AE-OT: A new Generative Model Based on Extended Semi-Discrete Optimal Transport

Dongsheng An¹, Yang Guo¹, Na Lei², Zhongxuan Luo², Shing-Tung Yau³, and Xianfeng Gu¹

¹Department of Computer Science, Stony Brook University
²DUT-RU Co-Research Center of Advanced ICT for Active Life, Dalian University of Technology
³Department of Mathematics, Harvard University
{doan, yangguo}@cs.stonybrook.edu, {nalei, zxluo}@dlut.edu.cn, yau@math.harvard.edu, gu@cs.stonybrook.edu

Abstract

Current generative models like generative adversarial networks (GANs) and variational autoencoders (VAEs) have attracted huge attention due to its capability to generate visual realistic images. However, most of the existing models suffer from the mode collapse or mode mixture problems. In this work, we give a theoretic explanation of the both problems by Figalli’s regularity theory of optimal transportation maps. Basically, the generator compute the transportation maps between the white noise distributions and the data distributions, which are in general discontinuous. However, deep neural networks (DNNs) can only represent continuous maps. This intrinsic conflict induces mode collapse and mode mixture. In order to tackle the both problems, we explicitly separate the manifold embedding and the optimal transportation; the first part is carried out using an autoencoder (AE) to map the images onto the latent space; the second part is accomplished using a GPU-based convex optimization to find the discontinuous transportation maps. Composing the extended optimal transport (OT) map and the decoder, we can finally generate new images from the white noise. This AE-OT model avoids representing discontinuous maps by DNNs, therefore effectively prevents mode collapse and mode mixture.

1 Introduction

Generative adversarial networks (GANs) [Goodfellow et al. (2014)] and variational autoencoders (VAEs) [Kingma & Welling (2013)] emerge as the dominant approaches for unconditional image generation. When trained on appropriate datasets, they are able to produce realistic and visual appealing samples. GAN methods train an unconditional generator that regresses real images from random noise and a discriminator that measures the difference between generated samples and real images. Despite GANs’ advantages, they have critical drawbacks. 1) Training of GANs are tricky and sensitive to hyperparameters. 2) GANs suffer from mode collapse, in which the generator only learns to generate few modes of data distribution while missing others, although samples from the missing modes occur throughout the training data (see e.g. Goodfellow (2016)). While for the VAEs, the encoder is used to map the data distribution to a Gaussian latent distribution, which is then mapped back to the data distribution by the decoder. While standard VAEs tend to capture all modes, they often generate ambiguous images on multimodal real data distributions. We propose that these phenomena relates deeply with the singularities of distribution transport maps.

Manifold Distribution Hypothesis In deep learning, the manifold distribution hypothesis is well accepted, which assumes the distribution of a specific class of natural data is concentrated on a low dimensional manifold embedded in the high dimensional data space [Tenenbaum et al. (2000)]. Therefore, GANs and VAEs implicitly aim to accomplish two major tasks: 1) manifold embedding: to find the encoding/decoding maps between the data manifold embedded in the image space and the latent space; 2) probability distribution transport: to transport a given white noise distribution to the data distribution, either in the latent or in the image space.
Figure 1: Mode collapse/mixture caused by the discontinuity of the transport map. Top row shows real data distributions, the bottom row gives the noise distributions. On top, each cluster represents a mode, the spurious generated samples are red crosses (mode mixture); at the bottom, red dotted lines are the singularity set, red crosses are mapped to be spurious samples by DNNs. (a) If the support of the target distribution is convex, DNN \( f_1 \) is able to approximate the transport map \( \hat{f}_1 \) well. When the support of the target distributions are concave, there are two situations: (b) single mode and (c) multi modes. In (b), DNN, represented by \( f_2 \), cannot approximate the transport map \( \hat{f}_2 \) well and generates some spurious samples. \( \hat{f}_3 \) gives the transport map of multi-mode, when approximating it with continuous DNNs, either mode collapse \( f_{31} \) or mode mixture \( f_{32} \) will happen.

**Distribution Transformation** The generator of GAN model and the decoder of VAE model are trained to compute a transport map that transforms a known continuous distribution (e.g. Gaussian white noise) to the real data distribution. Namely, the transport map pushes forward the given noise distribution to a generated distribution to approximate the real data distribution, the similarity between the two distributions determines the generalization ability of the generator [Ben-David et al. (2010)].

**Discontinuity and Mode Collapse/Mixture** It is a common practice among GAN/VAE models that the generators/decoders are expressed by deep neural networks, which can only represent continuous mappings. Unfortunately, as pointed out by works [Nagarajan & Kolter (2017); Khayatkhooi et al. (2018); Xiao et al. (2018)], the transport maps may be discontinuous when there are multiple modes in the data distribution. This intrinsic conflict can cause mode collapse or mode mixture. The later means that the generated samples are mixtures of multiple modes and look spurious or ambiguous. Furthermore, even when the real data distribution has a single mode, ambiguous data (e.g. a human face image with mismatched eye colors) can still present. This can be explained by Brenier’s polar factorization theorem [Brenier (1991b; 1987; 1991a) and Figalli’s regularity theorem [Figalli (2010); Chen & Figalli (2017)] (Thm. 5 in Appendix B), which asserts that if the support of the target distribution is not convex, then there will be singularity sets on the support of the source distribution, such that the transport map is discontinuous on these sets. This shows the intrinsic difficulties of conventional GANs/VAEs cannot be eliminated, as shown in Fig. 1.

**Conquering Mode Collapse/Mixture** However, according to [Brenier (1987; 1991a)] theorem, the optimal transport map can be represented as the gradient map of the Brenier potential. At the regular points, the Brenier potential is differentiable, its gradient map (the transport map) is continuous; at the singularities, the Brenier potential is continuous but not differentiable, and its gradient map is discontinuous. Conventional GANs and VAEs model the gradient map directly and encounter the trouble of discontinuity. In contrast, we propose to model the globally continuous Brenier potential to avoid mode collapse/mixture.

More specifically, our proposed AE-OT model separates the manifold embedding step and the probability distribution transformation step, the former is carried out by an autoencoder (AE), the latter is accomplished by a convex optimization framework (OT). The OT step computes the Brenier potential explicitly and is able to locate the singularity set (the discontinuous points of the gradient map) based on Figalli’s theory. Our experimental results demonstrate that the proposed method can not only cover all of the modes, but also avoid generating spurious samples (mode mixture).

**Contributions** (i) From theoretical aspect, this work gives a thorough explanation of mode collapse and mode mixture by the regularity theory of optimal transportation developed by Figalli (2018)}

---

1For example, a generator generates obscure digits mixing 0 and 8 but neither 0 nor 8 on the MNIST dataset.
Fields medalist) and the reasons why standard GANs/VAEs cannot solve this problem perfectly. (ii) From practical aspect, this work proposes a novel model called AE-OT, which first encodes the data manifold into the latent space, then compute the Brenier potential to represent the optimal transportation map in the latent space. The Figalli’s singularity set can be located efficiently and avoided when generating new samples. In this way, our model eliminates mode collapse and mode mixture successfully. (iii) The algorithm for finding the Brenier potential and the optimal transportation map can be accelerated with GPU based convex optimization algorithm. The method converges to the unique global optimum with bounded error estimate. (iv) Our experiment results demonstrate the efficiency and efficacy of the proposed method.

2 RELATED WORK

Optimal Transport Optimal transport plays an important role in various engineering fields. For more thorough reviews, we refer the readers to [Peyré & Cuturi (2018)] and [Solomon (2018)]. In [Gu et al. (2016)], the intrinsic connection between Brenier theory in OT and Alexandroff theory in convex geometry was established, and applied for deep learning in [Lei et al. (2017)] by a convex optimization. Figalli and the collaborators [Figalli (2010), Chen & Figalli (2017)] proposed that when the support of the data distribution is non-convex, the transport map will be discontinuous.

