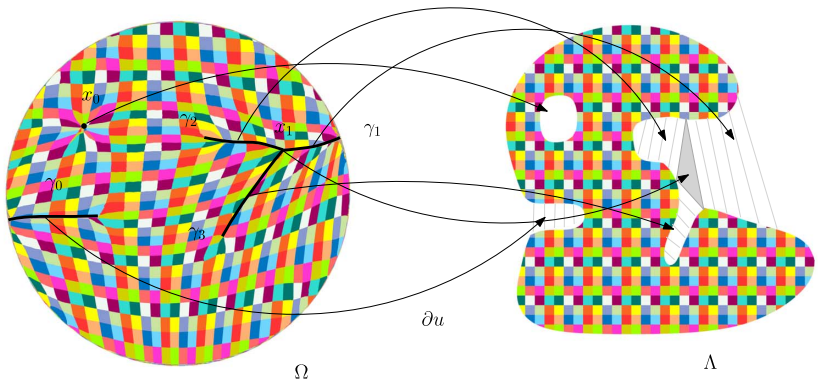


Figure 1. Singularity structure of an optimal transportation map [63].

Theorem 5 (Ma-Trudinger-Wang). *The potential function u is C^3 smooth if the cost function c is smooth, f,g are positive, $f \in C^2(\Omega)$, $g \in C^2(\Lambda)$, and*

- *A1* $\forall x,\xi \in \mathbb{R}^n$, $\exists! y \in \mathbb{R}^n$, *s.t.* $\xi = D_x c(x,y)$ *(for existence)*
- *A2* $|D^2_{xy} c| \neq 0$.
- *A3* $\exists c_0 > 0$ *s.t.* $\forall \xi,\eta \in \mathbb{R}^n$, $\xi \perp \eta$

$$\sum (c_{ij,rs} - c^{p,q} c_{ij,p} c_{q,rs}) c^{r,k} c^{s,l} \xi_i \xi_j \eta_k \eta_l \geq c_0 |\xi|^2 |\eta|^2.$$

- *B1* Λ *is c-convex w.r.t.* Ω, *namely* $\forall x_0 \in \Omega$,

$$\Lambda_{x_0} := D_x c(x_0, \Lambda)$$

is convex.

3.7.2 Non-Convex Target Domain

If Λ is not convex, there exist f and g smooth such that $u \notin C^1(\Omega)$, the optimal transportation map ∇u is discontinuous at singularities.

Definition 9 (subgradient). *Given an open set $\Omega \subset \mathbb{R}^d$ and $u : \Omega \to \mathbb{R}$ a convex function, for $x \in \Omega$, the subgradient (subdifferential) of u at x is defined as*

$$\partial u(x) := \{p \in \mathbb{R}^n : u(z) \geq u(x) + \langle p, z - x\rangle \quad \forall z \in \Omega\}.$$

It is obvious that $\partial u(x)$ is a closed convex set. Geometrically, if $p \in \partial u(x)$, then the hyper-plane

$$l_{x,p}(z) := u(x) + \langle p, z - x\rangle$$

touches u from below at x, namely $l_{x,p} \leq u$ in Ω and $l_{x,p}(x) = u(x)$, $l_{x,p}$ is a supporting plane to u at x.

The Brenier potential u is differentiable at x if its subgradient $\partial u(x)$ is a singleton. We classify the points according to the dimensions of their subgradients, and define the sets

$$\Sigma_k(u) := \left\{x \in \mathbb{R}^d | \ \dim(\partial u(x)) = k\right\}, \quad k = 0,1,2\ldots,d.$$

It is obvious that $\Sigma_0(u)$ is the set of regular points, $\Sigma_k(u)$, $k > 0$ are the set of singular points. We also define the *reachable subgradients* at x as

$$\nabla_* u(x) := \left\{\lim_{k\to\infty} \nabla u(x_k) | x_k \in \Sigma_0, x_k \to x\right\}.$$

It is well known that the subgradient equals to the convex hull of the reachable subgradient,

$$\partial u(x) = \text{Convex Hull}(\nabla_* u(x)).$$

Theorem 6 (Figalli [63]). *] Let $\Omega,\Lambda \subset \mathbb{R}^d$ be two bounded open sets, let $f,g : \mathbb{R}^d \to \mathbb{R}^+$ be two probability densities, that are zero outside Ω, Λ and are bounded away from zero and infinity on Ω, Λ, respectively. Denote by $T = \nabla u : \Omega \to \Lambda$ the optimal transport map provided by theorem 2. Then there exist two relatively closed sets $\Sigma_\Omega \subset \Omega$ and $\Sigma_\Lambda \subset \Lambda$ with $|\Sigma_\Omega| = |\Sigma_\Lambda| = 0$ such that $T : \Omega \setminus \Sigma_\Omega \to \Lambda \setminus \Sigma_\Lambda$ is a homeomorphism of class $C^{0,\alpha}_{loc}$ for some $\alpha > 0$.*

We call Σ_Ω as singular set of the optimal transportation map $\nabla u : \Omega \to \Lambda$. Fig. 1 illustrates the singularity set structure, computed using the algorithm based on theorem 8. We obtain

$$\Sigma_0 = \Omega \setminus \{\Sigma_1 \cup \Sigma_2\}, \quad \Sigma_1 = \bigcup_{k=0}^{3} \gamma_k, \quad \Sigma_2 = \{x_0, x_1\}.$$

The subgradient of x_0, $\partial u(x_0)$ is the entire inner hole of Λ, $\partial u(x_1)$ is the shaded triangle. For each point on $\gamma_k(t)$, $\partial u(\gamma_k(t))$ is a line segment outside Λ. x_1 is the bifurcation point of γ_1,γ_2 and γ_3. The Brenier potential on Σ_1 and Σ_2 is not differentiable, the optimal transportation map ∇u on them are discontinuous.

Fig. 2 shows the singularity structure of an optimal transport map between the uniform distribution inside a solid ball to that of the solid Stanford bunny. Since the target domain is non-convex, the boundary surface has complicated folding structure, which is the singularity of the map.

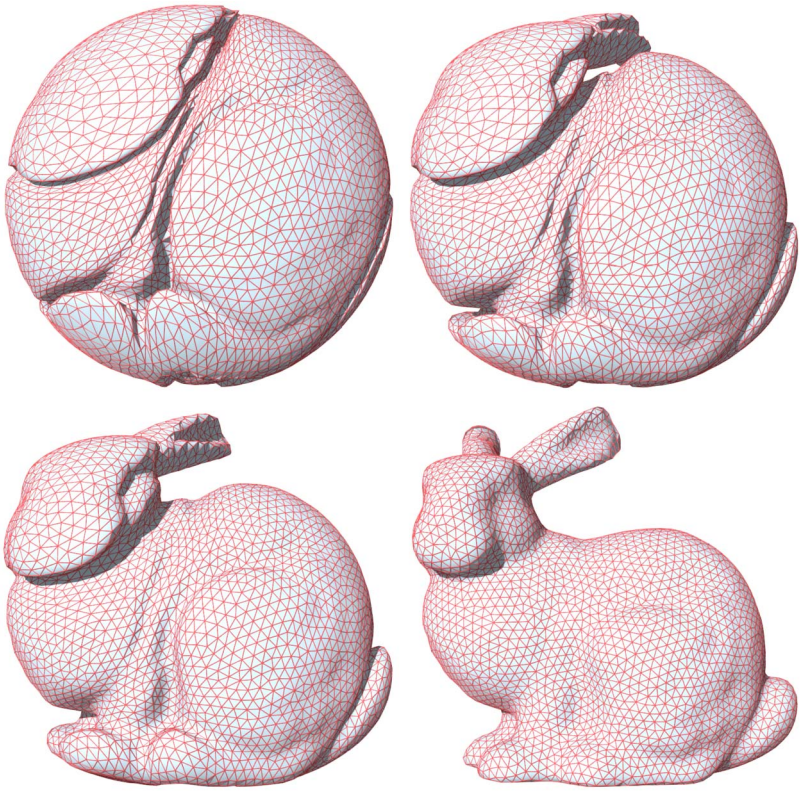


Figure 2. Singularity structure of an optimal transportation map.

4. Computational Algorithm

4.1 Semi-Discrete Optimal Transport Map

Brenier's theorem can be directly generalized to the discrete situation. The source measure μ is absolutely continuous with respect to Lebesgue measure, defined on a convex compact domain Ω; the target measure ν is the summation of Dirac measures

$$\nu = \sum_{i=1}^{n} \nu_i \delta(y - y_i), \tag{24}$$

where $Y = \{y_1, y_2, \cdots, y_n\}$ are training samples. The source and the target measures have equal total mass $\sum_{i=1}^{n} \nu_i = \mu(\Omega)$. Each sample y_i corresponds to a supporting plane of the Brenier potential, denoted as

$$\pi_{h,i}(x) := \langle x, y_i \rangle + h_i, \tag{25}$$

where the height h_i is an unknown variable. We represent all the height variables as $h = (h_1, h_2, \cdots, h_n)$.

An *envelope* of a family of hyper-planes in the Euclidean space is a hyper-surface that is tangent to each member of the family at some point, and these points of tangency together form the whole envelope. As shown in Fig. 3, the Brenier potential $u_h : \Omega \to \mathbb{R}$ is a piecewise linear convex function determined by h, which is the upper envelope of all its supporting planes,

$$u_h(x) = \max_{i=1}^{n} \{\pi_{h,i}(x)\} = \max_{i=1}^{n} \{\langle x, y_i \rangle + h_i\}. \tag{26}$$

The graph of Brenier potential is a convex polytope. Each supporting plane $\pi_{h,i}$ corresponds to a facet of the polytope. The projection of the polytope induces a cell decomposition of Ω, each supporting plane $\pi_i(x)$ projects onto a cell $W_i(h)$,

$$\Omega = \bigcup_{i=1}^{n} W_i(h) \cap \Omega, \quad W_i(h) := \{p \in \mathbb{R}^d | \nabla u_h(p) = y_i\}. \tag{27}$$

the cell decomposition is a *power diagram*.

The μ-measure of $W_i \cap \Omega$ is denoted as $w_i(h)$,

$$w_i(h) := \mu(W_i(h) \cap \Omega) = \int_{W_i(h) \cap \Omega} d\mu. \tag{28}$$

The gradient map $\nabla u_h : \Omega \to Y$ maps each cell $W_i(h)$ to a single point y_i,

$$\nabla u_h : W_i(h) \mapsto y_i, i = 1, 2, \ldots, n. \tag{29}$$

Given the target measure ν in Eqn. 24, there exists a discrete Brenier potential in Eqn. 26, whose projected μ-volume of each facet $w_i(h)$ equals to the given target measure ν_i. This was proved by Alexandrov in convex geometry.

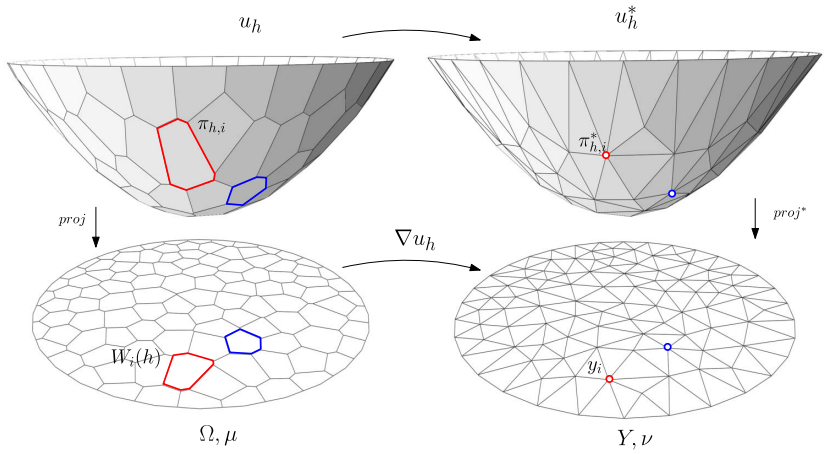


Figure 3. PL Brenier potential (left) and its Legendre dual (right).

