EFFICIENT SEMI-DISCRETE OPTIMAL TRANSPORT USING THE MAXIMUM RELATIVE ERROR BETWEEN DISTRIBUTIONS

Anonymous authors
Paper under double-blind review

ABSTRACT

Semi-Discrete Optimal Transport (SDOT) transforms a continuous distribution to a discrete distribution. However, existing SDOT algorithms for high dimensional distributions have two limitations. 1) It is difficult to evaluate the quality of the transport maps produced by SDOT algorithms, because computing a high-dimensional Wasserstein distance for SDOT is intractable and 2) The transport map cannot guarantee that all target points have the corresponding source points that map to them. To address these limitations, we introduce the Maximum Relative Error (MRE) and the L_1 distance between the target distribution and the transported distribution computed by an SDOT map. If the MRE is smaller than 1, then every target point is guaranteed to have an area in the source distribution that maps to it. We propose a statistical method to compute the lower and upper bounds of the MRE and the L_1 distance. We present an efficient Epoch Gradient Descent algorithm for SDOT (SDOT-EGD) that computes the learning rate, number of iterations, and number of epochs in order to guarantee an arbitrarily small expected value of the MRE. Experiments on both low and high-dimensional data show that SDOT-EGD is much faster and converges much better than state-of-the-art SDOT algorithms. We also show our method's potential to improve GAN training by avoiding the oscillation caused by randomly changing the association between noise and the real images.

1 INTRODUCTION

Optimal Transport (OT) is a principled way of measuring the discrepancy of two distributions by transforming the source distribution to the target distribution. In recent years, OT has been widely used in machine learning (Frogner et al., 2015; Arjovsky et al., 2017; Gulrajani et al., 2017; Liu et al., 2018), computer vision (Salimans et al., 2018; Liu et al., 2019; An et al., 2020a), computer graphics (Solomon & Vaxman, 2019; Lavenant et al., 2018), etc.

In Semi-Discrete Optimal Transport (SDOT) (Peyré et al., 2019), the source distribution is continuous while the target distribution is discrete. Recently, a number of methods have applied SDOT to generative models, an important topic in machine learning. An et al. (2020a) proposed an Auto-Encoder OT model applying SDOT in the latent space to tackle mode collapse in generative models. Further, An et al. (2020b) used SDOT to stabilize GAN training and produced highly realistic images. Leclaire & Rabin (2019; 2020) used SDOT on image patches for texture synthesis and style transfer.

While being a promising approach, compared to its discrete OT counterpart, the optimization of SDOT is underexplored, especially in high dimensional spaces. One may think of sampling from the source distribution and turn the SDOT into a discrete OT problem. However, it has been shown that if the discrete OT has an ϵ-suboptimal estimate of the SDOT in a d-dimensional space (Genevay et al., 2019), then the sample complexity is $O(\frac{1}{\epsilon^d})$, which is prohibitively expensive when d is large.

From the optimization point of view, existing SDOT algorithms can be broadly classified into two categories: deterministic methods and stochastic methods. Deterministic methods (Kitagawa, 2014; Kitagawa et al., 2016; Lévy, 2015; Mérigot, 2011; Gu et al., 2013) have faster convergence rates, but require computing the measure of a polytope. This is intractable for high dimensional distributions, and thus limits their applicability. Stochastic methods use Stochastic Gradient Descent (SGD) to optimize the SDOT objective and can be applied to any high dimensional distributions. SGD sets the step size to $\Theta(1/t)$, where t is the iteration number, to ensure convergence (Peyré et al., 2019). Averaging SGD (ASGD) (Aude et al., 2016; Peyré et al., 2019) uses a step size of $\Theta(1/\sqrt{t})$ and averages the past iterates. ASGD is proven to be faster than SGD (Peyré et al., 2019). Based on ASGD, Leclaire & Rabin (2019; 2020) proposed a multi-layer approach which has been shown to be much faster than single layer plain ASGD for SDOT. An et al. (2020a) used Monte-Carlo sampling to estimate the gradient and then the Adam optimizer (Kingma & Ba, 2015) to optimize SDOT.

SDOT algorithms decompose the source space into Laguerre cells (Gu et al., 2013). Each cell is mapped to a target point by the transport map outputted by an SDOT algorithm. For an optimal SDOT algorithm, the total probability value of each cell is equal to the probability value of the corresponding target point. However, all existing stochastic methods (Peyré et al., 2019; Leclaire & Rabin, 2019; 2020; An et al., 2020a) for SDOT in high dimensional space have two limitations. 1) They do not have a mechanism to evaluate the quality of their computed transport maps, because computing a high-dimensional Wasserstein distance for SDOT is intractable and 2) These SDOT algorithms (Peyré et al., 2019; Leclaire & Rabin, 2019; 2020; An et al., 2020a) cannot guarantee that each target point has an area in the source domain that maps to it. When training a GAN using SDOT, as shown in Fig. 1, each cell in the noise space is mapped to a real image (e.g. a to A, b to B). If an image does not have a corresponding area in the noise space that maps to it, we cannot sample a noise point that can be trained to resemble this image. This potential mode collapse is a consequence of the second limitation. One additional limitation for MC-Adam (An et al., 2020a) is that the sample complexity of Monte-Carlo sampling to estimate the gradient of the SDOT dual objective is not established.

To address the above limitations, we make the following contributions:

1) An SDOT map transforms a continuous source distribution to a discrete target distribution. If there is at least one probability value equal to 0 in the transported target distribution, this means that there is at least one target point that does not have an area in the source domain that maps to it. It further means that the relative error of the probability values for this target point is 1. Thus, if we force the Maximum value of the Relative Error (MRE) to be less than 1, then all probability values in the transported distribution will be greater than 0 and each target point is guaranteed to have a corresponding area in the source domain that maps to it.

2) We use the MRE and the L_1 distance between the target distribution and the transported distribution obtained by an OT map to measure the quality of an SDOT map. The L_1 distance is the L_1 norm of the gradient of the dual SDOT objective. We propose statistical methods to compute the lower bounds and upper bounds of the MRE and the L_1 distance. We establish the number of Monte-Carlo samples to compute the bounds of the MRE and the L_1 distance.

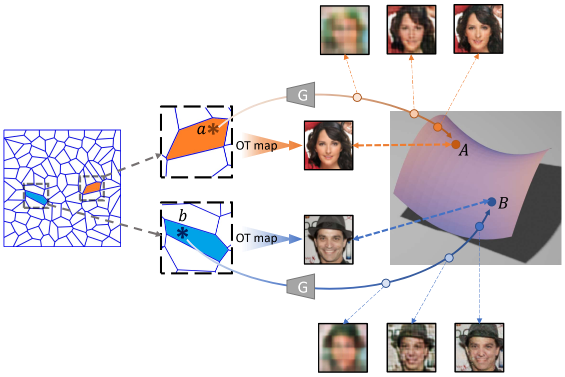


Figure 1: SDOT matched GAN training mechanism. a, b in the noise space are mapped to A and B in the image space, respectively.