Generative models In machine learning, generative models have been becoming more important and popular recently. A huge breakthrough for image generating comes from the scheme of Variational Autoencoders (VAEs) (e.g. [Kingma & Welling (2013)]), where the decoders approximate real data distributions from a Gaussian distribution in a variational approach (e.g [Kingma & Welling (2013)] and [Rezende et al. (2014)]). Various recent works followed this scheme, including Adversarial Autoencoders (AAEs) [Makhzani et al. (2015)] and Wasserstein Autoencoders (WAEs) [tolstikhin et al. (2018)]. Although VAEs are relatively simple to train, images they generate look blurry. Generative Adversarial Networks (GANs) [Goodfellow et al. (2014)] were proposed to solve this disadvantage. While being a powerful tool in generating realistically looking samples, GANs can be hard to train and suffer from mode collapsing. Various improvements have been proposed for better training of GANs, including changing the loss function (e.g. Wasserstein GAN [Arjovsky et al. (2017)]), regularizing the discriminators to be Lipschitz (clipping [Arjovsky et al. (2017)], gradient regularization [Gulrajani et al. (2017)], Mescheder et al. (2018) or spectral normalization [Miyato et al. (2018)]).

Besides, various non-adversarial methods has also been proposed recently. GLO [Bojanowski et al. (2017)] employs an “encoder-less autoencoder” approach where a generative model is trained with a non-adversarial loss function. IMLE [Li & Malik (2018)] proposed an ICP related generative model training approach. Later GLANN [Hoshen & Malik (2019)] combines advantages of GLO and GLANN, where an embedding from image space to latent space was first found using GLO and then a transformation between an arbitrary distribution and latent code was computed using IMLE.

Mitigating Mode Collapsing Recently, [Nagarajan & Kolter (2017), Khayatkhoei et al. (2018), Xiao et al. (2018)] also realize the training difficulties of GANs come from the approximation of discontinuous functions with continuous DNNs. By the gradient-based regularization, GDGAN [Nagarajan & Kolter (2017)] do relieve the mode collapse phenomenon of GANs, but mode mixture still exists. [Khayatkhoei et al. (2018)] proposes to use multiple GANs to overcome the mode collapse. [Xiao et al. (2018)] proposed to embed the images into a latent space according to Bourgain’s theorem, and train the generator by sampling a Gaussian mixture distribution in the latent space instead of a unimodal Gaussian. The recently introduced normalized diversification by [Liu et al. (2018)] can also help overcome mode collapse successfully. However, all of them cannot solve the mode mixture well.

All these works [Nagarajan & Kolter (2017), Khayatkhoei et al. (2018), Xiao et al. (2018)] explain that if the target data distribution has multiple modes, the transport map is discontinuous, but DNNs can only represent continuous mappings, the intrinsic conflict causes mode collapse.

3 COMPUTATIONAL ALGORITHMS

Overview of AE-OT Model Our AE-OT model is summarized in Fig. 2, it has two major components: i) (AE) An autoencoder is trained to encode \( f_\theta \) the data manifold from the image space \( \mathcal{X} \) to the latent space \( \mathcal{Z} \), and map the data distribution to the latent code distribution; then the decoder \( g_\xi \) decodes the latent code back to the data manifold. ii) (OT) This module computes the optimal transportation map \( T \) from the noise distribution to the latent code distribution. First, the Brenier potential is found by a convex optimization process according to [Gu et al. (2016)], whose gradient is
When the cost function is the $w$-distance,

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

$i = 1, 2, \ldots, n,$ with the equal total mass as the source measure, $\mu(\Omega) = \sum_{i=1}^{n} \nu_i$. Under a semi-discrete transport map $T : \Omega \rightarrow Y$, a cell decomposition is induced $\Omega = \bigcup_{i=1}^{n} W_i$, such that every $x$ in each cell $W_i$ is mapped to the target $y_i$, $T : x \in W_i \mapsto y_i$. The map $T$ is measure preserving, denoted as $T_{#} \mu = \nu$, if the $\mu$-volume of each cell $W_i$ equals to the $\nu$-measure of the image $T(W_i) = y_i$, $\mu(W_i) = \nu_i$. The cost function is given by $c : \Omega \times Y \rightarrow \mathbb{R}$, where $c(x, y)$ represent the cost for transporting a unit mass from $x$ to $y$. The total cost of $T$ is given by $\int_{\Omega} c(x, T(x))d\mu(x) = \sum_{i=1}^{n} \int_{W_i} c(x, y_i)d\mu_i(x)$. **Semi-discrete optimal transport map** is the measure-preserving map that minimizes the total cost, $T^* := \arg \min_{T_{#} \mu = \nu} \int_{\Omega} c(x, T(x))d\mu(x)$.

When the cost function is the $L^2$ distance $c(x, y) = 1/2 \| x - y \|^2$, Brenier’s theorem claims that the semi-discrete OT map is given by the gradient map of a piece-wise (PL) convex function, the so-called Brenier potential $u_h : \Omega \rightarrow \mathbb{R}$, $u_h(x) := \max_{i=1}^{n} \{ \pi_{h,i}(x) \}$, where $\pi_{h,i}(x) = \langle x, y_i \rangle + h_i$ is the hyperplane corresponding to $y_i \in Y$. As shown in Fig. 3(a), the projection of the graph of $u_h$ decomposes $\Omega$ into cells $W_i(h)$, each cell $W_i(h)$ is the projection of the supporting plane $\pi_{h,i}(x)$. The height vector $h$ is the unique optimizer of the following convex energy under the condition that $\sum_i h_i = 0$,

$$E(h) = \int_{\Omega} \sum_{i=1}^{n} w_i(\eta)d\eta_i - \sum_{i=1}^{n} h_i \nu_i,$$

(1)

where $w_i(\eta)$ is the $\mu$-volume of $W_i(\eta)$. The convex energy $E(h)$ can be optimized simply by gradient descend method with $\nabla E(h) = (w_i(h) - \nu_i)^T$.

The key is to compute the $\mu$-volume $w_i(h)$ of each cell $W_i(h)$, which can be estimated using conventional Monte Carlo method. We draw $N$ random samples from $\mu$ distribution, $\{ x_j \} \sim \text{i.i.d. } \mu$, $\forall j \in \mathcal{J}$, the estimated $\mu$-volume of each cell is $\hat{w}_i(h) = \# \{ j \in \mathcal{J} \mid x_j \in W_i(h) \} / N$. Given $x_j$, we can find $W_i$ in which $x_j \in W_i$ by $i = \arg \max_{i=1}^{n} \{ \langle x_j, y_i \rangle + h_i \}, i = 1, 2, \ldots, n$. When $N$ is large enough, $\hat{w}_i(h)$ converges to $w_i(h)$. Then the gradient of the energy is approximated as $\nabla E \approx (\hat{w}_i(h) - \nu_i)^T$. Once the gradient is estimated, we can use Adam algorithm [Kingma & Ba](https://arxiv.org/abs/1412.6980)
Algorithm 1 Semi-discrete OT Map

1: Input: Latent codes \( Y = \{y_i\}_i \in X \), empirical latent code distribution \( \nu = \frac{1}{|X|} \sum_{i \in X} \delta_{y_i} \), number of Monte-Carlo samples \( N \), positive integer. 
2: Output: Optimal transport map \( T(\cdot) \).
3: Initialize \( h = (h_1, h_2, \ldots, h_{|Z|}) \leftarrow (0, 0, \ldots, 0) \).
4: repeat
5: Generate \( N \) uniformly distributed samples \( \{x_j\}_{j=1}^N \).
6: Compute \( \nabla h = (\nu(x_j) - \nu(x_i))^T \).
7: Compute \( \nabla h = \nabla h - \text{mean}(\nabla h) \).
8: Update \( h \) by Adam algorithm with \( \beta_1 = 0.9, \beta_2 = 0.5 \).
9: if \( E(h) \) has not decreased for \( s \) steps then
10: \( N \leftarrow N \times 2 \).
11: end if
12: until Converge
13: OT map \( T(\cdot) \leftarrow \nabla (\max_{y_i \in Y} \langle \cdot, y_i \rangle + h_i) \).