Theorem 7 (Alexandrov [2]). *Suppose Ω is a compact convex polytope with non-empty interior in $\mathbb{R}^n$, $n_1,...,n_k \subset \mathbb{R}^{n+1}$ are distinct k unit vectors, the $(n+1)$-th coordinates are negative, and $v_1,...,v_k > 0$ so that $\sum_{i=1}^k v_i = vol(\Omega)$. Then there exists a convex polytope $P \subset \mathbb{R}^{n+1}$ with exact k codimension-1 faces $F_1,\dots,F_k$, so that n_i is the normal vector to F_i and the intersection between Ω and the projection of F_i is with volume v_i. Furthermore, such P is unique up to vertical translation.*

Alexandrov's proof for the existence is based on algebraic topology, which is not constructive. Recently, Gu et al. [20] gave a constructive proof based on the variational approach.

Theorem 8 (Gu-Luo-Yau [20]). *Let μ a probability measure defined on a compact convex domain Ω in $\mathbb{R}^d$, $Y = \{y_1, y_2, \dots, y_n\}$ be a set of distinct points in $\mathbb{R}^d$. Then for any $v_1, v_2, \dots, v_n > 0$ with $\sum_{i=1}^n v_i = \mu(\Omega)$, there exists $h = (h_1, h_2, \dots, h_n) \in \mathbb{R}^n$, unique up to adding a constant $(c, c, \dots, c)$, so that $w_i(h) = v_i$, for all i. The vector h is the unique minimum argument of the following convex energy*

$$E(h) = \int_0^h \sum_{i=1}^n w_i(\eta) d\eta_i - \sum_{i=1}^n h_i v_i, \tag{30}$$

defined on an open convex set

$$\mathcal{H} = \{h \in \mathbb{R}^n : w_i(h) > 0, i = 1, 2, \dots, n\}. \tag{31}$$

Furthermore, ∇u_h minimizes the quadratic cost

$$\frac{1}{2} \int_\Omega |x - T(x)|^2 d\mu(x) \tag{32}$$

among all transport maps $T_\# \mu = \nu$, where the Dirac measure $\nu = \sum_{i=1}^n v_i \delta(y - y_i)$.

The gradient of the above convex energy in Eqn. 30 is given by:

$$\nabla E(h) = (w_1(h) - v_1, w_2(h) - v_2, \dots, w_n(h) - v_n)^T \tag{33}$$

The Hessian of the energy is given by

$$\frac{\partial w_i}{\partial h_j} = -\frac{\mu(W_i \cap W_j \cap \Omega)}{|y_i - y_j|}, \quad \frac{\partial w_i}{\partial h_i} = \sum_{j \neq i} \frac{\partial w_i}{\partial h_j} \tag{34}$$

As shown in Fig. 3, the Hessian matrix has explicit geometric interpretation. The left frame shows the discrete Brenier potential u_h, the right frame shows its Legendre transformation u_h^* using definition 18. The Legendre transformation can be constructed geometrically: for each supporting plane $\pi_{h,i}$, we construct the dual point $\pi^*_{h,i} = (y_i, -h_i)$, the convex hull of the dual points $\{\pi^*_{h,1}, \pi^*_{h,2}, \dots, \pi^*_{h,n}\}$ is the graph of the Legendre transformation u_h^*. The projection of u_h^* induces a triangulation of $Y = \{y_1, y_2, \dots, y_n\}$, which is the *weighted Delaunay triangulation*. As shown in Fig. 4, the power diagram in Eqn. 27 and weighted Delaunay triangulation are Poincarè dual to each other: if in the power diagram, $W_i(h)$ and $W_j(h)$ intersect at a $(d-1)$-dimensional cell, then in the weighted Delaunay triangulation y_i connects with y_j. The element of the Hessian matrix Eqn. 34 is the ratio between the μ-volume of the $(d-1)$ cell in the power diagram and the length of dual edge in the weighted Delaunay triangulation.

The conventional power diagram can be closely related to the above theorem.

Definition 10 (power distance). *Given a point $y_i \in \mathbb{R}^d$ with a power weight ψ_i, the power distance is given by*

$$pow(x, y_i) = |x - y_i|^2 - \psi_i. \tag{35}$$

Definition 11 (power diagram). *Given weighted points $(y_1, \psi_1), \dots, (y_k, \psi_k)$, the power diagram is the cell decomposition of $\mathbb{R}^d$,*

$$\mathbb{R}^d = \cup_{i=1}^k W_i(\psi), \tag{36}$$

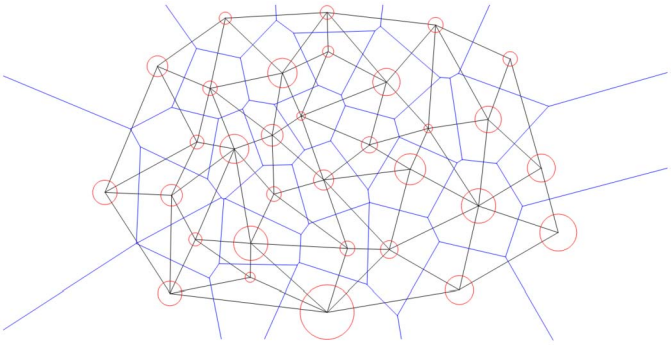


Figure 4. Power diagram (blue) and its dual weighted Delaunay triangulation (black).

where each cell is a convex polytope

$$W_i(\psi) = \{x \in \mathbb{R}^d \mid \mathrm{pow}(x,y_i) \le \mathrm{pow}(x,y_j), 1 \le j \le k\}. \tag{37}$$

The weighted Delaunay triangulation, denoted as $\mathcal{T}(\psi)$, is the Poincare dual to the power diagram, if $W_i(\psi) \cap W_j(\psi) \neq \emptyset$ then there is an edge connecting y_i and y_j in the weighted Delaunay triangulation. Note that $pow(x,y_i) \le pow(x,y_j)$ is equivalent to

$$\langle x, y_i\rangle + \frac{1}{2}(\psi_i - |y_i|^2) \ge \langle x, y_j\rangle + \frac{1}{2}(\psi_j - |y_j|^2). \tag{38}$$

Let $h_i = 1/2(\psi_i - |y_i|^2)$ then we re-write definition of $W_i(\psi)$ as

$$W_i(\psi) = \{x \in \mathcal{R}^d \mid \langle x, y_i\rangle + h_i \ge \langle x, y_j\rangle + h_j, \ \forall j\}. \tag{39}$$

4.2 Damping Newton's Method

Initially, we set $\mathbf{h}^0 = \frac{1}{2}(|y_1|^2, |y_2|^2, \ldots, |y_n|^2)$, where y_i represents the coordinates of the i-th sample in the target domain. The initial power diagram and Weighted Delaunay triangulation are conventional Voronoi diagram and Delaunay triangulation. This guarantees the initial Brenier potential and its Legendre dual are strictly convex, namely the initial height vector belongs to the admissible space, $\mathbf{h}^0 \in \mathcal{H}$.

Assume at the k-th step, we have got $\mathbf{h}^k$, the Brenier potential $u_{\mathbf{h}^k}$ and its Legendre dual $u^*_{\mathbf{h}^k}$, the power diagram $\{W^i_{\mathbf{h}^k}\}_{i=1}^n$. We compute the gradient of Alexandrov energy Eqn. (33) and Hessian matrix H as described in Eqn. (34). Then we solve the linear system:

$$\nabla E(\mathbf{h}^k) = \mathrm{Hess}(\mathbf{h}^k)\mathbf{d}.$$

Next, we need to determine the step length λ. We initialize λ as one, and compute the convex hull of the points

$$\{(y_1, h_1^k + \lambda d_1), (y_2, h_2^k + \lambda d_2), \cdots, (y_n, h_n^k + \lambda d_n)\}.$$

If the convex hull misses any point, then $\mathbf{h}^k + \lambda\mathbf{d}$ is outside the admissible space, the corresponding Brenier potential is not strictly convex. Then we reduce the step length λ by half, $\lambda \leftarrow \frac{1}{2}\lambda$ and repeat the trial. We repeat this procedure and find the minimal l, such that

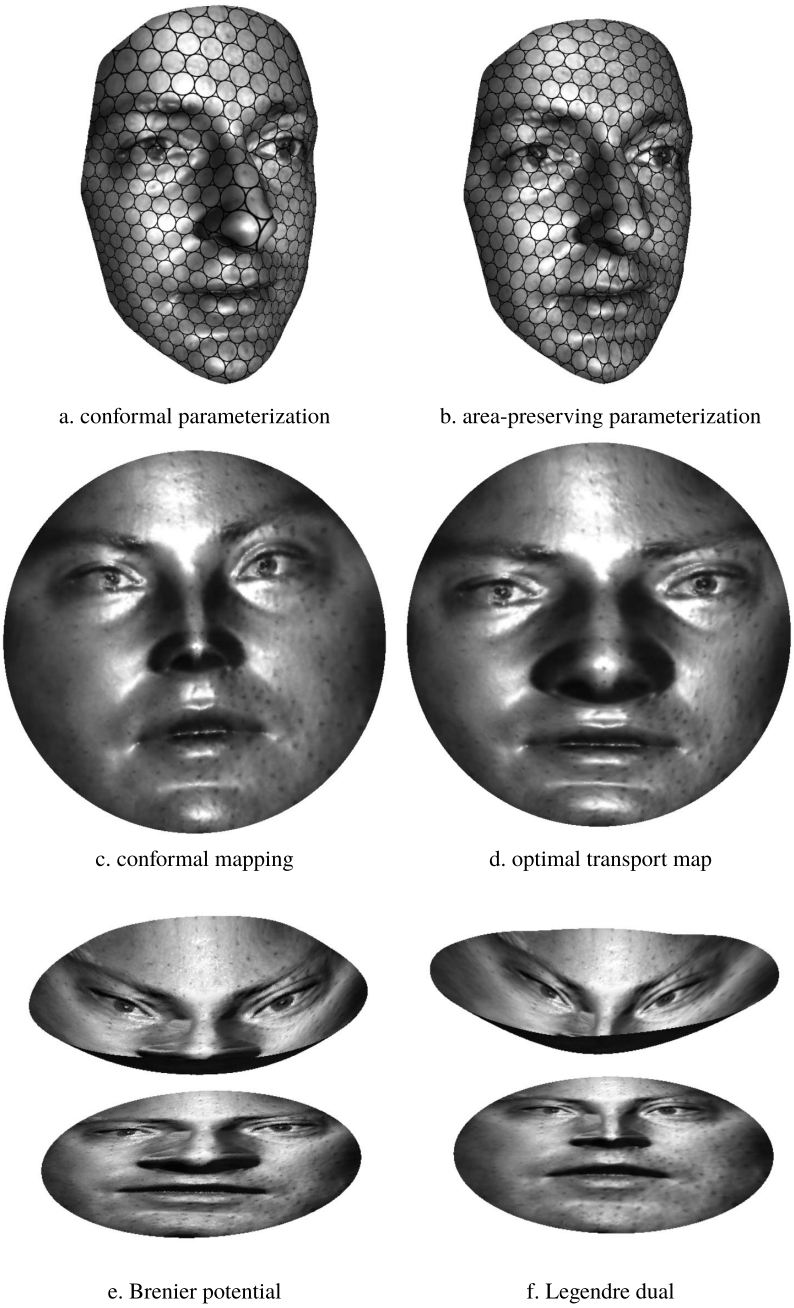
$$\min_l \mathbf{h}^k + 2^{-l}\mathbf{d} \in \Sigma.$$

By iterating this procedure, we reduce the Alexandrov energy monotonously, until the difference between the target measure and the current (measured by the norm of the gradient of the Alexandrov's potential, Eqn. (33)) is less than a prescribed threshold $\varepsilon > 0$.