3) We present an efficient Epoch Gradient Descent algorithm for SDOT (SDOT-EGD) that computes the learning rate, number of iterations and number of epochs in order to guarantee an arbitrarily small expected value of the MRE.

4) Experiments on 1D toy data show that our computed MRE lower and upper bounds are very close to each other and the real MRE and L_1 distance always lie in-between the lower and upper bounds. Experiments on 1D, 2D, 256D toy data and 256D real-world data show that SDOT-EGD converges significantly better and faster than state-of-the-art SDOT algorithms.

5) We propose a GAN training mechanism that uses SDOT-EGD to match the noise points and the real image during GAN training shown in Fig. 1, where each cell in the noise space is mapped to an real image. Experiments on the CelebA-HQ dataset (Karras et al.,

2018) and a COVID-19 X-ray image dataset (Konwer et al., 2021) show that our training mechanism improves GAN training performance by 20%-30% in most cases.

2 Semi-Discrete Optimal Transport

The Monge-Kantorovich dual Optimal Transport (OT) formulation (Villani, 2008) is given as:

Problem 1. *Given two bounded domains X and Y and their probability measures $\mu \in \mathbb{P}(X)$, and $\nu \in \mathbb{P}(Y)$, respectively, find a function v to solve*

$$\max_{v} \quad w(v) = \left\{ \int v^c(x) d\mu(x) + \int v(y) d\nu(y) \right\} \tag{1}$$

where $c : X \times Y \mapsto [0, +\infty]$ is the transport cost function, and $v^c(x)$ is the c-transform of v defined as: $v^c(x) = \min_{y \in Y}\{c(x, y) - v(y)\}$.

Consider μ and ν as the source and target distributions, respectively. Problem 1 is the continuous OT case. In this work, we focus on the semi-discrete OT case, i.e., the source distribution μ is continuous, but the target distribution ν is discrete. Let $\hat{Y} = \{y_i \in \mathbb{R}^d\}_{i=1}^n$ denote a finite set that contains n data points. Denote $v_{(i)} = v(y_i), i = 1, ..., n$. ν is a discrete target distribution with probability values in ν being $\nu_i, i = 1, ..., n$. Let $\nu_{\min}$ be the minimum probability value in ν, i.e., $\nu_{\min} = \min\{\nu_i\}_{i=1}^n$, Let $\mathcal{I}$ denote the set $\{1, 2, ..., n\}$. The Semi-Discrete OT (SDOT) can be formulated as maximizing

$$w(v) = \left\{ \int \min_{i \in \mathcal{I}} \left(c(x, y_i) - v_{(i)} \right) d\mu(x) + \sum_{i \in \mathcal{I}} v_{(i)} \nu_i \right\} \tag{2}$$

Let $h(x, v) = -\min_{i \in \mathcal{I}} \left(c(x, y_i) - v_{(i)} \right) - \sum_{i \in \mathcal{I}} v_{(i)} \nu_i$ and $f(v) = \mathbb{E}_{x \sim \mu} h(x, v)$. Then we have $w(v) = -f(v)$. Maximizing Eq. (2) is equivalent to minimizing $f(v)$. Since $f(v)$ has the form of the expectation of the function $h(x, v)$, it allows us to solve SDOT using Stochastic Gradient Descent (SGD) methods. The gradient of $f(v)$ w.r.t. each element of v is (Kitagawa et al., 2019):

$$\partial_{v_{(i)}} f(v) = \int_{\mathbb{L}_i(v)} d\mu(x) - \nu_i \tag{3}$$

where $\mathbb{L}_i(v)$ is defined as $\mathbb{L}_i(v) = \{x \in X : \forall j \neq i, c(x, y_i) - v_{(i)} \leq c(x, y_j) - v_{(j)}\}$. Each $\mathbb{L}_i(v)$ is called a Laguerre cell. The Laguerre cells form a partition of the space X, i.e. $X = \cup_i \mathbb{L}_i(v)$. In SDOT, all points in $\mathbb{L}_i(v)$ are transported to y_i. The gradient of $h(x, v)$ w.r.t. each element of v given x is:

$$\partial_{v_{(i)}} h(x, v) = \mathbb{1}_{\mathbb{L}_i(v)}(x) - \nu_i \tag{4}$$

where $\mathbb{1}_{\mathbb{L}_i(v)}$ is an indicator function of $\mathbb{L}_i(v)$.

Stochastic Gradient Descent (SGD) is commonly used to optimize SDOT. To ensure the convergence, the learning rate is typically $\Theta(1/t)$, where t is the iteration number. SGD becomes slow when t becomes large. Instead, almost all SDOT algorithms are based on Averaging Stochastic Gradient Descent (ASGD) (Peyré et al., 2019; Aude et al., 2016). ASGD uses a $\Theta(1/\sqrt{t})$ learning rate and guarantees that the SDOT objective converges in $O(1/\sqrt{T})$, where T is the total number of iterations.

3 Semi-Discrete Optimal Transport Using Epoch Gradient Descent

In this section, we introduce the Maximum Relative Error (MRE) and propose probabilistic bounds to evaluate the quality of an SDOT map outputted by an SDOT algorithm. Then, we will propose ASGD with a fixed step size and an efficient epoch gradient descent algorithm for solving SDOT. We start with an assumption.

Assumption 1. *We assume that all target points have probability values greater than 0, i.e., $\forall i \in \mathcal{I}, \nu_i > 0$.*

Assumption 1 is easily satisfied since we can remove the points that have zero probability value out of the target point set $\hat{Y}$.

3.1 The Maximum Relative Error (MRE)

A Semi-Discrete Optimal Transport (SDOT) algorithm divides the space of the source distribution into non-overlapping Laguerre cells. All the current stochastic optimization methods aim to achieve an ϵ-suboptimal SDOT objective. However, for SDOT, an ϵ-suboptimal objective value cannot truly reflect how well the source space is divided. For example, there might not exist regions in X that map to certain target points, even though the SDOT objective is ϵ-suboptimal, especially when the number of target points, n, is large.

In this subsection, we introduce the Maximum Relative Error (MRE) to evaluate how well the source space is divided using an SDOT map. Let p denote the transported distribution from the source distribution μ using the SDOT output v, i.e. $p_i = \int_{\mathbb{L}_i(v)} d\mu(x)$ denoting the total probability value of the Laguerre cell $\mathbb{L}_i(v)$. If there is a probability value in p that equals 0, e.g. $p_j = 0$, meaning that there is no area in the source domain that is mapped to y_j, then the relative error between p_j and ν_j is $|p_j - \nu_j|/\nu_j = 1$. Having the relative errors for all p_is and ν_is be less than 1 guarantees that all target points have preimages.