Algorithm 2 Generate latent code

1: Input: Optimal transport map \( T(\cdot) \), number of samples to generate \( n \), angle threshold \( \theta \).
2: Output: Generated latent code \( P \).
3: Compute \( \hat{c}_i \) by Monte Carlo method.
4: repeat
5: Sample \( x \sim \mu \). Find the smallest \( d + 1 \) vertex around \( x \) as \( \{d(x, \hat{c}_{i_0}), d(x, \hat{c}_{i_1}), \ldots, d(x, \hat{c}_{i_d})\} \).
6: Compute dihedral angles \( \theta_{ik} \) between \( \pi_{i_0} \) and \( \pi_{i_k} \).
7: Select \( \theta_{ik} \) with \( \theta_{ik} \leq \theta \), result in \( i_k = 0, 1, \ldots, d_1 \).
8: if \( \forall k, \theta_{ik} \geq \theta \) then Abandon \( x \).
9: else Generate latent code \( \hat{T}(x) = \sum_{k=0}^{d_1} \lambda_k T(\hat{c}_{i_k}) \) with \( \lambda_k = d^{-1} \langle x, \hat{c}_{i_k} \rangle / \sum_{j=0}^{d_1} d^{-1} \langle x, \hat{c}_{i_j} \rangle \).
10: end if
11: until Generate \( n \) new latent code

Piece-wise Linear Extension The semi-discrete OT map \( \nabla u_h : \Omega \rightarrow Y \) maps all \( x \in \Omega \) to the latent codes of training samples \( \{y_i\}_i \)’s and won’t generate new samples. Therefore, we extend the semi-discrete OT map \( T = \nabla u_h \) to a piecewise linear (PL) mapping \( \hat{T} \) as follows. The projection of \( u_h \) in the source domain induces a cell decomposition of \( \Omega \), of which each cell is of \( \mu \)-volume \( \nu_t \) and is mapped to the corresponding \( y_i \). By representing the cells by their \( \mu \)-mass centers as \( c_i := \int_{W_i} \mu(x) \, dx \), we can get the point-wise map \( i : c_i \mapsto y_i \). The Poincaré of the cell decomposition induces a triangulation of the centers \( C = \{c_i\} \): if \( W_{i_1} \cap W_{i_2} \neq \emptyset \), then \( c_i \) is connected with \( c_j \) to form an edge \( [c_i, c_j] \). Similarly, if \( W_{i_1} \cap W_{i_2} \cdots \cap W_{i_k} \neq \emptyset \), then there is a \( k \)-dimensional simplex \([c_{i_{0}}, c_{i_{1}}, \ldots, c_{i_{k}}]\). All these simplices form a triangulation of \( C \) (a simplicial complex), denoted as \( \mathcal{T}(C) \) (the green triangles in the left of Fig. 3b)). We can triangulate \( Y \) in the same way to obtain the triangulation \( \mathcal{T}(Y) \) (the green triangles in the right of Fig. 3b)). Once a random sample \( x \) is drawn from the distribution \( \mu \), we can find the simplex \( \sigma \) in \( \mathcal{T}(C) \) containing \( x \). Assume the simplex \( \sigma \) has \( d + 1 \) vertices \( \{c_{i_{0}}, c_{i_{1}}, \ldots, c_{i_{k}}\} \), the bary-centric coordinates of \( x \) in \( \sigma \) is defined as \( x = \sum_{k=0}^{d} \lambda_k c_{i_k} \), and \( \sum_{k=0}^{d} \lambda_k = 1 \) with all \( \lambda_k \) non-negative. Then the generated latent code of \( x \)
under this piece-wise linear map is given by $\hat{T}(x) = \sum_{k=0}^{d} \lambda_k y_{i_k}$ (In Fig. 3(b), the green dot $x_1$ is mapped to be $\hat{T}(x_1)$). Because all of the $y$'s are used to construct the simplicial complex $\mathcal{T}(Y)$ in the support of the target distribution, we can guarantee that no mode is lost.

In practice, the $\mu$-mass center $c_i$ is approximated by the mean value of all the Monte-Carlo samples inside $W_i(h), \hat{c}_i = \sum_{x_j \in W_i} x_j / \# \{ x_j \in W_i \}$, where $x_j \sim \mu$. The connectivity information $\mathcal{T}(C)$ is too complicated to construct and to store in high dimensional space, thus $\mathcal{T}(C)$ is not explicitly built. Instead, we find the simplex $\sigma \in \mathcal{T}(C)$ containing $x$ as follows: given a random point $x \in \Omega$, evaluate and sort its Euclidean distances to the centers $d(x, \hat{c}_i), i = 1, 2, \ldots, n$ in the ascending order. Suppose the first $d + 1$ items are $\{d(x, \hat{c}_{i_0}), d(x, \hat{c}_{i_1}), \ldots, d(x, \hat{c}_{i_d})\}$, then $\sigma$ is formed by $\{\hat{c}_{i_k}\}$. The bary-centric coordinates $\tilde{\lambda}_i$ are estimated as $\tilde{\lambda}_i = d^{-1}(x, \hat{c}_{i_k}) / \sum_{k=0}^{d} d^{-1}(x, \hat{c}_{i_k})$. However, this may generate some spurious samples. To overcome it, we need further to detect the singular set.

**Singular Set Detection** According to Figalli’s theory [Figalli (2010); Chen & Figalli (2017)], if there are multiple modes or the support of the target distribution is concave, the middle potential $\mu$ is prominently large. Therefore, on the graph of Brenier potential, we pick the pairs of facets $h_j \sim (\hat{c}_{i_k})$, which means that different vertices of the polyhedron belongs to different mode. If the dihedral angles are larger than a given threshold, the projection of their intersection gives a smooths the discrete function $\hat{T}(\cdot)$ in regions where latent codes are dense and keep the discontinuity of $\hat{T}(\cdot)$ where latent codes are very sparse. In this way we avoid generating spurious latent code and thus improve the generation quality. The algorithm to generate new code is shown in Alg. 2 and the effect of threshold filtering is further investigated in Appendix C.1.

4 Experiments

In order to validate that the proposed method can solve the mode collapse/mixture problems and generate controllable high quality images, several experiments are conducted.

The first experiment focuses on toy sets, so that the complexity of the tasks can be manually controlled and the mode and quality of the generated samples can be accurately computed. [Lin et al. (2018)] did a large-scale comparison with previous methods that explicitly proposed to mitigate mode collapse and thus established a baseline for comparison. For consistent evaluation, we set up our experiment on the same benchmark dataset as theirs, and make the comparison.

In the second experiment, we run the proposed method mainly on 4 public datasets, MNIST [LeCun & Cortes (2010)], MNIST-FANSION [Han Xiao & Volggraf (2017), CIFAR-10 [Krizhevsky (2009)] and CelebA [Zhang et al. (2018)], just like the authors of [Hoshen & Malik (2019); Sajjadi et al. (2018); Lucic et al. (2018)] did in their papers. Besides, the architecture of the decoder is the same as [Lucic et al. (2018)], in which the authors did a large-scale study to evaluated the best performance of 8 different generative models including various GAN models and VAE, and the encoder is set to be
We report our results in Tab. (2), in which the compared data comes from Lucic et al. (2018) Hoshen & Malik (2019). In general, the proposed model achieves better than or comparable scores to other models. The autoencoder architecture we use here cannot find a good encoding for the CelebA dataset due to its limited capacity. But the FID score of the generation model is still approach to the autoencoder. The mirror of decoder. The training details, parameter setting and time consuming information are introduced in Section C.5 of the Appendix.