Algorithm 1 Geometric Variational Method for Optimal Transportation Map

Input: Convex domain Ω with measure μ; Discrete samples $Y := \{y_1, y_2, \cdots, y_n\}$ with measures $\nu_1, \nu_2, \cdots, \nu_n$, with equal measures $\mu(\Omega) = \sum_{i=1}^n \nu_i$.
Output: Optimal transport map $T : \Omega \to Y$.
Initialize $\mathbf{h}^0 = (h_1, h_2, \ldots, h_n) \leftarrow 1/2(|y_1|^2, |y_2|^2, \cdots, |y_n|^2)$.
while true **do**
  Compute the Brenier potential $u_{\mathbf{h}^k}$ and its Legendre dual $u^*_{\mathbf{h}^k}$;
  Project $u_{\mathbf{h}^k}$ and $u^*_{\mathbf{h}^k}$ to obtain the power diagram and weighted Delaunay triangulation;
  Compute the gradient $\nabla E(\mathbf{h}^k)$ of Alexandrov energy Eqn. (33);
  if $\|\nabla E(\mathbf{h}^k)\|$ is less than ε **then**
    return $T = \nabla u_{\mathbf{h}^k}$.
  end if
  Compute the Hessian matrix of Alexandrov energy Eqn. (34) and (30);
  Solve linear system $\nabla E(\mathbf{h}^k) = \mathrm{Hess}(\mathbf{h}^k)\mathbf{d}$;
  Set the step length $\lambda \leftarrow 1$;
  repeat
    $\lambda \leftarrow \lambda/2$;
    Construct the convex hull of $\{(y_i, h_i^k + \lambda d_i)\}_{i=1}^n$;
  until all sample points are on the convex hull;
  update height vector $\mathbf{h}^{k+1} \leftarrow \mathbf{h}^k + \lambda\mathbf{d}$;
end while

As shown in Fig. 5, given a genus zero surface with a single boundary S, it has an induced Euclidean metric $\mathbf{g}$, which induces the surface area element $dA_{\mathbf{g}}$.



a. conformal parameterization

b. area-preserving parameterization

c. conformal mapping

d. optimal transport map

e. Brenier potential

f. Legendre dual

Figure 5. The optimal transportation map for a male face.

After the normalization, the total surface area is π. The Riemann mapping $\varphi : (S, \mathbf{g}) \to (\mathbb{D}, du^2 + dv^2)$ maps the surface onto the unit disk, and pushes the area element to the disk, denoted as $\varphi_{\#} dA_{\mathbf{g}}$. Since Riemann mapping is conformal, the surface area element can be written as

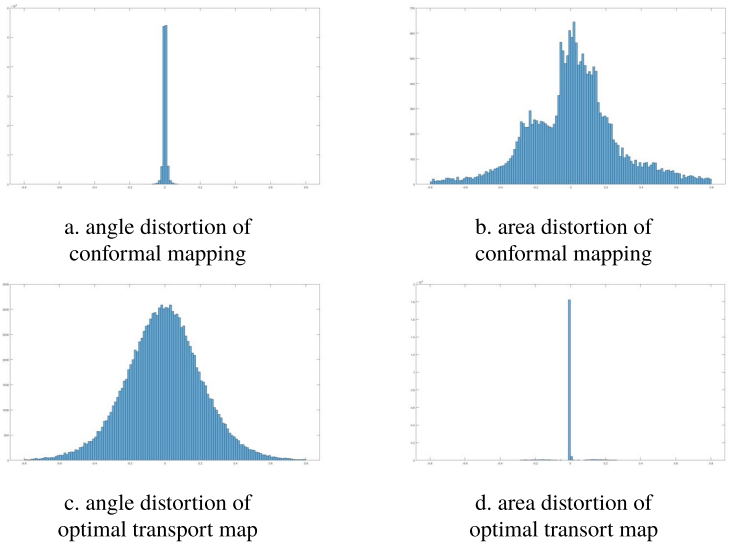
$$dA_{\mathbf{g}}(u,v) = e^{2\lambda(u,v)} dudv,$$

where $e^{2\lambda(u,v)}$ is the area distortion function, can be treated as the target density function.

On the disk, the Lebesgue measure, or equivalently the Euclidean metric $du^2 + dv^2$ induces the Euclidean area element $dudv$. We compute the optimal transportation $T : (\mathbb{D}, dudv) \to (\mathbb{D}, \varphi_{\#} dA_{\mathbf{g}})$ using the geometric variational method. The optimal mapping result is shown between the two planar images. The composition between the Riemann mapping φ and the inverse of the optimal transportation map T^{-1} gives an area-preserving mapping

$$T^{-1} \circ \varphi : (S, \mathbf{g}) \to (\mathbb{D}, dudv), \quad (T \circ \varphi)_{\#} dA_{\mathbf{g}} = dudv.$$

In order to visualize the mapping $T^{-1} \circ \varphi$ is area-preserving, we put circle packing texture on the planar unit disk, and pull it back to the original surface



a. angle distortion of conformal mapping

b. area distortion of conformal mapping

c. angle distortion of optimal transport map

d. area distortion of optimal transort map

Figure 6. Angle distortion and area distortion histograms of the male surface in Fig. 5.

as shown in the top right frame Fig. 5, we can see that the small circles are mapped to ellipses with similar area.

As shown in Fig. 6, we compute the histograms to measure the distortions. The top row shows the histograms of conformal mapping of Fig. 5, the bottom row show those of optimal transportation map. The left column shows the angle distortion histogram, the right column the area distortion histogram. The angle distortion histogram is calculated as follows: the triangle mesh S in $\mathbb{R}^3$ and its planar image share the same triangulation, each corner angle in S corresponds to a planar corner angle. We compute the logarithm of the ratio between the corresponding corner angles, and construct the histograms. From Fig. 6 left column, it is obvious that the angle distortion histogram of conformal mapping highly concentrates in the zero point, this shows the conformal mapping induces very small angle distortions, in contrast, the optimal transportation map induces large angle distortions. The right column shows the area distortion histograms, which is obtained by computing the logarithm of the ration between corresponding face areas. It can be seen that the optimal transport mapping induces very small area distortions, whereas the conformal mapping induces large area distortions.

Fig. 8 shows the computation process of the Buddha surface model. The conformal mapping is computed first, then the optimal transport map is obtained by finding the Brenier potential. The intermediate maps are shown in the figure.

4.3 Monte-Carlo Method

In practice, our goal is to compute the discrete Brenier potential in Eqn. (26) by optimizing the convex energy in Eqn. (30). For low dimensional cases, we can directly use Newton's method by computing the gradient Eqn. (33) and Hessian matrix Eqn. (34). For deep learning applications, direct computation of Hessian matrix is unfeasible, instead we can use gradient descend method or quasi-Newton's method with super-linear convergence. The key of the gradient is to estimate the μ-volume $w_i(h)$. This can be done use Monte-Carlo method: we draw n random samples from the distribution μ, and counts the number of samples falling in $W_i(h)$, the ratio converge to the μ-volume. This method is purely parallel and can be implemented using GPU. Furthermore, we can use hierarchical method to further improve the efficiency: first we classify the target samples to clusters, and compute the optimal transportation map to the mass centers of the clusters; second, for each cluster, we compute the OT map from the corresponding cell to the original target samples within the cluster.

In order to avoid mode collapse, we need to find the singularity sets in Ω. As shown in Fig. 7, the target Dirac measure has two clusters, the source is the uniform distribution on the unit planar disk. The graph of the Brenier potential function is a convex polyhedron with a ridge in the middle. The projection of the ridge on the disk is the singularity set $\Sigma_1(u)$, the optimal mapping is discontinuous on Σ_1. In general cases, if two cells $W_i(h)$ and $W_j(h)$ are adjacent, then we compute the angle between the normals to the corresponding support planes,

$$\theta_{ij} := \cos^{-1} \frac{\langle y_i, y_j \rangle}{|y_i| \cdot |y_j|}$$

if θ_{ij} is greater than a threshold, then the common facet $W_i(h) \cap W_j(h)$ is in the discontinuity singular set.

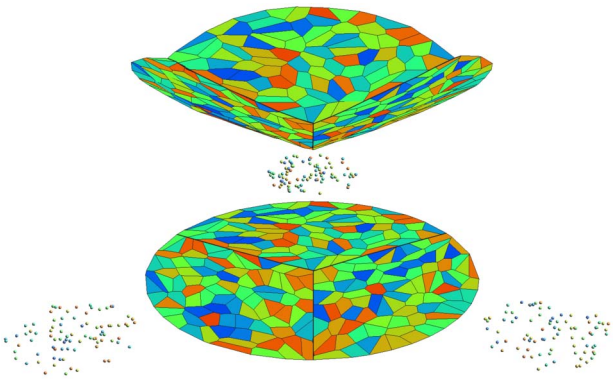


Figure 7. Singularity set of the Brenier potential function, discontinuity set of the optimal transportation map.

5. Manifold Distribution Principle

We believe the great success of deep learning can be partially explained by the well accepted manifold distribution principle.

Manifold Distribution Principle
A natural class of data can be treated as a probability distribution on a low dimensional manifold (data manifold) embedded in the high dimensional ambient space (image space).

Furthermore, the distances among the probability distributions of subclasses on the manifold are far enough to distinguish them.

As shown in Fig. 9, the MNIST data set is a collection of hand written images. Each image is 28×28, which can be treated as a single point in the image space $\mathbb{R}^{28\times 28}$, the MNIST data set is treated as a point cloud. Using Hinton's t-SNE embedding method, we can map the point cloud onto a planar domain, such that each image is mapped to a single point, the mapping is bijective. The images of the same digit are mapped to the same cluster. As shown in the right frame, there are ten clusters on the plane, corresponding to the ten hand written digits. This shows the MNIST point cloud is close to a two dimensional surface embedded in the 784 dimensional image space. We recall the concept of manifold:

Definition 12 (Manifold). *Suppose M is a topological space, covered by a set of open sets $M \subset \bigcup_\alpha U_\alpha$. For each open set U_α, there is a homeomorphism $\varphi_\alpha : U_\alpha \to \mathbb{R}^n$, the pair $(U_\alpha, \varphi_\alpha)$ form a chart. The union of charts form an atlas $\mathcal{A} = \{(U_\alpha, \varphi_\alpha)\}$. If $U_\alpha \cap U_\beta \neq \emptyset$, then the chart transition map is given by $\varphi_{\alpha\beta} : \varphi_\alpha(U_\alpha \cap U_\beta) \to \varphi_\beta(U_\alpha \cap U_\beta)$,*

$$\varphi_{\alpha\beta} := \varphi_\beta \circ \varphi_\alpha^{-1}.$$

The MNIST data set is treated as the *data manifold* Σ; the space of all possible images is the *image space* $\mathbb{R}^{784}$; the plane is the *latent space* $\mathcal{Z}$; the mapping from the data manifold to the latent space $\varphi : \Sigma \to D$ is called the *encoding map*; the inverse mapping $\varphi^{-1} : D \to \Sigma$ is called the *decoding map*. Each hand written digit image $p \in \Sigma$ is a *training sample* on the data manifold, its image of the encoding map $\varphi(p)$ is called the *latent code* of p. The data set can be treated as a probability distribution μ defined on the data manifold Σ, which is called *data distribution*.