Therefore, we propose to use the Maximum Relative Error (MRE) between two probability distributions to measure the quality of an SDOT map produced by an SDOT algorithm:

$$\texttt{MRE}(\nu, p) = \max_{i \in \mathcal{I}} \frac{|p_i - \nu_i|}{\nu_i}. \tag{5}$$

$\texttt{MRE} < 1$ indicates all target points have preimages. This is important when applying SDOT to generative models such as Generative Adversarial Networks (GANs) (Goodfellow et al., 2014) where mode collapse will happen when not all target points have preimages.

3.2 SDOT Map Evaluation

In SDOT, the evaluation of how good an SDOT map is in high dimensional space remains an open problem. This problem boils down to computing the gradient of an SDOT objective, and further to computing the volume of each Laguerre cell in high dimensional space, which is intractable in practice (Leclaire & Rabin, 2020). Instead in this paper we propose a probabilistic approximation for SDOT quality evaluation. In the following theorem, we present the main results that give probabilistic bounds for evaluating an SDOT map, the L_1 distance between the empirical transported distribution and the target distribution, and the L_1 distance between the estimated gradient and the true gradient of an SDOT objective.

Theorem 1. *Suppose Assumption 1 is satisfied, given a continuous source distribution μ, a discrete target distribution ν, target data points $\hat{Y} = \{y_i \in \mathbb{R}^d\}_{i=1}^n$, SDOT output v, a precision ϵ_* and a confidence $1 - \delta$, $0 < \delta < 1$, let $\xi = \frac{\epsilon_*^2}{4(\sqrt{1+\epsilon_*}+1)^2}$, and let p be the discrete transported distribution using v. Draw $\lceil 1/(4\delta\xi\nu_{\min}) \rceil$ i.i.d. samples from μ and compute the empirical discrete transported distribution $\hat{p}$ using v. Let $g = p - \nu$ be the gradient and $\hat{g} = \hat{p} - \nu$ be the empirical gradient of the SDOT objective at v. Let $\hat{d} \overset{def}{=} \mathit{MRE}(\nu, \hat{p}) = \max_{i=1}^n \{|\hat{p}_i - \nu_i|/\nu_i\}$ be the empirical $\texttt{MRE}$, $d_1 = \|\hat{p} - \nu\|_1$ be the empirical L_1 distance between $\hat{p}$ and ν. Let $\Delta = 4\sqrt{\delta\xi n\nu_{\min}} + \sqrt{8\delta\log(1/\delta)\xi\nu_{\min}}$, and $\omega = \sqrt{\xi^2 + \hat{d}\xi + \xi}$. Then, with probability of at least $1 - \delta$, each of the following holds:* [1]

1) the lower bound of $\mathit{MRE}(\nu, p)$ is $\epsilon_{lb} = \max(\hat{d} - 2\omega + 2\xi, 0)$.
2) the upper bound of $\mathit{MRE}(\nu, p)$ is $\epsilon_{ub} = \min(\hat{d} + 2\omega + 2\xi, (1 - \nu_{\min})/\nu_{\min})$.
3) the L_1 distance between $\hat{p}$ and p is bounded as $\|\hat{p} - p\|_1 \leq \Delta$.
4) the L_1 distance between $\hat{g}$ and g is bounded as $\|\hat{g} - g\|_1 \leq \Delta$.
5) the L_1 distance between p and ν is bounded as $\max(d_1 - \Delta, 0) \leq \|p - \nu\|_1 \leq \min(d_1 + \Delta, 2)$.

The above theorem shows the lower and upper bounds of MRE and the L_1 distance using Monte-Carlo sampling with $\lceil 1/(4\delta\xi\nu_{\min}) \rceil$ samples. We use MRE and the L_1 distance between p and ν to estimate the quality of an SDOT map with lower and upper bounds given by Theorem 1 1), 2) and 5). We set a precision ϵ_* and compute ξ using ϵ_*. The MRE lower bound ϵ_{lb} is computed in 1) and the MRE upper bound ϵ_{ub} in 2). 3) gives us the bound of the L_1 distance between the empirical transported

[1] All theorem proofs can be found in the appendix.

distribution $\hat{p}$ and the real transported distribution p. 4) gives the bound of the L_1 distance between the estimated gradient and the true gradient of the SDOT objective evaluated at v. Interestingly, 3) and 4) do not depend on the data dimension d. This is very appealing because computing the exact volume of the Laguerre cells in high dimensional space is intractable. In practice, we can set a desired confidence $1-\delta$ and precision ϵ_*, and use $\hat{p}$ to approximate p. To estimate the L_1 distance between the target distribution ν and the transported distribution p, we need to compute the empirical L_1 distance d_1 and use 5) to compute the lower bound and upper bound of the true L_1 distance. The smaller the ϵ_*, the smaller the gap between the upper bound and the lower bound.

The algorithm for computing the `MRE` bounds is presented in Algorithm 1. In this algorithm, we simply use the average of the theoretical `MRE` lower bound and upper bound to estimate the real `MRE`. The time complexity of this algorithm is $O(dn/(\epsilon_*^2 \nu_{\min}))$.

3.3 Averaging Stochastic Gradient Descent with a Fixed Step Size

In this subsection, we analyze the convergence of Averaging SGD with a Fixed step size (ASGDF) for SDOT, since our SDOT-EGD is built upon ASGDF.

Suppose X is bounded. Denote the maximum transport cost between X and $\hat{Y}$ by $C_\infty = \sup_{x,y} c(x,y)$. In the t-th iteration of ASGDF, $t \geq 1$, by randomly drawing a sample x_t from μ, we obtain a function $f_t(v_{t-1}) = h(x_t, v_{t-1})$ according to $f(v)$. The SGD recursion of v is expressed as: $v_t = v_{t-1} - \gamma f_t'(v_{t-1})$, where f_t' is the gradient of f_t.

As `MRE` is an effective indicator of how well the source space is divided, we propose an algorithm that minimizes the `MRE`. From the definition of `MRE` in Eq. (5), we observe that `MRE` is closely related to the gradient norm of the objective function $f(v)$. Let $\bar{v} = \frac{1}{T}\sum_{t=1}^{T} v_{t-1}$. In the following theorem, we give the optimal step size γ and the bound of the expectation of the gradient norm of f at $\bar{v}$ for a fixed number of total iterations T.