<table>
<thead>
<tr>
<th>Modes (Max 8)</th>
<th>2D-ring</th>
<th>Modes (Max 25)</th>
<th>2D-grid</th>
</tr>
</thead>
<tbody>
<tr>
<td>GAN</td>
<td>6.3±0.5</td>
<td>98.2±0.2%</td>
<td>0.45±0.09</td>
</tr>
<tr>
<td>ALI</td>
<td>6.6±0.3</td>
<td>97.6±0.4%</td>
<td>0.36±0.04</td>
</tr>
<tr>
<td>MD</td>
<td>4.3±0.8</td>
<td>36.6±8.8%</td>
<td>1.93±0.11</td>
</tr>
<tr>
<td>PacGAN2</td>
<td>7.9±0.1</td>
<td>95.6±2.0%</td>
<td>0.07±0.03</td>
</tr>
<tr>
<td>PacGAN3</td>
<td>7.8±0.1</td>
<td>97.7±0.3%</td>
<td>0.10±0.02</td>
</tr>
<tr>
<td>PacGAN4</td>
<td>7.8±0.1</td>
<td>95.9±1.4%</td>
<td>0.07±0.02</td>
</tr>
<tr>
<td>BourGAN</td>
<td>8.0±0.0</td>
<td>99.8±2.9%</td>
<td>4e-4±2e-4</td>
</tr>
<tr>
<td>AE-OT</td>
<td>8.0±0.0</td>
<td>99.6±0.3%</td>
<td>0.004±0.001</td>
</tr>
</tbody>
</table>

The autoencoder architecture we use here cannot find a good encoding for the CelebA dataset due to its limited capacity. But the FID score of the generation model is still approach to the autoencoder.
<table>
<thead>
<tr>
<th>Dataset</th>
<th>NS GAN</th>
<th>LSGAN</th>
<th>WGAN</th>
<th>BEGAN</th>
<th>VAE</th>
<th>GLO</th>
<th>GLANN</th>
<th>AE</th>
<th>Ours</th>
</tr>
</thead>
<tbody>
<tr>
<td>MNIST</td>
<td>6.8</td>
<td>7.8</td>
<td>6.7</td>
<td>13.1</td>
<td>23.8</td>
<td>49.6</td>
<td>8.6</td>
<td>5.5</td>
<td>6.4</td>
</tr>
<tr>
<td>Fansion</td>
<td>26.5</td>
<td>30.7</td>
<td>21.5</td>
<td>22.9</td>
<td>58.7</td>
<td>57.7</td>
<td>13.0</td>
<td>4.7</td>
<td>10.2</td>
</tr>
<tr>
<td>CIFAR-10</td>
<td>58.5</td>
<td>87.1</td>
<td>55.2</td>
<td>71.4</td>
<td>155.7</td>
<td>65.4</td>
<td>46.5</td>
<td>28.2</td>
<td>38.1</td>
</tr>
<tr>
<td>CelebA</td>
<td>55.0</td>
<td>53.9</td>
<td>41.3</td>
<td><strong>38.9</strong></td>
<td>85.7</td>
<td>52.4</td>
<td>46.3</td>
<td>67.5</td>
<td>68.4</td>
</tr>
</tbody>
</table>

Precision and recall proposed in Sajjadi et al. (2018) can compute the precision and the recall at the same time only given the same number of generated and reference images. In Section C.6, we report the comparison results with state-of-the-art methods.

5 CONCLUSION

This work gives a theoretic explanation for mode collapse/mixture by Brenier’s theory and Figalli’s regularity theory of optimal transport maps. When the target measure has concave support, the OT map is discontinuous on the singular sets. But DNNs can only represent continuous functions, this conflict causes the both problems. In order to solve this problem, the AE-OT model is proposed by separating manifold embedding and measure transformation. The former step is computed using an autoencoder, the latter is carried out using the extended semi-discrete OT map based on GPUs. The model is tested thoroughly and extensively by both synthetic and real data sets. The experimental
results validates the discontinuity of the OT maps and demonstrate the advantages comparing to the state-of-the-arts.

Acknowledgements

The project is partially supported by the National Natural Science Foundation of China (61936002, 61772105, 61720106005), NSF CMMI-1762287 collaborative research: computational framework for designing conformal stretchable electronics, Ford URP topology optimization of cellular mesostructures’ nonlinear behaviors for crash safety, and NSF DMS-1737812 collaborative research: ATD: theory and algorithms for discrete curvatures on network data from human mobility and monitoring.

REFERENCES


A Brenier’s Theory

In this subsection, we briefly introduce the basic concepts and theorems in optimal transport theory, which comes from Brenier theory Villani (2008); Brenier (1987; 1991a), and discrete theory Gu et al. (2016).

Optimal transport Problem Suppose $X, Y \subset \mathbb{R}^d$ are two subsets of $n$-dimensional Euclidean space, $\mu, \nu$ are two probability measures defined on $X$ and $Y$ respectively, with equal total measure, $\mu(X) = \nu(Y)$. A map $T : X \rightarrow Y$ is measure preserving, denoted as $T#\mu = \nu$, if for any measurable set $B \subset Y$, $\mu(T^{-1}(B)) = \nu(B)$. Given a cost function $c(x, y) : X \times Y \rightarrow \mathbb{R}_{\geq 0}$, indicating the cost of moving each unit mass from the source to the target, the total transport cost of the map $T$ is defined to be $\int_X c(x, T(x))d\mu(x)$.

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

$$\text{(MP) } \mathcal{W}_c(\mu, \nu) := \min_{T#\mu = \nu} \int_X c(x, T(x))d\mu(x).$$

The solutions to the Monge’s problem is called the optimal transport map, whose total transport cost is called the Wasserstein distance between $\mu$ and $\nu$, denoted as $\mathcal{W}_c(\mu, \nu)$.

Brenier’s Approach Brenier (1987; 1991a) proved the following theorem:

**Theorem 1** (Brenier (1987; 1991b)). Suppose $X$ and $Y$ are the Euclidean space $\mathbb{R}^d$ and the transport cost is the quadratic Euclidean distance $c(x, y) = 1/2\|x - y\|^2$. Furthermore $\mu$ is absolutely continuous and $\mu$ and $\nu$ have finite second order moments, then there exists a convex function $u : X \rightarrow \mathbb{R}$, the so-called Brenier potential, its gradient map $\nabla u$ gives the solution to the Monge’s problem. The Brenier potential is unique up to a constant.

Brenier’s polar factorization theorem claims that: for any measure preserving map $T#\mu = \nu$, $T$ can be uniquely decomposes into the forms $T = \nabla u \circ s$, where $s : X \rightarrow X$ is a volume preserving map and $\nabla u$ is the optimal transport map under $L^2$ cost. Therefore, the regularity of $T$ can be determined by that of $\nabla u$.

Discrete Brenier’s Theorem Brenier theorem can be directly generalized to discrete target measure. Suppose the source measure $\mu$ is defined on a compact convex set $\Omega$, the target measure $\nu = \sum_{i=1}^{n} \nu_i \delta(y - y_i)$, $\mu(\Omega) = \sum_{i=1}^{n} \nu_i$. The discrete Brenier potential is a piecewise linear function,

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

As shown in Fig. 3(a), the projection of the Brenier potential induces a cell decomposition of $\Omega$, each cell $W_i(h) := \{p \in \Omega | \nabla u_i(p) = y_i \}$, whose $\mu$-measure is denoted as $w_i(h)$.