Main Tasks In general, deep learning systems have two major tasks:

1. Learn the manifold structure Σ, represented as encoding and decoding maps.
2. Learn the data distribution μ on Σ.

We use manifold view to explain how the denoising is accomplished by a Deep Learning system. Traditional methods Fourier transform the noisy image, filter out the high frequency component, inverse Fourier transform back to the denoised image. Deep learning methods use the clean facial images to train the neural network, obtain a representation of the manifold, then project the noisy image to the manifold, the projection image point is the denoised image. As shown in Fig. 11 left frame, we use a deep learning system to learn the data manifold Σ of clean human facial images. An facial image with noise is $\tilde{p}$, which is not on Σ but close to the manifold. We project $\tilde{P}$ to Σ using the Riemannian metric in the image space $\mathbb{R}^n$, the closest point on Σ to $\tilde{p}$ is p, then p is the denoised image.

Traditional method is independent of the content of the image; ML method heavily depends on the content of the image. The prior knowledge is encoded by the manifold. If the wrong manifold is chosen, then the denoising result is of non-sense. As shown in Fig. 12 right frame, we use the cat face manifold to denoise a human face image, the result looks like a cat face.

6. Manifold Learning

Learning data manifold structure is equivalent to learning the encoding and decoding maps. The encoding mapping $\varphi : \Sigma \to \mathcal{Z}$ maps the data manifold to the latent space. It push-forwards μ to the *latent distribution*, denoted as $\phi_\# \mu$. Given the data manifold Σ and the latent space $\mathcal{Z}$. There are infinite many encoding mappings. In practice, it is crucial to choose the appropriate mapping, that preserves the data distribution. We use a low dimensional example to il-

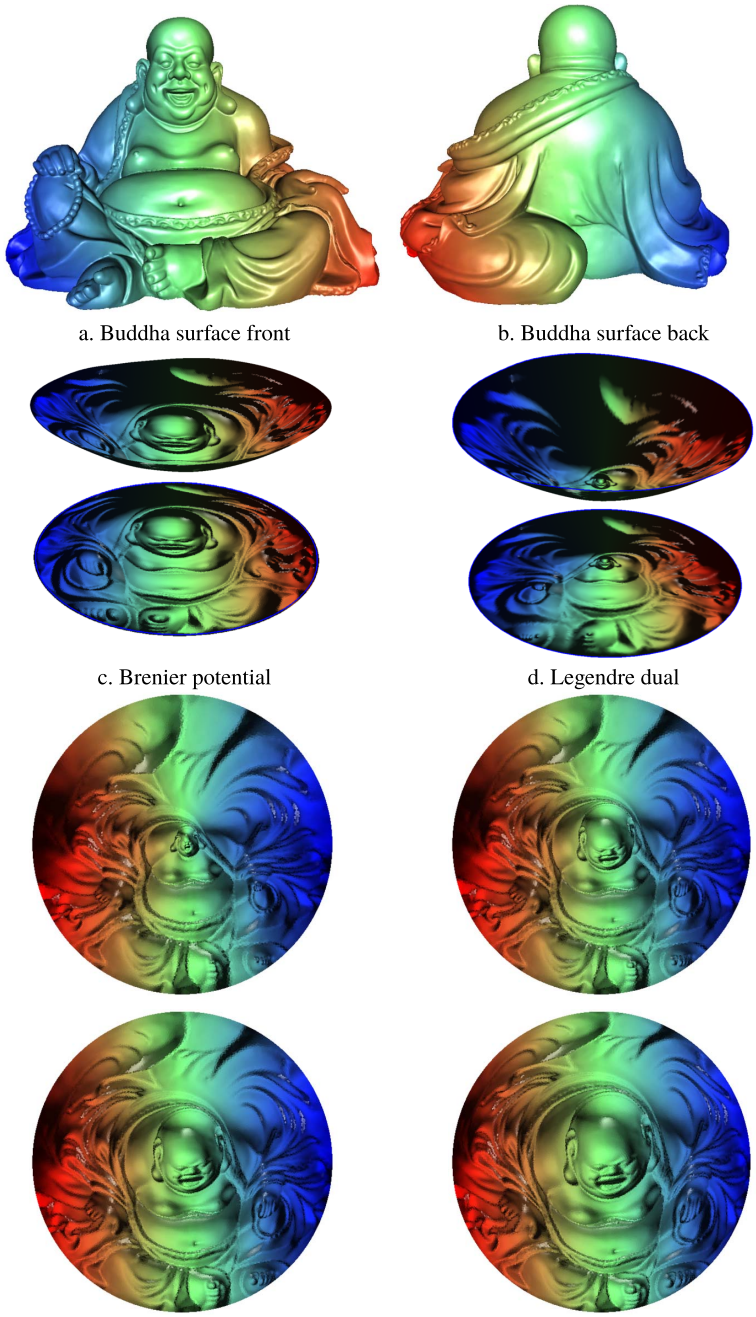


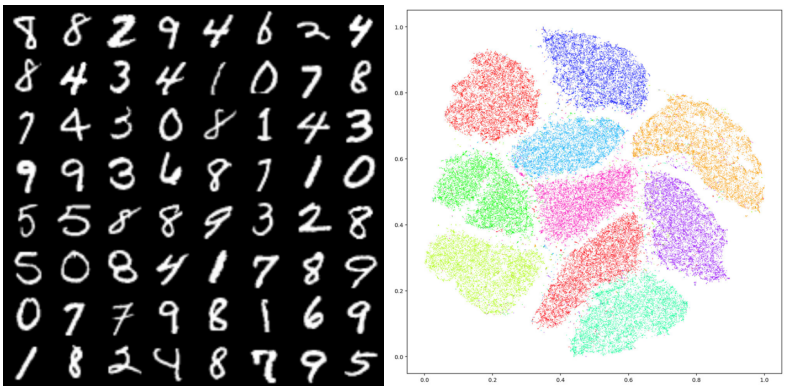
Figure 8. Buddha surface, the last two rows show the intermediate computational results during the optimization.

lustrate the concepts as shown in Fig. 13. The Buddha surface represents the data manifold Σ, μ is the uniform distribution on Σ. Each row shows one encoding map. In the top row, if we uniformly sample the unit disk in the latent space, the samples are pulled back to the surface by the decoding map, then the pullback samples on Σ are highly non-uniform. In contrast, in the bottom row, the uniform latent samples are pulled back to uniform samples on the surface. This shows the encoding map in the bottom row preserves the data distribution μ in the latent space.

In practice, many methods have been proposed to compute the encoding/decoding maps, such as VAE (variational auto-encoder), [31, 26], WAE (Wasserstein auto-encoer) [24], adversarial auto-encoder [39] and so on.

6.1 ReLu Deep Neural Network

In deep learning, the deep neural networks are used to approximate mappings between Euclidean spaces. One of the most commonly used activation function is the ReLU function, $\sigma(x) = \max\{x, 0\}$. When



a. LeCunn's MNIST handwritten digits samples on manifold

b. Hinton's t-SNE embemdding on latent space

Figure 9. The MNIST data set is a two dimensional surface in the image space.

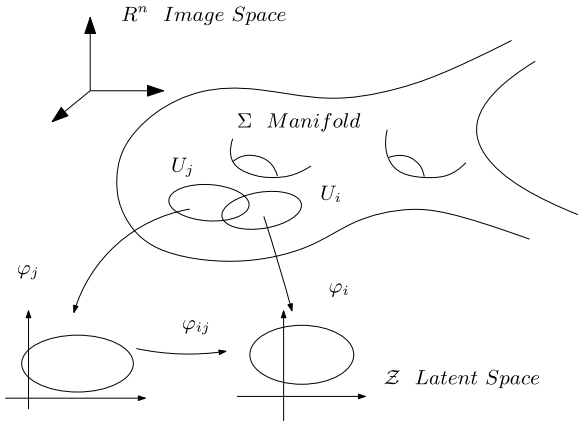


Figure 10. The concept of manifold.

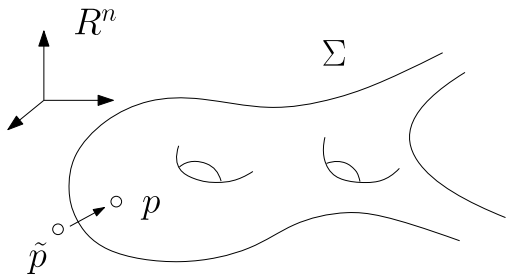


Figure 11. Image denoising as projecting to a manifold.

x is positive, we say the neuron is *activated*. One neuron represents a function $\sigma(\sum_{i=1}^{k}\lambda_i x_i - b_i)$, where λ_i's are *weights*, b_i the *bias*. Many neurons are connected to form a network. A ReLU deep neural network (DNN) represents a piecewise linear map.

Definition 13 (ReLU DNN). *For any number of hidden layers $k \in \mathbb{N}$, input and output dimensions $w_0, w_{k+1} \in \mathbb{N}$, a $\mathbb{R}^{w_0} \to \mathbb{R}^{w_{k+1}}$ ReLU DNN is given by specifying a sequence of k natural numbers $w_1, w_2, \ldots, w_k$ representing widths of the hidden layers, a set of k affine transformations $T_i : \mathbb{R}^{w_{i-1}} \to \mathbb{R}^{w_i}$ for $i = 1, \ldots, k$ and a linear transformation $T_{k+1} : \mathbb{R}^{w_k} \to \mathbb{R}^{w_{k+1}}$ corresponding to weights of hidden layers.*

The mapping $\varphi_\theta : \mathbb{R}^{w_0} \to \mathbb{R}^{w_{k+1}}$ represented by this ReLU DNN is

$$\varphi_\theta = T_{k+1} \circ \sigma_k \circ T_k \circ \cdots \circ T_2 \circ \sigma_1 \circ T_1, \tag{40}$$

where $\circ$ denotes mapping composition, θ represent all the weight and bias parameters, σ_i represents the mapping $\sigma_i : \mathbb{R}^{w_{i-1}} \to \mathbb{R}^{w_i}$ $\sigma_i = (\sigma_i^1, \sigma_i^2, \cdots, \sigma_i^{w_i})$,

$$\sigma_i^j = \sigma\left(\sum_{k=1}^{w_{i-1}} \lambda_i^{jk} x_k - b_i^j\right).$$

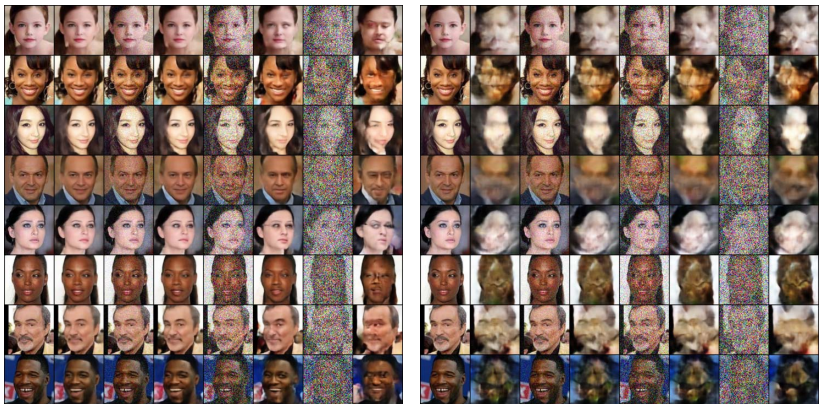
Definition 14 (Activated Path). *Given a point $\mathbf{x} \in \mathcal{X}$ in the input space $\mathcal{X}$, the* activated path *of $\mathbf{x}$ consists all the activated neurons when $\varphi_\theta(\mathbf{x})$ is evaluated, and denoted as $\rho(\mathbf{x})$. Then the activated path defines a set-valued function $\rho : \mathcal{X} \to 2^{\mathcal{S}}$ ($\mathcal{S}$ is the set of all neurons, $2^{\mathcal{S}}$ are all the subsets of $\mathcal{S}$).*

Fixing the parameter θ, the map φ_θ induces cell decomposition for the input space and the output space.