Algorithm 1 Estimate the Maximum Relative Error (Estimate-`MRE`)

```
Input: data $Y = \{y_i\}_{i=1}^n$, source distribution $\mu$, target discrete distribution $\{\nu_i\}_{i=1}^n$, SDOT output $v$, confidence $1-\delta$, precision $\epsilon_*$.
Output: $\epsilon_{lb}$, $\epsilon_{ub}$, $\epsilon_{est}$
Let $\nu_{\min} = \min_{i=1}^n\{\nu_i\}$, $\xi = \frac{\epsilon_*^2}{4(\sqrt{1+\epsilon_*}+1)^2}$
Let $T = \lceil 1/(4\delta\xi\nu_{\min}) \rceil$, initialize $\bar{g} = 0$
for $j = 1$ to $T$ do
    Sample $x \sim \mu$
    Compute the gradient $g \leftarrow \nabla_v h(x, v)$
    Update average $\bar{g} = \frac{j-1}{j}\bar{g} + \frac{1}{j}g$
end for
Compute $\hat{p} \leftarrow \bar{g} + \nu$
Compute $\hat{d} = \max_{i=1}^n\{|\hat{p}_i - \nu_i|/\nu_i\}$
Let $\epsilon_{lb} = \hat{d} - 2(\sqrt{\xi^2 + \hat{d}\xi + \xi} - \xi)$
MRE lower bound $\epsilon_{lb} = \max(\epsilon_{lb}, 0)$
Let $\epsilon_{ub} = \hat{d} + 2(\sqrt{\xi^2 + \hat{d}\xi + \xi} + \xi)$
MRE upper bound $\epsilon_{ub} = \min(\epsilon_{ub}, (1-\nu_{\min})/\nu_{\min})$
Estimated MRE $\epsilon_{est} = (\epsilon_{lb} + \epsilon_{ub})/2$
```

Theorem 2. *With constant step size $\gamma = \frac{\sqrt{n}C_\infty + 1}{12\sqrt{T}}$, the expectation of the gradient norm of function f at $\bar{v}$ is bounded by $\mathbb{E}\|f'(\bar{v})\|^2 \leq \frac{4}{T}\left(\frac{4}{\sqrt{T}} + 10 + 6\sqrt{n}C_\infty\right)^2$.*

The step size in Theorem 2 is $O(1/\sqrt{T})$, with detailed parameters computed for SDOT, and the convergence rate of the expectation of the gradient norm of the SDOT objective is $O(1/T)$. These convergence results match the state-of-the-art convergence results in Bach (2014).

We want to design an ASGDF algorithm that outputs an SDOT map using $\bar{v}$ such that $\texttt{MRE}(\nu, p) \leq \epsilon$ for any given $\epsilon > 0$. Based on Theorem 2, we show how to set the step size and the number of iterations in ASGDF in order to achieve a desired expected value of the `MRE` in the following theorem.

Theorem 3. *Suppose Assumption 1 is satisfied, $\forall \epsilon > 0$, with constant step size $\gamma = \frac{\epsilon\nu_{\min}}{24} \cdot \frac{(1+\sqrt{n}C_\infty)}{(14+6\sqrt{n}C_\infty)}$ and number of iterations $T = \frac{4(14+6\sqrt{n}C_\infty)^2}{\epsilon^2\nu_{\min}^2}$, we have $\mathbb{E}[\mathit{MRE}(\nu, p)] \leq \epsilon$.*

In Theorem 3, the transport cost does not play an important role in choosing γ. For any transport cost function, and any maximum transport cost C_∞, $(1+\sqrt{n}C_\infty)/(14+6\sqrt{n}C_\infty)$ ranges in $[1/14, 1/6]$. If the target distribution ν is uniform, then γ is proportional to $1/n$.

The ASGDF for SDOT is shown in Algorithm 2. In line 4 and 5, we compute the optimal step size and the maximum number of iterations. Line 14-17 are standard ASGD operations. Line 6, 9-12 are implemented to check if we can stop early. We evaluate the MRE for the current $\bar{v}$. We use the ϵ_{est} to approximate the real MRE. $\epsilon_{est} < \epsilon_k$ means that the current $\bar{v}$ is good enough and we return the $\bar{v}$. The time complexity of this algorithm is $O\left(dn^2C_\infty^2/(\epsilon^2\nu_{\min}^2)\right)$.

Algorithm 2 Averaging Stochastic Gradient Descent with Fixed Step Size for SDOT (ASGDF)

```
1:  Input: data $Y = \{y_i\}_{i=1}^n$, source distribution $\mu$, target discrete distribution $\{\nu_i\}_{i=1}^n$, initial $v$, confidence $1-\delta$, precision $\epsilon_*$, current precision $\epsilon_k$, maximum transport cost $C_\infty$
2:  Output: $\bar{v}$, $v$, $\epsilon_{lb}$, $\epsilon_{ub}$, $\epsilon_{est}$
3:  Let $\nu_{\min} = \min_{i=1}^n\{\nu_i\}$, $\xi = \frac{\epsilon_*^2}{4(\sqrt{1+\epsilon_*}+1)^2}$
4:  Let $\gamma_k = \epsilon_k \cdot \nu_{\min}/24 \cdot (1+\sqrt{n}C_\infty)/(14+6\sqrt{n}C_\infty)$
5:  Let $T_k = \lceil 4(14+6\sqrt{n}C_\infty)/(\epsilon_k^2\nu_{\min}^2)\rceil$
6:  Let $T_e = \lceil n/\xi \rceil$
7:  for $j = 1$ to $T_k$ do
8:    if $j \bmod T_e$ equals 0 then
9:      $\epsilon_{lb}, \epsilon_{ub}, \epsilon_{est} \leftarrow$ Estimate-MRE($Y$, $\mu$, $\nu$, $\bar{v}$, $1-\delta$, $\epsilon_*$)
10:     if $\epsilon_{est} < \epsilon_k$ then
11:       return
12:     end if
13:   else
14:     Sample $x \sim \mu$
15:     Compute the gradient $g \leftarrow \nabla_v h(x, v)$ (Eq. (4))
16:     Update $v \leftarrow v - \gamma_k g$
17:     Update average $\bar{v} \leftarrow \frac{j-1}{j}\bar{v} + \frac{1}{j}v$
18:   end if
19: end for
```

3.4 Epoch Gradient Descent for SDOT (SDOT-EGD)

Although the ASGDF algorithm ensures the convergence for any given ϵ, the objective decreases slowly in early iterations. To make ASGDF converge quickly in early iterations, motivated by the epoch gradient descent approach (Hazan & Kale, 2014), we design an Epoch Gradient Descent (EGD) algorithm for solving SDOT (SDOT-EGD). [2]

We list the steps of the SDOT-EGD algorithm in Algorithm 3. The algorithm initializes $\epsilon_k = 2n, k = 1$. At the end of each epoch, we decrease ϵ_k by half. In each epoch, we run the ASGDF (Algorithm 2) to update the v obtained from the last epoch. The algorithm terminates and returns $\bar{v}$ when the current $\epsilon_k < \epsilon_*/2$ or the estimated MRE is less than or equal to ϵ_*. The maximum number of epochs is $\lceil \log_2(8n/\epsilon_*) \rceil$.