**Theorem 2** (Discrete Brenier Theorem Gu et al. (2016)). For any $\nu_1, \nu_2, \ldots, \nu_n > 0$ with $\sum_{i=1}^{n} \nu_i = \mu(\Omega)$, there exists $h = (h_1, h_2, \ldots, h_n) \in \mathbb{R}^n$, unique up to adding a constant $(c, c, \ldots, c)$, so that $w_i(h) = \nu_i$ for all $i$. The vector $h$ is the unique minimum argument of the following convex energy

$$E(h) = \int_{\Omega} \sum_{i=1}^{n} w_i(\eta)d\eta - \sum_{i=1}^{n} h_i \nu_i,$$  

defined on an open convex set $\mathcal{H} = \{ h \in \mathbb{R}^n : w_i(h) > 0, i = 1, 2, \ldots, n \}$. Furthermore, $\nabla u_{h_i}$ minimizes the quadratic cost $\int_{\Omega} \|x - T(x)\|^2d\mu(x)$ among all transport maps $T#\mu = \nu$. The gradient of above energy is given by $\nabla E(h) = (w_1(h) - \nu_1, w_2(h) - \nu_2, \ldots, w_n(h) - \nu_n)^T$. The Hessian of the energy is given by

$$\frac{\partial w_i}{\partial h_j} = -\frac{\mu(W_i \cap W_j)}{\|y_i - y_j\|} \frac{\partial w_i}{\partial h_i} = \sum_{j \neq i} \frac{\partial w_i}{\partial h_j}.$$
Theorem 3 (Kitagawa-Mérigot-Thibert Kitagawa et al. (2019)). Assume the cost function is quadratic distance, \( \mu \) has convex support and also that (i) The probability density of \( \mu \) is \( C_{0, \alpha}^0(\Omega) \) for \( \alpha \) in \( (0; 1] \), (ii) \( \mu \) has positive Poincaré-Wirtinger constant. Then the Newton algorithm for semi-discrete optimal transport converges globally with linear rate and locally with rate \( 1 + \alpha \).

Though in our method the gradient descend method is applied, the above theorem also ensures its convergence because of the convexity of the energy function \( E(h) \) we adopted.

B Figalli’s Theory

In this section, we show the fact that even for the case of single mode, the transport map may still be discontinuous, which will cause the instability of the training process of GANs. The arguments are mainly based on the regularity theory of transport maps developed by Figalli [Chen & Figalli] (2017); Figalli (2010) and so on.

According to Brenier’s Theorem [Brenier (1987); 1991a], any transport map can be decomposed into a measure preserving map and a solution to the Monge-Ampére equation, which is the optimal transport map under the \( L^2 \) cost function. Therefore, the continuity of the transport map can be reduced to the regularity (smoothness) of the solution to the Monge-Ampére equation. When the support of the target measure is convex and the density functions are smooth, Caffarelli showed the map is differentiable; otherwise if the target domain is not convex, Figalli showed the map is discontinuous, and gave precise description of the singularity set. In this section, we briefly introduce Figallis’ theory, and conduct an experiment using CelebA data set to show the existence of the singularity set, hence demonstrate the fact that the transport maps computed in GANs are discontinuous.

B.1 Convex Domains - Caffarelli Theorem

Let \( \Omega \) and \( \Lambda \) are two bounded open sets in \( \mathbb{R}^n \), and let \( f: \mathbb{R}^n \to \mathbb{R} \) and \( g: \mathbb{R}^n \to \mathbb{R} \) be two positive functions such that \( f = 0 \) in \( \mathbb{R}^n \setminus \Omega \), \( g = 0 \) in \( \mathbb{R}^n \setminus \Lambda \), and

\[
\int_{\Omega} f = \int_{\Lambda} g = 1.
\]

According to Brenier’s Theorem [Brenier (1987); 1991a], there exists a globally Lipschitz convex function \( \varphi: \mathbb{R}^n \to \mathbb{R} \) such that \( \nabla \varphi \# f = g \) and \( \nabla \varphi(x) \in \tilde{\Lambda} \) for \( L^2 \)-a.e. \( x \in \mathbb{R}^n \). We say \( \varphi \) weakly solves the Monge-Ampére equation

\[
\det(D^2 \varphi) = \frac{f}{g \circ \nabla \varphi} \quad \text{in } \mathbb{R}^n,
\]

together with the boundary condition \( \nabla \varphi(\mathbb{R}^n) \subset \tilde{\Lambda} \). \( \varphi \) is called the Brenier potential.

As shown by Caffarelli [9], if \( \Lambda \) is convex, then \( \varphi \) is strictly convex, and it solves the Monge-Ampére equation[6]. The regularity theory has been established (see Caffarelli (1990a;b,1991)), such as

1. if \( \lambda \leq f, g \leq 1/\lambda \) for some \( \lambda > 0 \), then \( \varphi \in C_{loc}^{1, \alpha}(\Lambda) \).
2. if \( f \in C_{loc}^{k, \alpha}(\Omega) \) and \( g \in C_{loc}^{k, \alpha}(\Lambda) \), then \( \varphi \in C_{loc}^{k+2, \alpha}(\Omega) \), \( (k \geq 0, \alpha \in (0, 1)) \).

B.2 Non-convex Domains - Figalli Theorem

However, if \( \Lambda \) is not convex, the regularity of the Brenier potential can not be guaranteed. For example one can find an example, such that

1. \( \Omega \) is convex, \( \Lambda \) is simply connected, but non-convex;
2. the density functions \( f \) and \( g \) are smooth, \( f \in C^\infty(\Omega) \) and \( g \in C^\infty(\Lambda) \);
3. the Brenier potential \( \varphi \not\in C^1(\Omega) \), the transport map \( \nabla \varphi \) is not continuous.

In this scenario, the transport map can not be learned using DNNs, and training process is unstable or the GAN model generates unrealistic samples.
**Figalli’s construction**  Let $\varphi : \mathbb{R}^n \to \mathbb{R}$ be a convex function. Its subdifferential at a point $x$ is defined by
\[ \partial \varphi(x) := \{ y \in \mathbb{R}^n | \varphi(z) \geq \varphi(x) + y \cdot (z - x), \forall z \in \mathbb{R}^n \}. \]
$\varphi$ is differentiable at a point $x$ if and only if $\partial \varphi(x)$ is a singleton. Figalli decomposes the set of non-differentiability points according to the dimension of the singular set:
\[ \Sigma_k(\varphi) := \{ x \in \mathbb{R}^n | \text{dim}(\partial \varphi) = k \}, k = 0, \ldots, n. \] (7)
For any $k = 0, \ldots, n$, the set $\Sigma_k(\varphi)$ is $(n-k)$-rectifiable. The set of reachable subgradients at $x$ as
\[ \nabla_* \varphi := \left\{ \lim_{k \to +\infty} \nabla \varphi(x_k) | x_k \in \Sigma_0, x_k \to x \right\}. \]
It is known that the convex hull of $\nabla_* \varphi(x)$, coincides with $\partial \varphi(x)$.

**Theorem 4** (Figalli). Assume that there exists $\lambda > 0$ such that $\lambda \leq f \leq 1/\lambda$ in $\Omega$, $\lambda \leq g \leq 1/\lambda$ in $\Lambda$, and that $\partial \Omega$ and $\partial \Lambda$ are continuous. Then $\varphi$ is strictly convex inside $\Omega$. Moreover there exist two open sets $\Omega' \subset \Omega$ and $\Lambda' \subset \Lambda$, with $L^2(\Omega' \setminus \Omega') = L^2(\Lambda' \setminus \Lambda') = 0$, such that $\varphi \in C^{1,\alpha}(\Omega')$, $\nabla \varphi$ is a bi-Hölder homeomorphism between $\Omega'$ and $\Lambda'$, and $\varphi$ is an Alexandrov solution of $\Omega$ inside $\Omega'$. In particular, Caffarelli’s regularity theory for strictly convex Alexandrov solutions of the Monge-Ampère equations applies to $\varphi$ inside $\Omega'$.