Definition 15 (Cell Decomposition). *Fix a map φ_θ represented by a ReLU DNN, two data points $\mathbf{x}_1, \mathbf{x}_2 \in \mathcal{X}$ are* equivalent, *denoted as $\mathbf{x}_1 \sim \mathbf{x}_2$, if they share the same activated path, $\rho(\mathbf{x}_1) = \rho(\mathbf{x}_2)$. Then each equivalence relation partitions the ambient space $\mathcal{X}$ into cells,*

$$\mathcal{D}(\varphi_\theta) : \mathcal{X} = \bigcup_\alpha U_\alpha,$$

each equivalence class corresponds to a cell: $\mathbf{x}_1, \mathbf{x}_2 \in U_\alpha$ if and only if $\mathbf{x}_1 \sim \mathbf{x}_2$. $\mathcal{D}(\varphi_\theta)$ is called the cell decomposition induced by the encoding map φ_θ. The number of cells is denoted as $|\mathcal{D}(\varphi_\theta)|$.



a. projection to a human facial photo manifold b. projectial to a cat face image manifold

Figure 12. Human facial image denoising by projection to the data manifold.

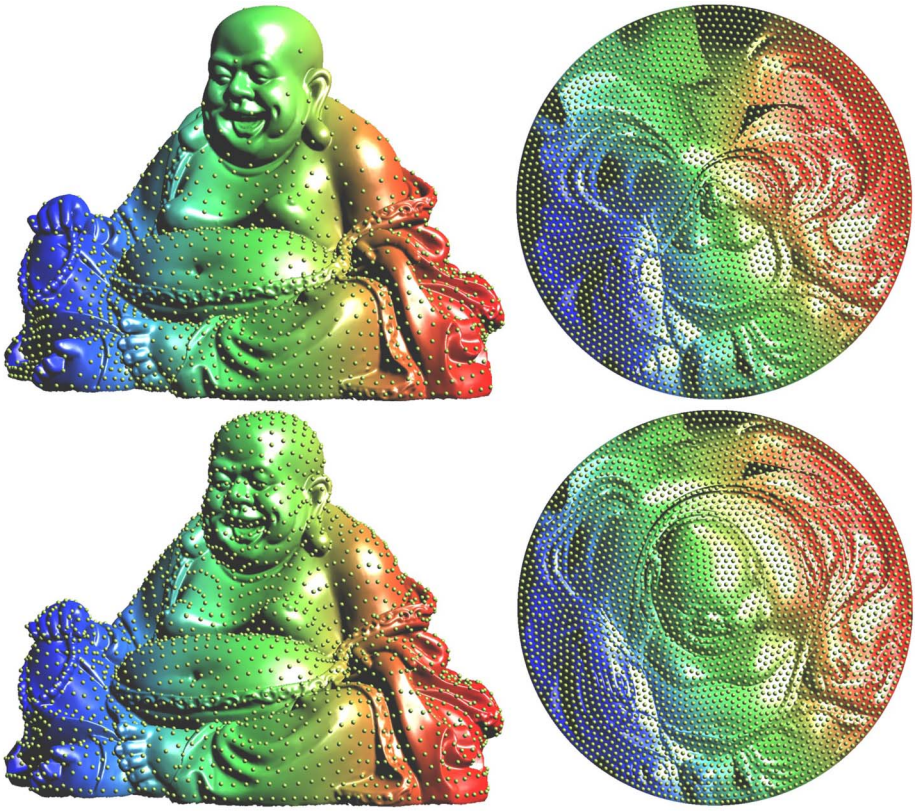


Figure 13. Different encoding mappings from the manifold to the planar disk.

Furthermore, φ_θ maps the cell decomposition in the ambient space $\mathcal{D}(\varphi_\theta)$ to a cell decomposition in the latent space. The restriction of φ_θ on each cell is a linear map. The number of cells in $\mathcal{D}(\varphi_\theta)$ describes the capacity of the network, namely the learning capability of the network.

Definition 16 (Learning Capability). *Given a ReLU DNN N with fixed architecture, the complexity of the network $\mathcal{N}(N)$ is defined as the maximal number of cells of $\mathcal{D}(\varphi_\theta)$,*

$$\mathcal{N}(N) := \max_\theta |\mathcal{D}(\varphi_\theta)|.$$

We can explicitly estimate the upper bound of the network capacity $\mathcal{N}(N)$. The maximum number of parts one can get when cutting d-dimensional space $\mathbb{R}^d$ with n hyperplanes is denoted as $\mathcal{C}(d,n)$, then by induction, one can easily show that

$$\mathcal{C}(d,n) = \binom{n}{0} + \binom{n}{1} + \binom{n}{2} + \cdots + \binom{n}{d}. \tag{41}$$

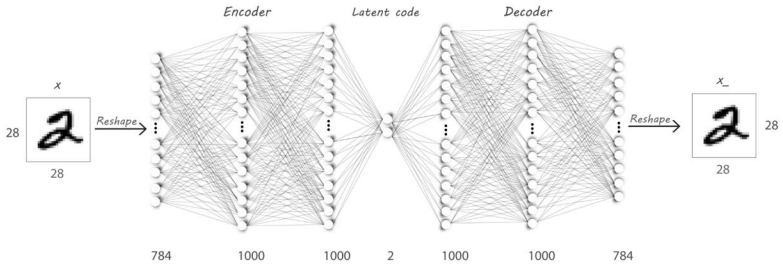
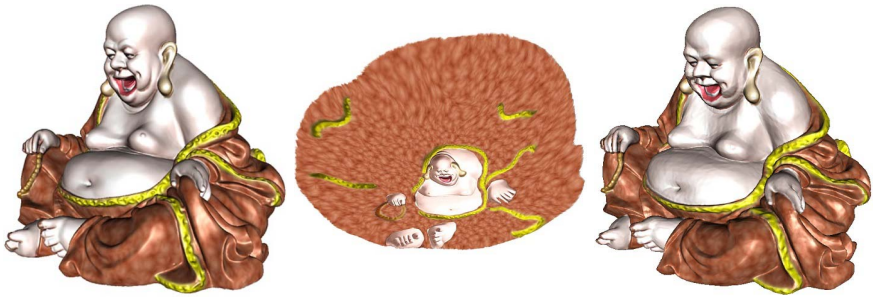


Figure 14. Auto-encoder architecture.



a. Input manifold $\Sigma \subset \mathcal{X}$

b. latent representation $D = \varphi_\theta(M) \subset \mathcal{Z}$

c. reconstructed manifold $\tilde{\Sigma} = \psi_\theta(D) \subset \mathcal{X}$

Figure 15. Manifold embedding computed by an Auto-encoder.

We can easily get the upper bound estimation,

Theorem 9. *Given a ReLU DNN $N(w_0,\ldots,w_{k+1})$, representing PL mappings $\varphi_\theta : \mathbb{R}^{w_0} \to \mathbb{R}^{w_{k+1}}$ with k hidden layers of widths $\{w_i\}_{i=1}^k$, then the linear rectified complexity of N has an upper bound,*

$$\mathcal{N}(N) \leq \Pi_{i=1}^{k+1} \mathcal{C}(w_{i-1}, w_i). \tag{42}$$

6.2 AutoEncoder

One of the most popular model for learning the encoding and decoding maps is *AutoEncoder* as shown in Fig. 14. The AutoEncoder model consists two symmetric Deep Neural Networks, the first network represents the encoder, the second network represents the decoder. The number of nodes in the input and the output layers equals to the dimension of the ambient space. Between the encoder and decoder, there is a bottle neck layer. The number of nodes in the bottle neck layer equals to the dimension of the latent space.

We denote the ambient space as $\mathcal{X}$, latent space as $\mathcal{Z}$, encoding map $\varphi_\theta : \mathcal{X} \to \mathcal{Z}$, decoding map $\psi_\theta : \mathcal{Z} \to \mathcal{X}$. We sample the data manifold $\Sigma \subset \mathcal{X}$ to get training samples $\{x_1, x_2, \cdots, x_n\} \subset \Sigma$, and apply the L^2-norm as the loss function $\mathcal{L}_\theta$. The training process is the optimization

$$\min_\theta \mathcal{L}_\theta(x_1,\ldots,x_n) = \min_\theta \sum_{i=1}^n |x_i - \psi_\theta \circ \varphi_\theta(x_i)|^2. \tag{43}$$

Fig. 15 shows one example of surface embedding using an Auto-encoder. We uniformly sample the Buddha surface Σ in (a), then train an Auto-encoder using formula Eqn. 43, the latent codes of the samples are shown in (b), the decoded surface $\tilde{\Sigma}$ is shown in (c). We can see the reconstructed surface is very similar to the input surface, with user controlled Hausdorff distance. Fig. 16 shows the cell decomposition of the ambient space $\mathcal{X}$ and the latent space $\mathcal{Z}$ induced by the encoding map φ_θ and the decoding map ψ_θ.

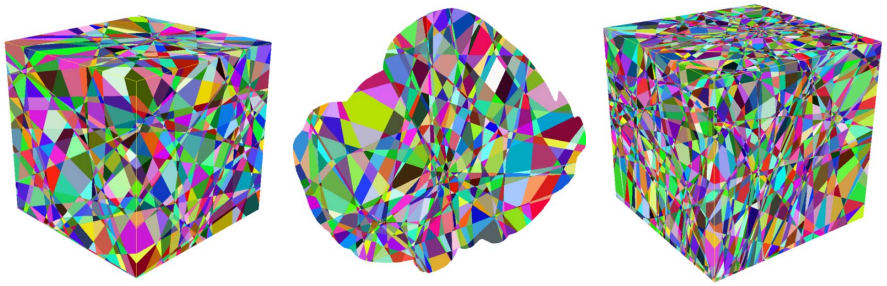
In the following, we analyze the accuracy of manifold learning using a surface example. Given the input surface Σ embedded in $\mathbb{R}^3$, given any point $p \in \mathbb{R}^3$, the *closest point* on Σ to p is defined as

$$\pi(p,\Sigma) := \text{argmin}_{q\in\Sigma} |p-q|^2.$$

The *medial axis* of the surface Σ is defined as

$$\Gamma(\Sigma) := \{p \in \mathbb{R}^3 : |\pi(p,\Sigma)| > 1\}.$$

where $|\cdot|$ represents the cardinality of the set. For any point $p \in \Sigma$, the *local feature size* of p is the distance from p to the medial axis $\Gamma(\Sigma)$. Suppose the samplings on Σ are $X = \{x_1, x_2, \ldots, x_n\}$, such that for any point $q \in \Sigma$, the geodesic disk $c(q,\delta)$ intersects X is non-empty, and the geodesic distance between any pair of samples is greater than ε, then X is called a (δ,ε) *sampling*. Given such a sampling, we can compute the geodesic Delaunay triangulation of X, this induces a polyhedral surface $\tilde{\Sigma}$. By geometric approximation theory,



d. cell decomposition $\mathcal{D}(\varphi_\theta)$ | e. latent space cell decomposition | f. cell decomposition $\mathcal{D}(\psi_\theta \circ \varphi_\theta)$

Figure 16. The cell decompositions induced by the Auto-encoder.