Algorithm 3 Epoch Gradient Descent for SDOT (SDOT-EGD)

```
Input: data $Y = \{y_i\}_{i=1}^n$, source distribution $\mu$, target discrete distribution $\{\nu_i\}_{i=1}^n$, confidence $1-\delta$, precision $\epsilon_*$, maximum transport cost $C_\infty$.
Output: $\bar{v}$.
Initialize $\epsilon_1 = 2n$, $k = 1$, $\epsilon_{est} \leftarrow \epsilon_1$, and $v = 0$.
while $\epsilon_k \geq \epsilon_*/2$ and $\epsilon_{est} > \epsilon_*$ do
  $\bar{v}, v, \epsilon_{lb}, \epsilon_{ub}, \epsilon_{est} \leftarrow$ ASGDF($Y, \mu, \nu, v, 1-\delta, \epsilon_*, \epsilon_k, C_\infty$)
  $k = k+1$, $\epsilon_k = \epsilon_{k-1}/2$
end while
```

In this manner, the algorithm uses a larger step size at the start of the optimization. Thus, the objective decreases fast in the beginning. As the ϵ_k becomes smaller, the step size also decreases to ensure convergence. The time complexity of Algorithm 3 is the same as that of Algorithm 2.

4 Applications to GANs

In traditional GAN training methods, a batch of randomly sampled noise points and a batch of randomly sampled real data are used to train a GAN in each iteration. However, this can cause oscillations in GAN training. For example, a_1 and a_2 are noise points that are very close to each other in a local neighbourhood. A and B are real images that are very far from each other. In a GAN training iteration, a_1 and image A are sampled to train the GAN. Hence, the generator is trained to generate an image that is similar to A using a_1. But in the next iteration, a_2 and image B could be sampled to train a GAN. The generator is trained to generate an image that is similar to B using a_2. Therefore, the generator could be trained to generate very different images using noise points from the same local area in different training iterations. This leads to unstable GAN training.

We propose the SDOT matched GAN training mechanism shown in Fig. 1, where G is a Generator, a, b are noise points, and A, B are real images. The leftmost box is the noise space and the rightmost image is the data manifold. The data manifold has a low-dimensional embedding in the feature space.

[2] Despite its name, SDOT-EGD is in fact a stochastic method.

The feature space and the noise space have the same dimensionality. We use a feature extractor (See Sec. 5.3) to extract image features. To train a GAN, the feature extractor is fixed and **an SDOT map is computed in advance from the noise space to the feature space**. As each region in the noise space is mapped to an image feature using the SDOT map and the image feature corresponds to a real image, we have a mapping from the noise points to the real images. For example, in Fig. 1, noise in the orange cell is mapped to image A and noise in the blue cell is mapped to image B, consistently throughout the whole GAN training process. A generator is trained to generate images similar to image A using noise points in the orange cell and images similar to image B using noise points in the blue cell. This stabilizes GAN training. The detailed training process is in the appendix.

5 Experiments

In the experiments, we compare the proposed SDOT-EGD against the ASGD (Peyré et al., 2019; Aude et al., 2016) because it is widely used in optimizing SDOT. We compare against the recently proposed MC-Adam (An et al., 2020a) and the 2-layer ASGD (Leclaire & Rabin, 2020). We also compare with the original Vanilla-EGD (Hazan & Kale, 2014) because SDOT-EGD is an EGD-type method. We experiment on 1D, 2D and high dimensional toy data, and high dimensional real data.

5.1 1D Experiments

The SDOT has a closed form solution for 1D data (Leclaire & Rabin, 2020). We use the exact solution to compute `MRE` and the L_1 distance between p nad ν. In this experiment, the source distribution μ is a uniform distribution in $[0, 1]$. We sample 1000 target points in $[-1, 1]$ with equal intervals between every two adjacent points. The target probability for each point is $1/1000$.

In Fig. 2 a) and d), the shaded area depicts the computed lower and upper bounds given by our theoretical analysis. The solid curves in a) and d) are exact values. The exact values always lie inside the computed bounds. This verifies the correctness of our theoretical analysis of the bounds. Note that the shaded area in Fig. 2 (a) is too thin to be easily visible because the average upper bound and lower bound gap is smaller than 0.2. In both figures, SDOT-EGD converges rapidly to 0, while other curves converge at high values. This demonstrates the effectiveness of SDOT-EGD.

5.2 2D Experiments

In 2D experiments, the source distribution μ is uniformly distributed in $[0, 1] \times [0, 1]$. We randomly sample 10000 points in $[0, 1] \times [0, 1]$ as target points, with each point having probability value of 1e-4. See Fig. 3 f) for the distribution of these target points. Therefore, an optimal SDOT algorithm should divide $[0, 1] \times [0, 1]$ into 10000 Laguerre cells, with the area of each cell equal to 1e-4.

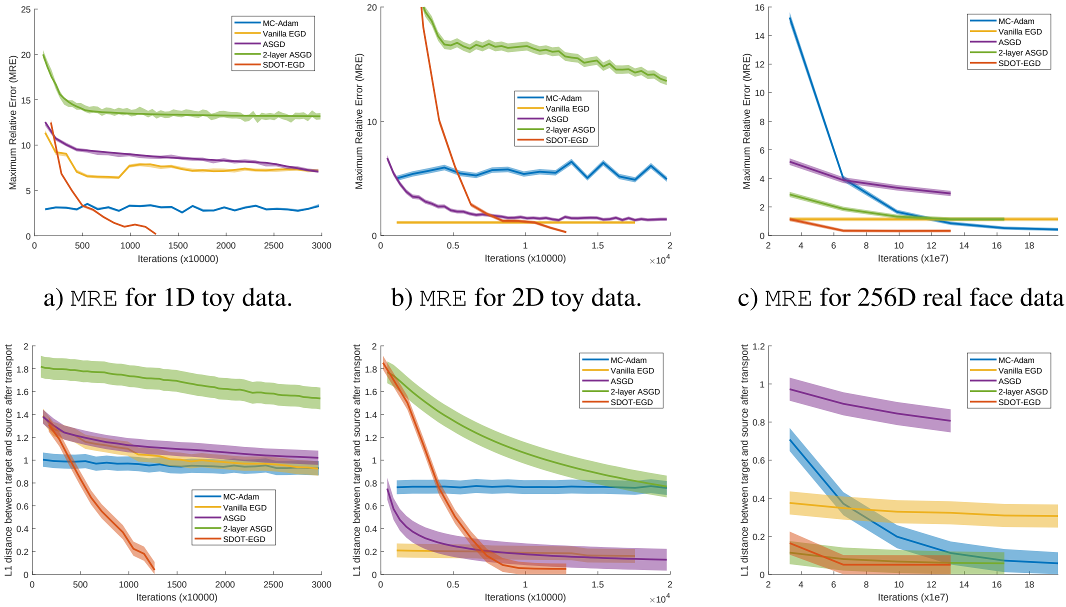
Fig. 3 visualizes the partition of the space by various methods. Each cell is colored according to its area from small (blue) to large (red). We observe that SDOT-EGD divides the space into more uniformly distributed cells.