Figalli studies the singular set of $\varphi$ in $\Omega$, i.e. the set of points $x \in \Omega$ where $\varphi$ is not differentiable, denoted as $\text{Sing}$. Figalli shows the singularity set has the following characterization,
\[ \text{Sing} = \{ x \in \Omega | \partial \varphi(x) \cap \Lambda = \emptyset, \nabla_* \varphi(x) \subset \partial \Lambda, \partial \varphi(x) \not\subset \Lambda \}. \]
it can be decomposed into connected components $\text{Sing} := \bigcup_i S_i$. For planar case, there are two cases.

**Theorem 5** (Figalli Singularity Set). The number of connected components of $\text{Sing}$ is at most countable. Moreover:

1. either $S_i$ coincides with an isolated point $\{ x_i \}$ for some $x_i \in \Omega$, and in this case the boundary of $\partial(x_i)$ is entirely contained inside $\partial \Lambda$ (so that $\partial \varphi(x_i)$ completely fills a hole in $\Lambda$);

2. or $S_i$ can be written as a disjoint union as follows:
\[ S_i = \bigcup_j \gamma_{ij}, \]
where $\gamma_{ij} : I_{ij} \to \text{Sing}$ are embedded Lipschitz curves parameterized by arc-length, $I_{ij}$ is an interval.

**B.3 Elementary Experiments**

We have designed several numerical experiments to verify Figalli’s theorems in low dimensional cases.

As shown in Fig.5 the source domain $\Omega$ is a rectangle, the target domain $\Lambda$ is a dumb-cell shape, the density functions $f$ and $g$ are constant $1$, namely, uniform distribution. The optimal transport maps is obtained using our method, the Brenier potential $\varphi$ is not differentiable, the singularity set is near the middle of the rectangle, $\Sigma_1(\varphi) = \gamma_1 \cup \gamma_2$ as shown in the figure. At $\gamma_1$ and $\gamma_2$, $\varphi$ is continuous but not differentiable. Each point $x \in \Omega$ is mapped onto $\nabla \varphi(x) \in \Lambda$ with the same color. This shows even the target domain is simply connected, the concavity will induce the discontinuity of the transport map.

Fig.6 shows another computational result, which demonstrates the singularity structure in Figalli’s theorem. The source domain $\Omega$ is the unit disk, the target domain $\Lambda$ is with complicated geometry. The singularity set of the optimal transport map satisfies the description of Figalli’s theorem [4].

\[ \Sigma_1 = \bigcup_{i=0}^3 \gamma_k, \Sigma_2 = \bigcup_{j=0}^1 x_j. \]
Figure 5: Discontinuous Optimal transport map. $\gamma_1$ and $\gamma_2$ are two singularity sets.

Figure 6: Singularity structure of an optimal transport map.

Figure 7: Optimal transport between a solid ball to the Stanford bunny. The singular sets are the foldings on the boundary surface.

$\partial \varphi(x_0)$ fills the hole on $\Lambda$. For any interior point $p \in \gamma_1$, $\partial \varphi(p)$ is a line segment connecting two points on the boundary of $\Lambda$.

Fig. 7 shows the singularity set of an optimal transport map between volumetric domains. $\Omega$ is the solid ball, $\Gamma$ is the interior of Stanford bunny. The probability distributions are the uniform distribution. The Brenier potential $\varphi$ is obtained by solving the Monge-Ampère equation. The optimal transport map is visualized by a morphing sequence: $f_t := (1-t)id + t\nabla \varphi$ for $0 \leq t \leq 1$, the intermediate shape is given by $f_t(\Lambda)$. It is obvious that the boundary surface of the bunny is folded inside the ball, which form the singularity set of the optimal transport map $\nabla \varphi$. 
B.4 Experiments on Real Data Sets

The low dimensional experiments demonstrates Figalli’s theorems, which are general for any dimension. In the following, we design and carry out an experiment for high dimensional real data, CelebA.

As shown in Fig. 8, we use an Autoencoder to encode the CelebA data samples to the latent space, which can be treated as a probability distribution with density function \( g : \Lambda \rightarrow \mathbb{R} \), whose support set is \( \Lambda \). We define the source domain \( \Omega \) as the unit cube, the source density function \( f \equiv 1 \), namely uniform distribution. Then we compute the optimal transport map from \( (\Omega, f) \) to \( (\Lambda, g) \), the Brenier potential is \( \varphi \). Then we random generate a sample \( x \) from \( (\Omega, f) \), then map it to the data distribution \( (\Lambda, g) \), the image is \( \nabla \varphi(x) \). Finally, \( \nabla \varphi(x) \) is mapped back to the image space by the decoder map to obtain a generated facial image. Fig. 8 left frame shows the examples of generated facial images.

We then search the evidence of the existence of the singularity set of the Brenier potential \( \varphi \). We randomly draw a line segment, \( \gamma \), which produce morphing sequence between two facial images as shown in the right frame of Fig. 8. The starting point \( \gamma(0) \) corresponds to the boy face with brown eyes on the left top corner, the ending point \( \gamma(1) \) represents the girl face with blue eyes on the right bottom corner. For each \( t \in [0,1] \), \( \gamma(t) \) is mapped by \( \nabla \varphi \) to the latent data distribution \( (\Lambda, g) \), then decoded to an facial image interpolating the boy and the girl facial images.

In the center of frame (b), for some specific \( t_0 \in (0,1) \), the generated image by \( \gamma(t_0) \) is with one blue eye and one brown eye. In reality, such kind of persons are extremely rare, therefore, we can treat such kind of facial images as on the boundary of the support \( \Lambda \) of the data distribution \( g \) in the latent space. This means \( t_0 \) is in the singularity set of \( \varphi \), the subgradient \( \partial \varphi(\gamma(t_0)) \) intersects \( \partial \Lambda \) at multiple points. The Brenier potential at \( \gamma(t_0) \) is only continuous but not differentiable, \( \nabla \varphi \) at \( \gamma(t_0) \) is discontinuous.

In summary, the generated unrealistic facial image shows the transport map is discontinuous, which verifies our hypothesis: the concavity of the support of the real data distribution causes the discontinuity of the transport map, which can not be directly represented by DNNs and induces instability of the training process or generating unrealistic samples.
C ADDITIONAL EXPERIMENTS

C.1 SINGLE PARAMETER SELECTIVE INTERPOLATION

On synthetic datasets, effects of angle threshold filtering can be visually inspected. As illustrated in Fig. 9, number of mode is a monotonically increasing function with respect to angle threshold $\hat{\theta}$. Quality of generated samples is effected directly by choosing different $\hat{\theta}$. Generally, small $\hat{\theta}$ encourages interpolation in between closely related real samples while too large $\hat{\theta}$ will result in interpolation between samples from different modes, which might in turn lower generation quality. On synthetic datasets, where modes are isotropic and different modes are clearly separable, an ideal $\hat{\theta}$ that captures all modes while avoids generating low quality samples can be chosen within a relatively wide band. For real world datasets of unknown modes, an ideal $\hat{\theta}$ needs to be hand tuned as the separability of different modes depends largely on input data pattern and quality of the embedding map.

Figure 9: Effect of increasing angle threshold $\hat{\theta}$. (a) shows target distribution. (b) and (c) shows AE-OT results when $\hat{\theta}$ is too small (as in (b)) or too large (as in (c)). (d) shows a proper choice of $\hat{\theta}$ that precisely captured and generalized all modes.