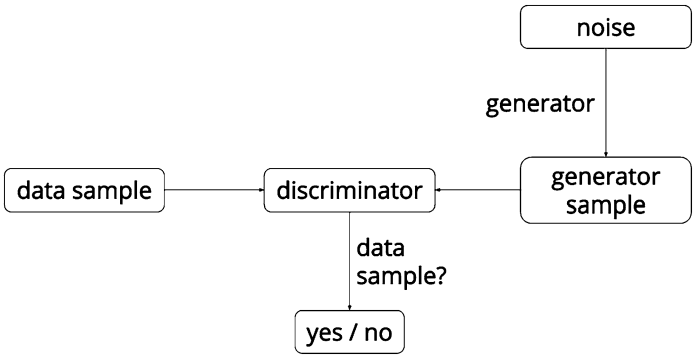


Figure 17. The framework of a GAN model.

suppose Σ is C^2 smooth, we can determine the parameters δ, ε by the injective radius, the principle curvature and the local feature size, such that $\tilde{\Sigma}$ approximates the original surface Σ with arbitrary precision in terms of Hausdorff distance, Riemannian metric, Laplace-Beltami operator, curvature measures and so on.

Assume the network capacity for the auto-encoder is big enough, the (δ, ε) samples are the training set and the optimization reduces the loss function to be 0, then the restriction of $\psi_\theta \circ \varphi_\theta$ equals to identity, then the auto-encoder recovers $\tilde{\Sigma}$. By the construction, the decoded surface approximates the original surface with user desired accuracy. This argument can be generalized to higher dimensional manifolds. In reality, the data manifold is unknown and it is hard to figure out its injective radius, curvatures and local feature size, the optimization of deep networks often gets stuck at the local optima. There are many widely open challenges for learning the manifold structure.

7. Generative Adversarial Networks

Generative Adversarial Networks (GAN) is one of the most popular generative model in deep learning. It has many merits, such as it can automatically generate samples, the requirement for the data samples is reduced; it can model arbitrary data distribution without closed form expression. As shown in Fig. 17, a GAN model includes two deep neural networks, the generator and the discriminator. The generator converts a white noise (user prescribed distribution in the latent space) to generated samples, the discriminator takes both the real data samples and the fake generated samples and verify whether the current sample is authentic or fake.

7.1 Competition vs. Collaboration

The generator and the discriminator competes with each other, the generator improves the quality of the generated samples to confuse the discriminator, and the discriminator improves the discriminating capability and detect the fake samples. Eventually, the system reaches the Nash equilibrium, the discriminator can not differentiate the generated ones from the real samples, then the generated samples can be applied to real applications, such as training other recognition systems and so on.

Wasserstein GAN applies optimal transport method as shown in Fig. 18. The generator G computes the optimal transport map $g_\theta : \mathcal{Z} \to \Sigma$, which transforms the white noise ζ in the latent space $\mathcal{Z}$ to the generated distribution $\mu_\theta = (g_\theta)_\# \zeta$. The discriminator D computes the Kantorovich potential

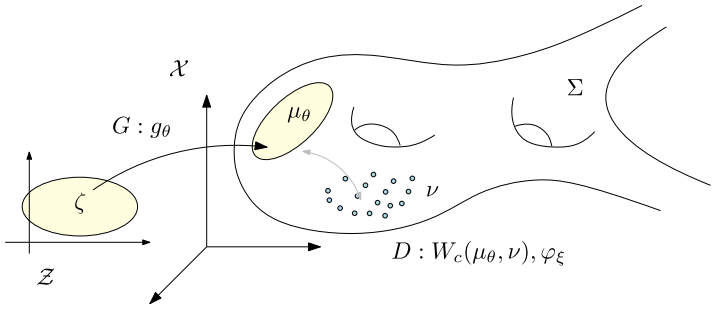


Figure 18. The framework of a GAN model, $\mathcal{Z}$ is the latent space, ζ the white noise, $\mathcal{X}$ the image space, Σ the data manifold, G generator, D discriminator.

φ_ξ, then compute the Wasserstein distance between μ_θ and the real data distribution ν,

$$\mathcal{W}_c(\mu_\theta, \nu) = \max_{\varphi_\xi} \int_X \varphi_\xi(x) d\mu_\theta(x) + \int_Y \varphi_\xi^c(y) d\nu(y),$$

where X and Y should be the data manifold Σ, in practice, they are replaced by the image space $\mathcal{X}$ in [6]. The whole training process of WGAN model is a min-max optimization,

$$\min_\theta \max_\xi \int_X \varphi_\xi(x) d\mu_\theta(x) + \int_Y \varphi_\xi^c(y) d\nu(y).$$

One can choose L^1-cost, then $c(x,y) = |x-y|$, $\varphi^c = -\varphi$, given φ is 1-Lipsitz, then the WGAN model optimizes

$$\min_\theta \max_\xi \int_X \varphi_\xi \circ g_\theta(z) d\zeta(z) - \int_Y \varphi_\xi(y) d\nu(y),$$

namely

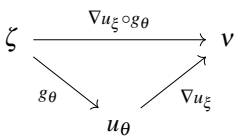
$$\min_\theta \max_\xi \quad \mathbb{E}_{z\sim\zeta}(\varphi_\xi \circ g_\theta(z)) - \mathbb{E}_{y\sim\nu}(\varphi_\xi(y)),$$

with the constraint that φ_ξ is 1-Lipsitz.

If we use L^2 cost, then the discriminator computes the Kantorovich potential φ_ξ for the purpose of Wasserstein distance $\mathcal{W}_2(\mu_\theta, \nu)$, then the Brenier potential u_ξ and the optimal transport map T_ξ can be derived directly

$$u_\xi = \frac{1}{2}|x|^2 - \varphi_\xi(x), \quad T_\xi = \nabla u_\xi.$$

T_ξ transforms the generated distribution μ_θ to the real data distribution ν. The generator g_θ transforms ζ to μ_θ, then the composition $\nabla u_\xi \circ g_\theta$ maps the latent white noise ζ to the data distribution ν, as shown in the following commutative diagram,



The generator seeks a measure preserving map to transform ζ to ν. In each optimization step, the generator finds the current g_θ, which gives a transport map from ζ to $\mu_x i$, the discriminator computes u_ξ, which transport μ_ξ to ν. The composition $\nabla u_\xi \circ g_\theta$ gives a transport map from ζ to ν. Therefore, we can use $\nabla u_\xi \circ g_\theta$ to update the generator g_θ, this will improve the convergence rate. Currently, the generator and the discriminator do not share intermediate computational results, which make the system highly inefficient. The competition between the generator and the discriminator should be replaced by collaboration.

7.2 Memorization vs. Learning

In general deep neural networks have huge amount of parameters, such that their capacities are big enough to memorize all the training samples. So the following question is naturally raised:

Memorization vs. Learning

Does a deep learning system really learn something or just memorize all the training samples?

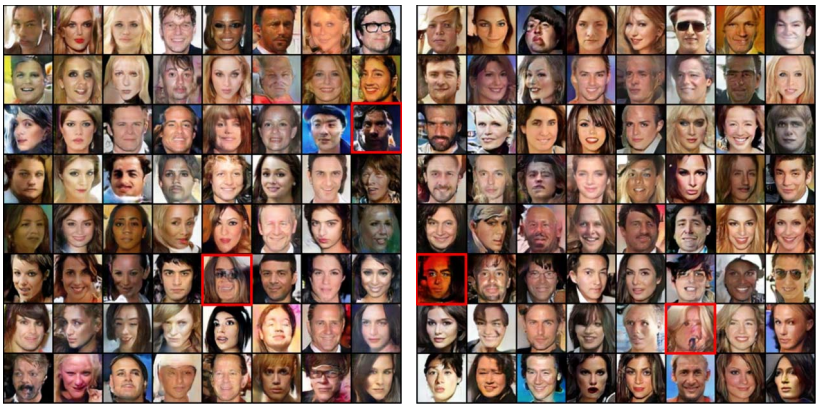
Generally speaking, in deep learning applications, the real data distribution ν is approximated by the empirical distribution: $\hat{\nu} = \frac{1}{n}\sum_{i=1}^n \delta(y - y_i)$, where $\{y_1, y_2, \ldots, y_n\}$ are the training samples, either the raw samples on the data manifold or the latent codes in the latent space. If we use the quadratic Euclidean distance as the cost function, then both the generator and the discriminator compute the optimal transport maps, or equivalently the Brenier potentials. From the formula of the semi-discrete Brenier potential,

$$u = \max_{i=1}^n \{\langle x, y_i\rangle - h_i\}$$

we can tell that the system really memorizes all the training samples $\{y_i\}$; but also learns the probability for each sample represented by $\{h_i\}$, which are obtained by non-linear optimization.

(a) VAE (b) WGAN (c) Our model, AE-OT

Figure 19. Comparison between conventional models VAE and WGAN with our model AE-OT using MNIST data set.



(a) WGAN-GP (b) WGAN-div

Figure 20. Mode collapsing in WGAN-GP and WGAN-div model on CelebA data set.

Hence deep learning systems both memorize the training samples and learn the probability measure.

7.3 Mode Collapsing

GANs are sensitive to hyper-parameters and notoriously difficult to train. The training process is highly unstable, and often diverge. GANs suffer from *mode collapsing*, the generated distributions often miss some modes in the training data set. For example, if a GAN model is trained to learn the MNIST data sets, which has multiple modes representing the ten hand written digits, then the GAN model may only learn 6 of them and forget the other 4 modes, or it captures some modes in the intermediate stage, but forgets part of them in the final stage. GANs also suffer from *mode mixture*, they generate unrealistic samples mixing different modes. As shown in Fig. 19, VAE [31] or WGAN [6] models suffer from mode mixture, they generate unrecognizable hand written digit images, which look like the interpolation/mixture of some digits. Fig. 20 shows mode collapsing on CelebA data set using WGAN-GP [22] and WGAN-div [28] models.

Mode collapsing can be explained using the regularity theory of optimal transport maps. As shown in Fig. 21, we use Monte-Carlo method to compute the optimal transport map between the uniform distribution defined on a rectangle to that on a dumb bell shape. Even the target domain is simply connected, because it is concave, the OT map is discontinuous at the singular sets γ_1 and γ_2 as shown in the left frame.

As we analyzed before, deep neural networks can only represent continuous mappings, but the optimal transport map is discontinuous given the target support is concave, this intrinsic conflict causes mode collapse and mode mixture.