Since we do not compute the exact `MRE` and L_1 distance between p and ν for 2D data, we compute the estimated `MRE` and L_1 distance which are the average values of their lower and upper bounds, respectively. We plot the estimated `MRE` and L_1 curves in Fig. 2 b) and e) as well as their bounds. Fig. 2 b) shows that SDOT-EGD can achieve much smaller `MRE` than other methods close to the 1.3×10^8 iteration point. Fig. 2 c) shows that SDOT-EGD converges faster than other methods.

5.3 High Dimension Experiments

To see whether these methods scale to high dimensional data, we evaluate the performance of different methods in the 256D space. We evaluate various methods on both real-world data and toy data.

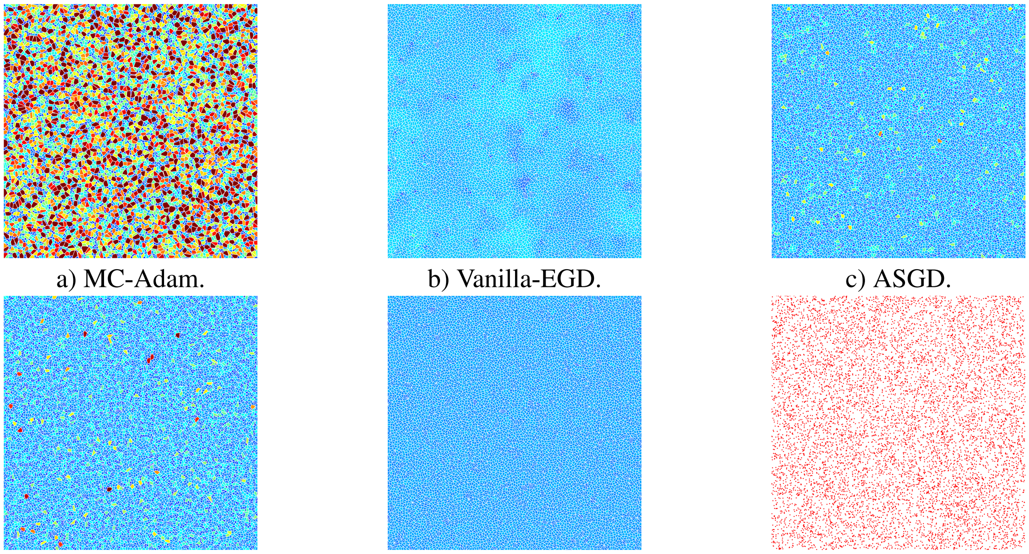
The CelebA-HQ dataset contains 29970 unique images. We downsize each image to 16×16 size and transform the RGB images to gray images. The pixel values are re-scaled in $[-1, 1]$. The 16×16 gray images become 256D vectors. These 256D vectors are regarded as features and used as the target distribution. The source distribution is a uniform distribution in $[-1, 1]^{256}$. Fig. 2 (c) and (f) show the performance of the various methods. In Fig. 2 (c), SDOT-EGD always has the lowest `MRE`. After 70M iterations, SDOT-EGD's `MRE` is close to 0.2. This guarantees that each image has an area



a) MRE for 1D toy data. b) MRE for 2D toy data. c) MRE for 256D real face data.

d) L_1 distance for 1D toy data. e) L_1 distance for 2D toy data. f) L_1 distance for 256D real face data.

Figure 2: **Comparison of MRE and L_1 distance in 1D and 2D toy data, and 256D real face data.** In each figure, the shaded area depicts the computed lower bound and upper bound given by our theoretical analysis. The solid curves in a) and d) are real MRE values. The computed bounds coincide very well with the real MRE values. This verifies the correctness of our theoretical analysis of the bounds for SDOT Map evaluation. Solid curves in b) c) e) and f) are estimated curves using the mean of the lower bound and upper bound. The MRE shaded bands are much thinner than the L_1 distance ones. This means that MRE evaluates SDOT maps more accurately. In all experiments, the proposed SDOT-EGD converges much better and faster than other methods.



a) MC-Adam. b) Vanilla-EGD. c) ASGD.

d) 2-layer ASGD. e) SDOT-EGD. f) Target distribution.

Figure 3: Laguerre cell partition. Cells are colored per their area from small (blue) to large (red). Cell partition from our method SDOT-EGD is more uniform. MC-Adam has high cell area variance. ASGD and 2-layer ASGD have some cells that are large. The vanilla-EGD has large cells in the center, and has small cells in the lower-right and upper-left corners.

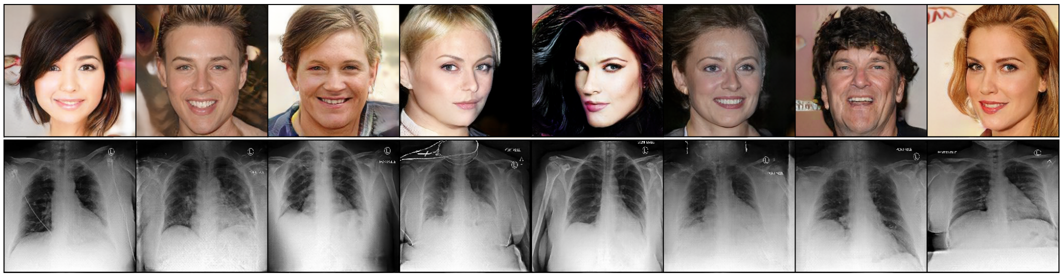


Figure 4: Images generated by GAN-0GP (SDOT-EGD). Fist row shows images generated on the CelebA-HQ dataset, and the second row show images generated on the COVOC dataset.

in the source domain that maps to it. All the other methods in Fig. 2 (c) have MRE values greater than 1. Fig. 2 (f) shows the results using L_1 distance. SDOT-EGD surpasses the 2-layer ASGD after 70M iterations and is the best method among all the methods. This shows that SDOT-EGD not only works well with MRE but also achieves lower L_1 distance compared to state-of-the-art methods. The experiments of 256D toy data can be found in the appendix.

5.4 GAN Experiments

We evaluate the SDOT matched training mechanism in training GANs. We apply the SDOT-EGD on matching noise points and real images in GAN training. We denote the SDOT-EGD matched training mechanism using WGAN-GP (Gulrajani et al., 2017) by WGAN-GP (SDOT-EGD) and using GAN-0GP by GAN-0GP (SDOT-EGD). We perform experiments on two datasets: The CelebA-HQ dataset (Karras et al., 2018) and a COVID-19 Outcome dataset (COVOC) (Konwer et al., 2021; Zhou et al., 2021) containing 248 X-ray images. More details of this dataset could be found in the appendix. The image size used in our experiments is 256×256.