C.2 MORE RESULTS IN SYNTHETIC DATASETS

For illustration purpose, we plot our results on 2D-ring dataset along with those of GAN, PacGAN and BourGAN in Fig. 10. It is obvious that our method not only covers all of the modes, also the generation of meaningless data is overcome.

Figure 10: Experiments results on synthetic datasets. The generation of 8 Gaussians in 2D-circle by GAN, PacGAN, BourGAN and AE-OT. (a) Target Distribution. (b) Generated samples of GAN model. (c) Generated samples of PacGAN. (d) Generated samples of BourGAN (e) Generated samples of AE-OT. It is observed that samples generated by the GAN model fail to capture all modes. Samples from PacGAN capture all modes, but with inter-mode spurious samples. Though both BourGAN and AE-OT capture the target multimodal distribution precisely, the generation quality of the proposed method is better than BourGAN.

C.3 STACK MNIST EXPERIMENT

Experiments of varisou GAN models on stacked MNIST dataset are in consistent with Lin et al. (2018). For AE-OT model, we use the architecture shown in table 3 and table 4, with the decoder architecture same as the consistent generator architectures in GANs, and encoder having mirrored architecture.
Table 3: Encoder architecture for stack MNIST

<table>
<thead>
<tr>
<th>layer</th>
<th>number of outputs</th>
<th>kernel size</th>
<th>stride</th>
<th>BN</th>
<th>activation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Input $x \sim P_{data}$</td>
<td>28<em>28</em>3</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Convolution</td>
<td>14<em>14</em>16</td>
<td>5*5</td>
<td>2</td>
<td></td>
<td>LeakyReLU</td>
</tr>
<tr>
<td>Convolution</td>
<td>7<em>7</em>32</td>
<td>5*5</td>
<td>2</td>
<td></td>
<td>LeakyReLU</td>
</tr>
<tr>
<td>Convolution</td>
<td>4<em>4</em>64</td>
<td>5*5</td>
<td>2</td>
<td></td>
<td>LeakyReLU</td>
</tr>
<tr>
<td>Convolution</td>
<td>2<em>2</em>128</td>
<td>5*5</td>
<td>2</td>
<td></td>
<td>LeakyReLU</td>
</tr>
<tr>
<td>Fully connected</td>
<td>100</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 4: Decoder architecture for stack MNIST

<table>
<thead>
<tr>
<th>layer</th>
<th>number of outputs</th>
<th>kernel size</th>
<th>stride</th>
<th>BN</th>
<th>activation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Input $z \sim P_{latent}$</td>
<td>100</td>
<td></td>
<td></td>
<td></td>
<td>ReLU</td>
</tr>
<tr>
<td>Fully connected</td>
<td>2<em>2</em>128</td>
<td></td>
<td></td>
<td>Yes</td>
<td>ReLU</td>
</tr>
<tr>
<td>Transposed Convolution</td>
<td>4<em>4</em>64</td>
<td>5*5</td>
<td>2</td>
<td></td>
<td>ReLU</td>
</tr>
<tr>
<td>Transposed Convolution</td>
<td>7<em>7</em>32</td>
<td>5*5</td>
<td>2</td>
<td></td>
<td>ReLU</td>
</tr>
<tr>
<td>Transposed Convolution</td>
<td>14<em>14</em>16</td>
<td>5*5</td>
<td>2</td>
<td></td>
<td>ReLU</td>
</tr>
<tr>
<td>Transposed Convolution</td>
<td>28<em>28</em>3</td>
<td>5*5</td>
<td>2</td>
<td></td>
<td>Tanh</td>
</tr>
</tbody>
</table>

We test diversity of generated samples from our AE-OT method on stack MNIST dataset that consists of 128,000 samples in 1,000 modes with each sample stacking three handwritten digit images from MNIST dataset [LeCun et al. (1998)]. Number of modes counts the amount of modes captured by samples produced a generative model. The reverse KL divergence is computed by first assign each samples to their nearest mode, and compute the KL divergence between histogram of sample count on each mode and the histogram of real data. We choose angle threshold $\hat{\theta} = 0.5$ for AE-OT method. Details of network architectures are listed in supplementary materials. Experiments results are summarized in table 5, which show our method achieves best performance in terms of modes captured and reverse KL divergence on stacked MNIST dataset.

Table 5: Experiments on stacked MNIST. Results have shown that our method achieves best results in terms of mode captured and reverse KL divergence. (*) In WGAN, PacWGAN and AE-OT experiments, number of feature maps in each network layer is a quarter of those in other experiments.

<table>
<thead>
<tr>
<th>Stacked MNIST</th>
<th>Modes</th>
<th>KL</th>
</tr>
</thead>
<tbody>
<tr>
<td>DCGAN</td>
<td>99.0</td>
<td>3.40</td>
</tr>
<tr>
<td>ALI</td>
<td>16.0</td>
<td>5.40</td>
</tr>
<tr>
<td>Unrolled GAN</td>
<td>48.7</td>
<td>4.32</td>
</tr>
<tr>
<td>VEEGAN</td>
<td>150.0</td>
<td>2.95</td>
</tr>
<tr>
<td>MD</td>
<td>24.5 ± 7.67</td>
<td>5.49 ± 0.42</td>
</tr>
<tr>
<td>PacDCGAN4</td>
<td>1000.0 ± 0.00</td>
<td>0.07 ± 0.005</td>
</tr>
<tr>
<td>WGAN(*)</td>
<td>314.3 ± 38.54</td>
<td>2.44 ± 0.170</td>
</tr>
<tr>
<td>PacWGAN4(*)</td>
<td>965.7 ± 19.07</td>
<td>0.42 ± 0.094</td>
</tr>
<tr>
<td>AE-OT(*)</td>
<td>1000.0 ± 0.0</td>
<td>0.03 ± 0.0008</td>
</tr>
</tbody>
</table>

C.4 CELEBA EXPERIMENT

we evaluate our method on CelebA dataset by measuring collision probability in a batch of 1024 generated images of size 64-by-64. If a pair of identical images appear, a collision is declared, and thus higher collision probability means lower generation diversity. The same metric has been used in [Lin et al. (2018)] for evaluation of PacGAN. To make a consistent comparison, we design our autoencoder network with encoder having the same architecture as in previous work and decoder having a mirrored architecture of encoder. Angle threshold $\hat{\theta}$ is chosen to be 0.7 for AE-OT test. Results are listed in table 5 with corresponding images can be downloaded [here]. Results have
shown that our method achieves best result in terms of probability of collision. Autoencoder network structures can be found at table 7 and 8.

Table 6: Probability of identical images in a batch of 1024 generated images from DCGAN, PacGAN2 and AE-OT. Results have shown that our method achieves best result in terms of collision probability on CelebA dataset.