If the target measure ν has multiple modes, its support has multiple connected components, then the continuous map may cover one connected component and miss the other modes, this induces mode collapse; or the continuous map covers all the modes but also the gaps among the modes, then the samples generated in the gap area will mix samples from different modes, hence this induces mode mixture. As shown in Fig. 22, each orange spot represents a mode

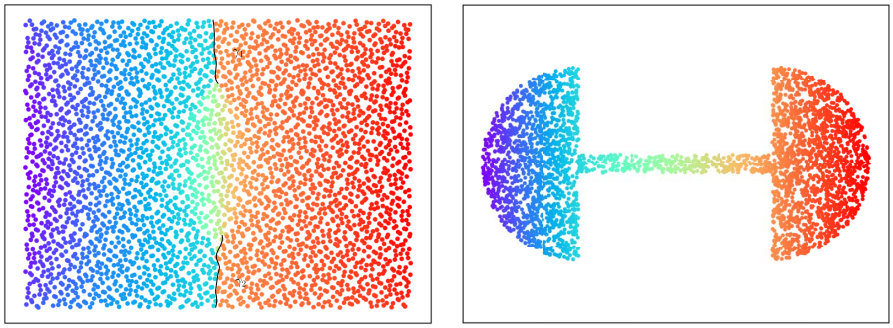
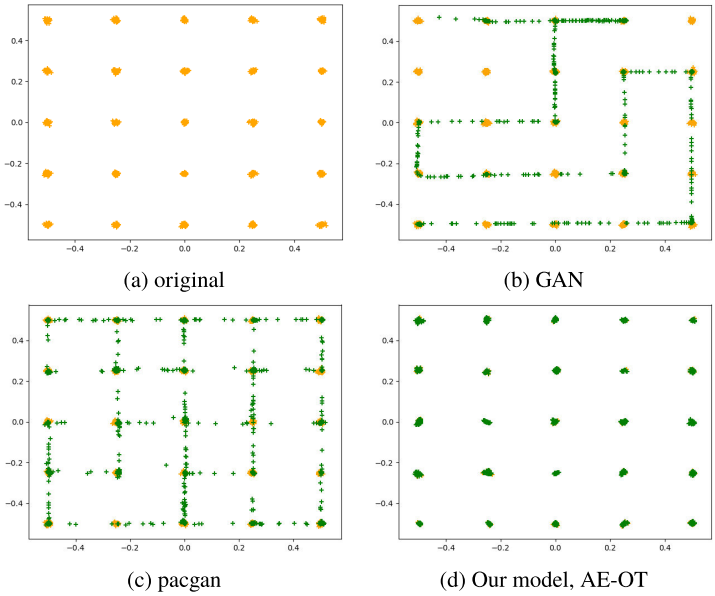


Figure 21. Discontinuous Optimal transportation map, produced by a GPU implementation of algorithm based on regularity theorem. γ_1 and γ_2 are two singularity sets.



(a) original (b) GAN

(c) pacgan (d) Our model, AE-OT

Figure 22. Comparison between conventional models with AE-OT.

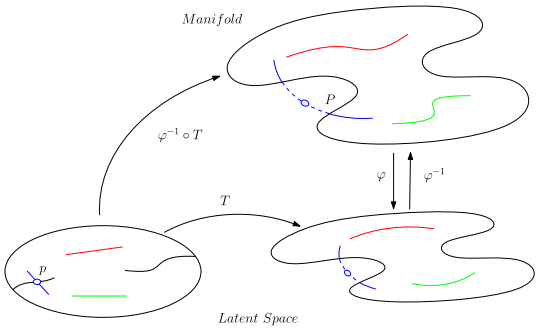


Figure 23. Singularity set detection.

in frame (a); the GAN model [19] misses some modes and also covers the gaps among the modes in frame (b); the pacgan model [36] covers all the modes but also covers the gaps among them. Hence GAN model and pacgan model suffer from both mode collapse and mode mixture.

In order to verify our hypothesis that the transport map is discontinuous on the singularity sets in real applications, we design and perform a experiment using human facial image data set celebA. As shown in Fig. 23, we use an auto-encoder to encode the data manifold Σ to the latent space, $\varphi : \Sigma \to \mathcal{Z}$, φ push-forwards the data distribution μ to the latent code distribution $\varphi_{\#}\mu$; then in the latent space, we compute an optimal transport map from a uniform distribution on the unit ball to the latent code distribution $\varphi_{\#}\mu$; we draw line segments in the unit ball, which are maps to curves on the data manifolds, each curve is a interpolation in the facial image set. As shown in Fig. 24, each role is an interpolation curve on the human facial image manifold.

As shown in Fig. 23, there are singularity sets in the unit ball and a blue line segment intersects the singularity sets at p, then $T(p)$ is outside the latent code set $\varphi(\Sigma)$, the decoded image $\varphi^{-1}(T(p))$ is outside the data manifold Σ. In this way, we can de-

Figure 24. Interpolation curves on facial photo manifold.

Figure 25. Facial images generated by an AE-OT model, the central image shows the boundary of the facial photo manifold.

tect the boundary of the data manifold Σ. An image on the human facial image manifold means a human face, which is physically "allowable", satisfies all the anatomically, biologically laws, but with zero probability to appear in reality. As shown in Fig. 25, we start from a boy image with brown eyes and end at a girl image with blue eyes. In the middle of the interpolation, we generate a facial image with one blue eye and one brown eye. This type of human faces exist in real world, but the probability to encounter such a person is almost zero in practice. All the training facial images are either brown eyes or blue eyes, the generated facial image with different eye colors is on the boundary of the data manifold. This demonstrates that the existence of singularity set Γ, and the transport map T is discontinuous at the Γ.

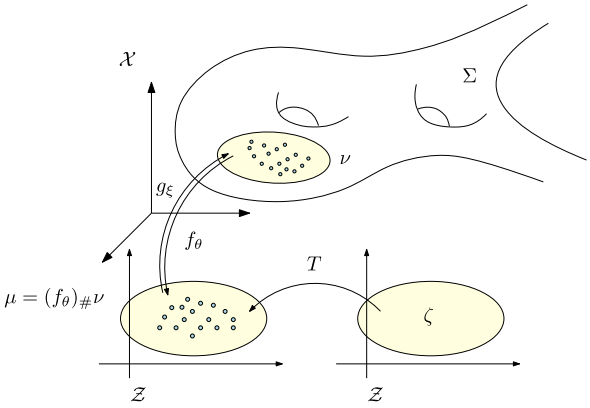


Figure 26. The framework of AE-OT model.

7.4 AE-OT Model

In order to eliminate mode collapse, improve the stability and make the whole model more understandable, we propose a novel generative model: AE-OT model. As shown in Fig. 26, the model consists two parts AE and OT. The AE network is an auto-encoder, which focuses on manifold learning and computes the encoding map $f_\theta : \Sigma \to \mathcal{Z}$ and the decoding map $g_\xi : \Sigma \to \mathcal{Z}$; the OT module is in charge of probability distribution transformation and finds the optimal transport map using our geometric variational approach. The OT module can be implemented either using a deep neural network and optimized by training or directly using geometric method, such as Monte Carlo OT algorithm on GPU.

The mode collapses in conventional generative models are mainly caused by the step of computing transport map, because the transport map is discontinuous but DNNs can only represent continuous maps. The AE-OT model conquers this fundamental difficulty in the following way: observe Fig. 27, in the

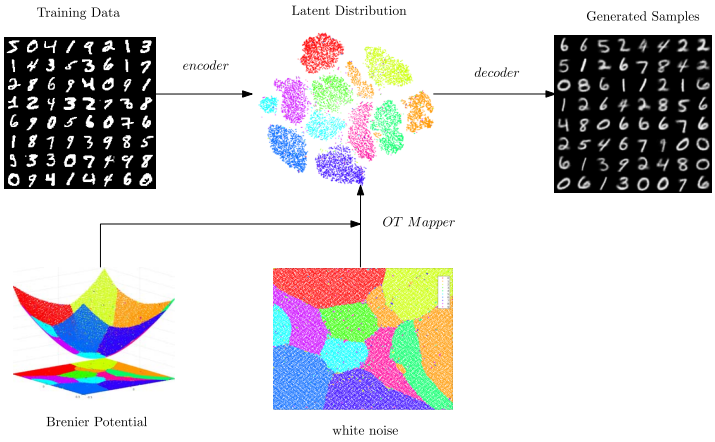
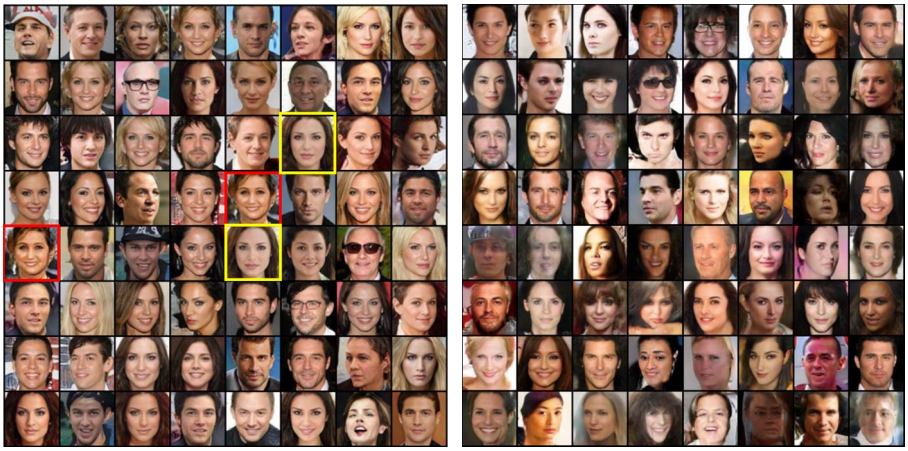


Figure 27. AE-OT model for MNIST data set.



(c) CRGAN - mode collapsing (d) Our model

Figure 28. Comparison between CRGAN [41] and our model.

latent space the latent code distribution has multiple clusters, the support rectangle of the white noise is partitioned into 10 cells as well, each cell is mapped to a cluster with the same color. Therefore, the optimal transport map between the noise and the latent code is discontinuous across the cell boundaries. Instead of computing the OT map itself, the AE-OT model computes the Brenier potential (lower-left corner), which is continuous (but not globally differentiable) and representable by neural networks. Since the OT map covers all the clusters of the latent code distribution, and skips all the gaps among the clusters, no mode collapse or mode mixture can happen.

Furthermore, the AE-OT model has the merits: solving Monge-Ampère equation is reduced to a convex optimization, which has unique solution due to the Brenier theorem 2. The optimization won't be trapped in a local optimum; the Hessian matrix of the energy has explicit formulation. The Newton's method can be applied with second order convergence; or the quasi-Newton's method can be used with super-linear convergence. Whereas conventional gradient descend method has linear convergence; the approximation accuracy can be fully controlled by the density of the sampling density by using Monte-Carlo method; the algorithm can be refined to be hierarchical and self-adaptive to further improve the efficiency; the parallel algorithm can be implemented using GPU. By comparing Fig. 20 and Fig. 28, we can see that the AE-OT model greatly reduces the mode collapse and mode mixture. Fig. 29 shows the generated facial images by training our model on the CelebAHQ data set.

8. Conclusion

This work focuses on a geometric view of optimal transport to understand deep learning models, such as generative adversarial networks (GANs). By manifold distribution principle, deep learning systems learn probability distributions on manifolds, therefore they have two major tasks: one is manifold learning, the other is probability measure learning.

Figure 29. Human facial images generated by our model.

Manifold learning is reduced to construct encoding and decoding maps between the data manifold and the latent space. The probability distribution learning can be achieved by optimal transport methods. The Brenier theory in optimal transport has intrinsic relation with Alexandrov theorem in convex geometry via Monge-Ampère equation. This leads to a geometric variational algorithm to compute optimal transport maps. By applying OT theory, we analyze the conventional generative models, and find that the generator and discriminator in a GAN model should collaborate instead of compete with each other; the GAN model both memorizes all the training samples and learns the probability measure; furthermore, the regularity theory of Monge-Ampère equation explains the intrinsic reason for mode collapse. In order to eliminate mode collapse, a novel AE-OT model is introduced, which computes the continuous Brenier potential instead of the discontinuous transport maps.

Optimal transport theory and Riemannian geometry lay down the theoretic foundation of deep learning. In the future, we will explore further to use modern geometry theories to understand deep learning algorithms and design novel models.

References

[1] Koray Kavukcuoglu, Aaron van den Oord, Oriol Vinyals. Neural discrete representation learning. In *NeurIPS*, 2017.