Table 1 lists the Fréchet Inception Distances (FIDs) (Heusel et al., 2017; Segal et al., 2021) of different methods. On the CelebA-HQ dataset, GAN-0GP (SDOT-EGD) supasses AE-OT-GAN and achieves the state-of-the-art FID of 6.99 in 256×256 image size. On the CelebA-HQ datset, WGAN-GP (SDOT-EGD) reduces the FID of WGAN-GP from 12.94 to 9.18 and GAN-0GP (SDOT-EGD) reduces the FID of GAN-0GP from 9.82 to 6.99. On the COVOC dataset, GAN-0GP (SDOT-EGD) reduces the FID of GAN-0GP from 98.76 to 76.30. The performance improvements are bewteen 20% - 30%. This shows the effectiveness of the SDOT matching mechanism in GAN training. Fig. 4 shows the face images and X-ray images generated by GAN-0GP (SDOT-EGD). These images look realistic. Please see the appendix to see more generated images.

Method	CelebA-HQ	COVOC
AE-OT-GAN	7.20	-
WGAN-GP	12.94	96.49
WGAN-GP (SDOT-EGD)	9.18	94.91
GAN-0GP	9.82	98.76
GAN-0GP (SDOT-EGD)	**6.99**	**76.30**

Table 1: FID of the various methods. Image size on both datasets are 256×256.

6 Conclusion

In this paper, we proposed to use the MRE and the L_1 distance to measure the performance of an SDOT map. We proposed to compute the lower and upper bounds of the MRE and the L_1 distance. The number of samples of Monte-Carlo sampling to compute the bounds for SDOT is established. Furthermore, we proposed the SDOT-EGD algorithm that can achieve an arbitrarily small expected value of the MRE. Experiments show that SDOT-EGD performs much better than state-of-the-art SDOT algorithms. We proposed to apply SDOT-EGD in GAN training. Experiments show that a GAN that uses SDOT-EGD to match the noise points and real data can significantly improve the performance of its counterpart that does not use SDOT-EGD.

Ethics Statement

The GAN application of this paper generates realistic synthetic images, that can be used for fraudulent or misinformation purposes which would be a potential negative social impact shared with most GAN methods. Any methods that try to mitigate potential biases in CelebA-HQ would also address bias concerns for our paper. The COVOC dataset used in our paper is a de-identified dataset (via DICOM deidentification process) and does not contain personally identifiable information. The COVOC dataset is obtained after Institutional Review Board (IRB) approval as non-human subjects research.

Reproducibility Statement

The assumption of using SDOT-EGD is given in Section 3. The proofs of all the theorems in the paper are provided in Section A.2. In Section A.3.2 and A.3.3, we run SDOT-EGD multiple times and plot the error bars in Fig. 7. The standard deviations are very small when SDOT-EGD terminates, indicating SDOT-EGD is reproducible.

References

Jayadev Acharya, Ilias Diakonikolas, Chinmay Hegde, Jerry Zheng Li, and Ludwig Schmidt. Fast and near-optimal algorithms for approximating distributions by histograms. In *Proceedings of the 34th ACM SIGMOD-SIGACT-SIGAI Symposium on Principles of Database Systems*, pp. 249–263, 2015.

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. *ICLR 2020*, 2020a.

Dongsheng An, Yang Guo, Min Zhang, Xin Qi, Na Lei, and Xianfang Gu. Ae-ot-gan: Training gans from data specific latent distribution. In *European Conference on Computer Vision*, pp. 548–564. Springer, 2020b.

Martin Arjovsky, Soumith Chintala, and Léon Bottou. Wasserstein generative adversarial networks. In *Proceedings of the International Conference on Machine Learning*, 2017.

Genevay Aude, Marco Cuturi, Gabriel Peyré, and Francis Bach. Stochastic optimization for large-scale optimal transport. *arXiv preprint arXiv:1605.08527*, 2016.

Francis Bach. Adaptivity of averaged stochastic gradient descent to local strong convexity for logistic regression. *The Journal of Machine Learning Research*, 15(1):595–627, 2014.

Andrew Brock, Jeff Donahue, and Karen Simonyan. Large scale GAN training for high fidelity natural image synthesis. In *Proceedings of the International Conference on Learning Representations*, 2019.

Siu On Chan, Ilias Diakonikolas, Rocco A Servedio, and Xiaorui Sun. Near-optimal density estimation in near-linear time using variable-width histograms. In Z. Ghahramani, M. Welling, C. Cortes, N. Lawrence, and K. Q. Weinberger (eds.), *Advances in Neural Information Processing Systems*, volume 27, pp. 1844–1852. Curran Associates, Inc., 2014.

Ilias Diakonikolas. Beyond histograms: structure and distribution estimation. In *Found at http://www. iliasdiakonikolas. org/stoc14-workshop/diakonikolas. pdf*, volume 1, pp. 1. Citeseer, 2014.

Charlie Frogner, Chiyuan Zhang, Hossein Mobahi, Mauricio Araya, and Tomaso A Poggio. Learning with a wasserstein loss. In *Advances in Neural Information Processing Systems*, 2015.

Aude Genevay, Lénaic Chizat, Francis Bach, Marco Cuturi, and Gabriel Peyré. Sample complexity of sinkhorn divergences. In *The 22nd International Conference on Artificial Intelligence and Statistics*, pp. 1574–1583. PMLR, 2019.

Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in Neural Information Processing Systems*, 2014.

Xianfeng Gu, Feng Luo, Jian Sun, and S-T Yau. Variational principles for minkowski type problems, discrete optimal transport, and discrete monge-ampere equations. *arXiv preprint arXiv:1302.5472*, 2013.

Ishaan Gulrajani, Faruk Ahmed, Martin Arjovsky, Vincent Dumoulin, and Aaron C Courville. Improved training of Wasserstein GANs. In *Advances in Neural Information Processing Systems*, 2017.

Elad Hazan and Satyen Kale. Beyond the regret minimization barrier: optimal algorithms for stochastic strongly-convex optimization. *The Journal of Machine Learning Research*, 15(1): 2489–2512, 2014.

Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. GANs trained by a two time-scale update rule converge to a local Nash equilibrium. In *Advances in Neural Information Processing Systems*, 2017.

Tero Karras, Timo Aila, Samuli Laine, and Jaakko Lehtinen. Progressive growing of gans for improved quality, stability, and variation. In *Proceedings of the International Conference on Learning Representations*, 2018.

Diederik P Kingma and Jimmy Ba. Adam: a method for stochastic optimization. In *Proceedings of the International Conference on Learning Representations*, 2015.