<table>
<thead>
<tr>
<th>Discriminator size (Decoder size)</th>
<th>DCGAN</th>
<th>PacDCGAN2</th>
<th>AE-OT</th>
</tr>
</thead>
<tbody>
<tr>
<td>273K</td>
<td>1</td>
<td>0.33</td>
<td>0</td>
</tr>
<tr>
<td>4×273K</td>
<td>0.42</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>16×273K</td>
<td>0.86</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>25×273K</td>
<td>0.65</td>
<td>0.17</td>
<td>0</td>
</tr>
</tbody>
</table>

Table 7: Encoder architecture in CelebA experiment

<table>
<thead>
<tr>
<th>layer</th>
<th>number of outputs</th>
<th>kernel size</th>
<th>stride</th>
<th>BN</th>
<th>activation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Input</td>
<td>x ~ Pdata</td>
<td>64×64×3</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Convolution</td>
<td>32×32×dim_f</td>
<td>4×4</td>
<td>2</td>
<td></td>
<td>LeakyReLU</td>
</tr>
<tr>
<td>Convolution</td>
<td>16×16×dim_f×2</td>
<td>4×4</td>
<td>2</td>
<td>Yes</td>
<td>LeakyReLU</td>
</tr>
<tr>
<td>Convolution</td>
<td>8×8×dim_f×4</td>
<td>4×4</td>
<td>2</td>
<td>Yes</td>
<td>LeakyReLU</td>
</tr>
<tr>
<td>Convolution</td>
<td>4×4×dim_f×8</td>
<td>4×4</td>
<td>2</td>
<td>Yes</td>
<td>LeakyReLU</td>
</tr>
<tr>
<td>Convolution</td>
<td>100</td>
<td>4×4</td>
<td>1</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 8: Decoder architecture in CelebA experiment

<table>
<thead>
<tr>
<th>layer</th>
<th>number of outputs</th>
<th>kernel size</th>
<th>stride</th>
<th>BN</th>
<th>activation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Input</td>
<td>z ~ Platent</td>
<td>100</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Transposed Convolution</td>
<td>4×4×dim_f×8</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Transposed Convolution</td>
<td>8×8×dim_f×4</td>
<td>4×4</td>
<td>2</td>
<td>Yes</td>
<td>ReLU</td>
</tr>
<tr>
<td>Transposed Convolution</td>
<td>16×16×dim_f×2</td>
<td>4×4</td>
<td>2</td>
<td>Yes</td>
<td>ReLU</td>
</tr>
<tr>
<td>Transposed Convolution</td>
<td>32×32×dim_f</td>
<td>4×4</td>
<td>2</td>
<td>Yes</td>
<td>ReLU</td>
</tr>
<tr>
<td>Transposed Convolution</td>
<td>64×64×3</td>
<td>4×4</td>
<td>2</td>
<td></td>
<td>Tanh</td>
</tr>
</tbody>
</table>

C.5 Parameter Setting and Training Details of AE-OT on 4 Public Datasets

During the training of the autoencoders for MNIST, FANSION, Cifar10 and CelebA, when the $L_2$ loss stops descending, which means that the network has found a good encoding function of the image space, we freeze the encoder part and continue to train the network from latent space to image space. And for the autoencoders, we run 200 epochs in total, including 150 epochs before the freezing of decoder, and 50 epochs after. The training loss before and after the freezing of decoder is shown in Tab. 9.

Besides, the parameters involved in the OT computation is set as follows: we set the parameters of Adam algorithm to be $\alpha = 1.0, \beta_1 = 0.9, \beta_2 = 0.5$ for all the experiments. The $\hat{\theta}$ is different for different tasks. Specifically, for the MNIST and the FANSION-MNIST dataset, $\hat{\theta} = \arccos 0.8$; for CIFAR10 and CelebA, $\hat{\theta} = \arccos 0.65$. When the sum of measure difference $\sum_{i=1}^N |\hat{\omega}_i(h) - \nu_i|$ is less than 0.05, the loops stop.

The time consumed by AE-OT is mainly composed of two parts: the training of autoencoder and the computation of semi-discrete OT. The training details of the former is illustrated in Section C.5 of the Appendix and we report the latter in table 10 with the Intel Core i7-7820X CPU and NVIDIA GTX1080Ti GPU. Here the dimension of the latent code is 64 for all of the four datasets.
Table 9: The $L_2$ loss of the autoencoders before and after the freezing of encoder

<table>
<thead>
<tr>
<th></th>
<th>MNIST</th>
<th>FANSION</th>
<th>CIFAR-10</th>
<th>CelebA</th>
</tr>
</thead>
<tbody>
<tr>
<td>Before</td>
<td>0.0013</td>
<td>0.0026</td>
<td>0.0022</td>
<td>0.0077</td>
</tr>
<tr>
<td>After</td>
<td>0.0005</td>
<td>0.0011</td>
<td>0.0018</td>
<td>0.0074</td>
</tr>
</tbody>
</table>

Table 10: Time used to compute the semi-discrete OT for the four datasets.

<table>
<thead>
<tr>
<th></th>
<th>MNIST</th>
<th>FANSION</th>
<th>CIFAR10</th>
<th>CelebA</th>
</tr>
</thead>
<tbody>
<tr>
<td>Num</td>
<td>60k</td>
<td>60k</td>
<td>50k</td>
<td>~190k</td>
</tr>
<tr>
<td>Time(min)</td>
<td>~120</td>
<td>~120</td>
<td>~95</td>
<td>~480</td>
</tr>
</tbody>
</table>

C.6 QUANTITATIVE COMPARISON WITH PRECISION AND RECALL

FID score is an effective method to test the difference between the generated distribution and real data, but it mainly focuses on precision, and cannot accurately capture how much portion of real data a generative model could cover. The method proposed in Sajjadi et al. (2018) can compute the precision and the recall at the same time only given the same number of generated and reference images. Firstly, the images are encoded to the feature space by the inception net. Salimans et al. (2016). Secondly, the features corresponding to generated and real data were put together and clustered. Thirdly, the histograms, marked as $P(\omega)$ and $Q(\omega)$ for both kinds of images appeared in each cluster are computed. Finally, the precision $\alpha(\lambda)$ and recall $\beta(\lambda)$ can be defined as follows:

$$\alpha(\lambda) = \sum_{\omega \in \Omega} \min(\lambda P(\omega), Q(\omega))$$

$$\beta(\lambda) = \sum_{\omega \in \Omega} \min(P(\omega), \frac{Q(\omega)}{\lambda})$$

With different $\lambda$, we can get different pairs of $(\alpha(\lambda), \beta(\lambda))$. After the above definition, we can used the concept of $(F_8, F_{1/8})$ defined on Sajjadi et al. (2018) to quantify the relative importance of precision and recall.

We add the results of Hoshen & Malik (2019) and ours into the original recall-precision point sets and display them in Fig. 11, with khaki dot and red dot.

Previously, the best $(F_8, F_{1/8})$ pairs were got by Hoshen & Malik (2019) in MNIST, FASHION-MNIST, and we get slightly better scores on the both datasets. For the CIFAR-10 dataset, the precision of our model slightly underperform gan, but achieve better recall scores. Our recall score is also better than GLANN Hoshen & Malik (2019). The proposed model does not achieve better score than the highest one of GANs on CelebA dataset, due to imprecision of pre-trained autoencoder. If we expand the capacity of the autoencoder network, our model can get better scores. In particular, when adopting the GCGAN architecture Radford et al. (2016) as the decoder and its mirror as encoder, our model attains the best $(F_8, F_{1/8})$ score at $(0.807, 0.906)$. We draw this point in Fig. 11 as purple dot and the generated images is shown in Fig. 12.

The core problem of AE-OT is that if the autoencoder fails to approximate the manifold in the image space well, then the proposed method may not give a realistic result, just as the results of CelebA. But if the autoencoder approximates the original manifold well, our theory can guarantee to generate good results (proved by the FID and PRD scores of the expanded autoencoder and the images shown in Fig. 12).

C.7 LINEAR INTERPOLATION IN THE LATENT SPACE

Given any two images in the dataset, we can find the images between them by linear interpolation in the noise space because the one to one correspondence between $\mu$ mass centers in the noise space and the images in the dataset is provided by the proposed algorithm. For other generation models, though the interpolation can be done successfully in the noise space, they cannot find the correspondence from the noise space and the image space. The results of the linear interpolation are shown in Fig. 13.
Figure 11: The comparison of Precision-Recall pair in $(F_8, F_{1/8})$ in the 4 datasets. The khaki dots are the results of Hoshen & Malik (2019). The red dots are the results of the proposed method. The purple dot in the forth subfigure corresponds to the results of the architecture with two more convolutional layers.
Figure 12: The generated human faces with the architecture originated from DCGAN [Radford et al. (2016)]
Figure 13: The linear interpolation between given two faces in the dataset.