[2] Aleksandr Danilovich Alexandrov. *Convex polyhedra*. Translated from the 1950 Russian edition by N. S. Dairbekov, S. S. Kutateladze and A. B. Sossinsky. Springer Monographs in Mathematics, 2005.

[3] Dongsheng An, Yang Guo, Na Lei, Zhongxuan Luo, Shing-Tung Yau, and Xianfeng Gu. Ae-ot: A new generative model based on extended semi-discrete optimal transport. In *International Conference on Learning Representations*, 2020.

[4] Karen Simonyan, Andrew Brock, Jeff Donahue. Large scale gan training for high fidelity natural image synthesis. In *International Conference on Learning Representations*, 2019.

[5] Sigurd Angenent, Steven Haker, and Allen Tannenbaum. Minimizing flows for the Monge-Kantorovich problem. *SIAM J. Math. Ann.*, 35(1):61–97, 2003.

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

[7] J.-D. Benamous and Y. Brenier. A numerical method for the optimal time-continuous mass transport problem and related problems. In Luis A. Caffarelli and Mario Milman, editors, *Monge Ampère Equation: Applications to Geometry and Optimization (Deerfield Beach, FL)*, volume 226 of *Contemporary Mathematics*, pages 1–11. American Mathematics Society, Providence, RI, 1999.

[8] Nicolas Bonnotte. From Knothe's rearrangement to Brenier's optimal transport map. *SIAM Journal on Mathematical Analysis*, 45(1):64–87, 2013.

[9] Yann Brenier. Polar factorization and monotone rearrangement of vector-valued functions. *Communications on Pure and Applied Mathematics*, 44(4):375–417, 1991.

[10] Luis A. Caffarelli. Some regularity properties of solutions of Monge-Ampère equation. *Comm. Pure Appl. Math.*, 44 (8-9):965–969, 1991.

[11] Li Cui, Xin Qi, Chengfeng Wen, Na Lei, Xinyuan Li, Min Zhang, and Xianfeng Gu. Spherical optimal transportation. *Computer-Aided Design*, 115:181–193, 2019.

[12] Marco Cuturi. Sinkhorn distances: Lightspeed computation of optimal transport. In C. J. C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 26*, pages 2292–2300. Curran Associates, Inc., 2013.

[13] Bin Dai and David Wipf. Diagnosing and enhancing VAE models. In *International Conference on Learning Representations*, 2019.

[14] Fernando De Goes, Katherine Breeden, Victor Ostromoukhov, and Mathieu Desbrun. Blue noise through optimal transport. *ACM Transactions on Graphics (TOG)*, 31(6):171, 2012.

[15] Fernando De Goes, David Cohen-Steiner, Pierre Alliez, and Mathieu Desbrun. An optimal transport approach to robust reconstruction and simplification of 2D shapes. In *Computer Graphics Forum*, volume 30, pages 1593–1602. Wiley Online Library, 2011.

[16] Ayelet Dominitz and Allen Tannenbaum. Texture mapping via optimal mass transport. *Visualization and Computer Graphics, IEEE Transactions on*, 16(3):419–433, 2010.

[17] Jeff Donahue and Karen Simonyan. Large scale adversarial representation learning. *arXiv preprint* arXiv:1907.02544, 2019.

[18] Ian Goodfellow. Nips 2016 tutorial: Generative adversarial networks. *arXiv preprint* arXiv:1701.00160, 2016.

[19] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *NIPS*, pages 2672–2680, 2014.

[20] David Xianfeng Gu, Feng Luo, Jian Sun, and Shing-Tung Yau. Variational principles for Minkowski type problems, discrete optimal transport, and discrete Monge-Ampère equations. *Asian Journal of Mathematics*, 2016.

[21] Pengfei Guan, Xu-Jia Wang, et al. On a Monge-Ampère equation arising in geometric optics. *J. Diff. Geom.*, 48(48):205–223, 1998.

[22] Ishaan Gulrajani, Faruk Ahmed, Martin Arjovsky, Vincent Dumoulin, and Aaron C. Courville. Improved training of Wasserstein gans. In *NIPS*, pages 5769–5779, 2017.

[23] Cristian E. Gutiérrez and Qingbo Huang. The refractor problem in reshaping light beams. *Archive for rational mechanics and analysis*, 193(2):423–443, 2009.

[24] Sylvain Gelly, Bernhard Schoelkopf, Ilya Tolstikhin, and Olivier Bousquet. Wasserstein auto-encoders. In *ICLR*, 2018.

[25] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros. Image-to-image translation with conditional adversarial networks. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2017.

[26] Unnat Jain, Ziyu Zhang, and Alexander G. Schwing. Creativity: Generating diverse questions using variational autoencoders. In *CVPR*, pages 5415–5424, 2017.

[27] Trevor Darrell, Jeff Donahue, and Philipp Krähenbühl. Adversarial feature learning. In *International Conference on Learning Representations*, 2017.

[28] Janine Thoma, Dinesh Acharya, Luc Van Gool, Jiqing Wu, and Zhiwu Huang. Wasserstein divergence for gans. In *ECCV*, 2018.

[29] Leonid Vitalevich Kantorovich. On a problem of Monge. *Journal of Mathematical Sciences*, 133(4):1383–1383, 2006.

[30] Leonid Vitalevich Kantorovich. On a problem of Monge. *Uspekhi Mat. Nauk.*, 3:225–226, 1948.

[31] Diederik P. Kingma and Max Welling. Auto-encoding variational Bayes. *arXiv preprint* arXiv:1312.6114, 2013.

[32] Anders Boesen Lindbo Larsen, Søren Kaae Sønderby, Hugo Larochelle, and Ole Winther. Autoencoding beyond pixels using a learned similarity metric. 2016.

[33] Christian Ledig, Lucas Theis, Ferenc Huszar, Jose Caballero, Andrew Cunningham, Alejandro Acosta, Andrew Aitken, Alykhan Tejani, Johannes Totz, Zehan Wang, and Wenzhe Shi. Photo-realistic single image super-resolution using a generative adversarial network. 2017.

[34] Na Lei, Dongsheng An, Yang Guo, Kehua Su, Shixia Liu, Zhongxuan Luo, Shing-Tung Yau, and Xianfeng Gu. A geometric understanding of deep learning. *Engineering*, 6(3):361–374, 2020.

[35] Na Lei, Kehua Su, Li Cui, Shing-Tung Yau, and Xianfeng David Gu. A geometric view of optimal transportation and generative model. *Computer Aided Geometric Design*, 68:1–21, 2019.

[36] Zinan Lin, Ashish Khetan, Giulia Fanti, and Sewoong Oh. Pacgan: The power of two samples in generative adversarial networks. In *Advances in Neural Information Processing Systems*, pages 1505–1514, 2018.

[37] Huidong Liu, Xianfeng Gu, and Dimitris Samaras. Wasserstein gan with quadratic transport cost. In *ICCV*, 2019.

[38] Xi-Nan Ma, Neil S. Trudinger, and Xu-Jia Wang. Regularity of potential functions of the optimal transportation problem. *Arch. Ration. Mech. Anal.*, 177(2):151–183, 2005.

[39] Alireza Makhzani, Jonathon Shlens, Navdeep Jaitly, Ian Goodfellow, and Brendan Frey. Adversarial autoencoders. *arXiv preprint* arXiv:1511.05644, 2015.

[40] Quentin Mérigot. A multiscale approach to optimal transport. In *Computer Graphics Forum*, volume 30, pages 1583–1592. Wiley Online Library, 2011.

[41] Lars M. Mescheder, Sebastian Nowozin, and Andreas Geiger. Which training methods for gans do actually convergence? In *International Conference on Machine Learning (ICML)*, 2018.

[42] Jocelyn Meyron, Quentin Mérigot, and Boris Thibert. Light in power: a general and parameter-free algorithm for caustic design. In *SIGGRAPH Asia 2018 Technical Papers*, page 224. ACM, 2018.

[43] Takeru Miyato, Toshiki Kataoka, Masanori Koyama, and Yuichi Yoshida. Spectral normalization for generative adversarial networks. In *ICLR*, 2018.

[44] Saad Nadeem, Zhengyu Su, Wei Zeng, Arie E Kaufman, and Xianfeng Gu. Spherical parameterization balancing angle and area distortions. *IEEE Trans. Vis. Comput. Graph.*, 23(6):1663–1676, 2017.

[45] Alec Radford, Luke Metz, and Soumith Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks. In *ICLR*, 2016.

[46] Ali Razavi, Aaron Oord, and Oriol Vinyals. Generating diverse high-fidelity images with vq-vae-2. In *ICLR 2019 Workshop DeepGenStruct*, 2019.

[47] Danilo Jimenez Rezende, Shakir Mohamed, and Daan Wierstra. Stochastic backpropagation and approximate inference in deep generative models. *arXiv preprint* arXiv:1401.4082, 2014.

[48] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition, 2014.

[49] 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 Transactions on Graphics (TOG)*, 34(4):66, 2015.

[50] Justin Solomon, Raif Rustamov, Leonidas Guibas, and Adrian Butscher. Earth mover's distances on discrete surfaces. *ACM Transactions on Graphics (TOG)*, 33(4):67, 2014.

[51] Kehua Su, Wei Chen, Na Lei, Li Cui, Jian Jiang, and Xianfeng David Gu. Measure controllable volumetric mesh parameterization. *Comput. Aided Des.*, 78(C):188–198, September 2016.

[52] Kehua Su, Wei Chen, Na Lei, Junwei Zhang, Kun Qian, and Xianfeng Gu. Volume preserving mesh parameteri-

zation based on optimal mass transportation. *Comput. Aided Des.*, 82:42–56, 2017.
[53] Kehua Su, Li Cui, Kun Qian, Na Lei, Junwei Zhang, Min Zhang, and Xianfeng David Gu. Area-preserving mesh parameterization for poly-annulus surfaces based on optimal mass transportation. *Comput. Aided Geom. Des.*, 46(C):76–91, August 2016.
[54] Zhengyu Su, Wei Zeng, Rui Shi, Yalin Wang, Jian Sun, and Xianfeng Gu. Area preserving brain mapping. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2235–2242, 2013.
[55] Tauseef ur Rehman, Eldad Haber, Gallagher Pryor, John Melonakos, and Allen Tannenbaum. 3D nonrigid registration via optimal mass transport on the gpu. *Medical image analysis*, 13(6):931–940, 2009.
[56] Cédric Villani. *Optimal transport: old and new*, volume 338. Springer Science & Business Media, 2008.
[57] Xu-Jia Wang. On the design of a reflector antenna. *Inverse problems*, 12(3):351, 1996.
[58] Xu-Jia Wang. On the design of a reflector antenna ii. *Calculus of Variations and Partial Differential Equations*, 20(3):329–341, 2004.
[59] Chang Xiao, Peilin Zhong, and Changxi Zheng. BourGAN: Generative networks with metric embeddings. In *NeurIPS*, 2018.
[60] Shing-Tung Yau. *SS Chern: a great geometer of the twentieth century*. International PressCo, 1998.
[61] Xiaokang Yu, Na Lei, Xiaopeng Zheng, and Xianfeng Gu. Surface parameterization based on polar factorization. *J. Comput. Appl. Math.*, 329(C):24–36, February 2018.
[62] Ruslan Salakhutdinov, Yuri Burda, Roger Grosse. Importance weighted autoencoders. In *ICML*, 2015.
[63] Alessio Figalli. Regularity properties of optimal maps between nonconvex domains in the plane. *Comm. Partial Differential Equations*, 35(3):465–479, 2010.