Jun Kitagawa. An iterative scheme for solving the optimal transportation problem. *Calculus of Variations and Partial Differential Equations*, 51(1):243–263, 2014.

Jun Kitagawa, Quentin Mérigot, and Boris Thibert. Convergence of a newton algorithm for semi-discrete optimal transport. *arXiv preprint arXiv:1603.05579*, 2016.

Jun Kitagawa, Quentin Mérigot, and Boris Thibert. Convergence of a newton algorithm for semi-discrete optimal transport. *Journal of the European Mathematical Society*, 21(9):2603–2651, 2019.

Aishik Konwer, Joseph Bae, Gagandeep Singh, Rishabh Gattu, Syed Ali, Jeremy Green, Tej Phatak, Amit Gupta, Chao Chen, Joel Saltz, and Prateek Prasanna. Predicting COVID-19 lung infiltrate progression on chest radiographs using spatio-temporal LSTM based encoder-decoder network. In *Medical Imaging with Deep Learning*, 2021. URL `https://openreview.net/forum?id=96BhL_MERil`.

Hugo Lavenant, Sebastian Claici, Edward Chien, and Justin Solomon. Dynamical optimal transport on discrete surfaces. *ACM Transactions on Graphics (TOG)*, 37(6):1–16, 2018.

Arthur Leclaire and Julien Rabin. A fast multi-layer approximation to semi-discrete optimal transport. In *International Conference on Scale Space and Variational Methods in Computer Vision*, pp. 341–353. Springer, 2019.

Arthur Leclaire and Julien Rabin. A stochastic multi-layer algorithm for semi-discrete optimal transport with applications to texture synthesis and style transfer. *Journal of Mathematical Imaging and Vision*, pp. 1–27, 2020.

Bruno Lévy. A numerical algorithm for l2 semi-discrete optimal transport in 3d. *ESAIM: Mathematical Modelling and Numerical Analysis*, 49(6):1693–1715, 2015.

Huidong Liu, Xianfeng Gu, and Dimitris Samaras. A two-step computation of the exact GAN Wasserstein distance. In *Proceedings of the International Conference on Machine Learning*, 2018.

Huidong Liu, Xianfeng Gu, and Dimitris Samaras. Wasserstein gan with quadratic transport cost. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 4832–4841, 2019.

Colin McDiarmid. On the method of bounded differences. *Surveys in combinatorics*, 141(1):148–188, 1989.

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

Lars Mescheder, Andreas Geiger, and Sebastian Nowozin. Which training methods for gans do actually converge? In *Proceedings of the International Conference on Machine Learning*, 2018.

Gabriel Peyré, Marco Cuturi, et al. Computational optimal transport. *Foundations and Trends in Machine Learning*, 11(5-6):355–607, 2019.

Iosif Pinelis. Optimum bounds for the distributions of martingales in banach spaces. *The Annals of Probability*, pp. 1679–1706, 1994.

Jawad Rasheed, Alaa Ali Hameed, Chawki Djeddi, Akhtar Jamil, and Fadi Al-Turjman. A machine learning-based framework for diagnosis of covid-19 from chest x-ray images. *Interdisciplinary Sciences: Computational Life Sciences*, 13(1):103–117, 2021.

Hojjat Salehinejad, Shahrokh Valaee, Tim Dowdell, Errol Colak, and Joseph Barfett. Generalization of deep neural networks for chest pathology classification in x-rays using generative adversarial networks. In *2018 IEEE international conference on acoustics, speech and signal processing (ICASSP)*, pp. 990–994. IEEE, 2018.

Tim Salimans, Han Zhang, Alec Radford, and Dimitris Metaxas. Improving gans using optimal transport. In *Proceedings of the International Conference on Learning Representations*, 2018.

Bradley Segal, David M Rubin, Grace Rubin, and Adam Pantanowitz. Evaluating the clinical realism of synthetic chest x-rays generated using progressively growing gans. *SN Computer Science*, 2(4): 1–17, 2021.

Justin Solomon and Amir Vaxman. Optimal transport-based polar interpolation of directional fields. *ACM Transactions on Graphics (TOG)*, 38(4):1–13, 2019.

Devansh Srivastav, Akansha Bajpai, and Prakash Srivastava. Improved classification for pneumonia detection using transfer learning with gan based synthetic image augmentation. In *2021 11th International Conference on Cloud Computing, Data Science & Engineering (Confluence)*, pp. 433–437. IEEE, 2021.

Tijmen Tieleman and Geoffrey Hinton. Lecture 6.5-rmsprop: Divide the gradient by a running average of its recent magnitude. *COURSERA: Neural networks for machine learning*, 4(2):26–31, 2012.

Cédric Villani. *Optimal transport: old and new*, volume 338. Springer Science & Business Media, 2008.

Lei Zhou, Joseph Bae, Huidong Liu, Gagandeep Singh, Jeremy Green, Dimitris Samaras, and Prateek Prasanna. Chest radiograph disentanglement for covid-19 outcome prediction. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pp. 345–355. Springer, 2021.

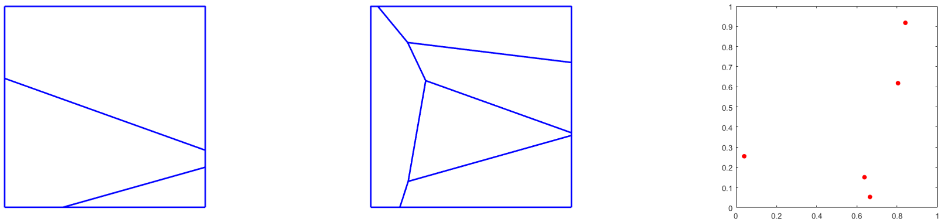
A Appendix

A.1 The Maximum Relative Error

The Maximum Relative Error (MRE) between the transported discrete distribution p, and the target distribution ν is defined as:

$$\mathtt{MRE}(\nu, p) = \max_{i \in \mathcal{I}} \frac{|p_i - \nu_i|}{\nu_i}. \tag{6}$$

Fig. 5 shows the relationship between different MRE values and the quality of their respective SDOT maps. The source distribution is a continuous uniform distribution in $[0,1] \times [0,1]$, shown in a) and b), and the discrete target distribution has 5 points , shown in c), with each point having the probability of 0.2. Fig. 5 a) and b) show the source space is divided by two SDOT maps. An optimal SDOT algorithm should divide the source space into 5 regions with equal area. In a), the source space



a) MRE = 1.8 b) MRE = 0.05 c) Target distribution.

Figure 5: An illustration of the relationship between the MREs and the quality of SDOT maps. The source distribution is a continuous uniform distribution in $[0, 1] \times [0, 1]$, shown in a) and b), and the discrete target distribution has 5 points , shown in c), with each point having the probability of 0.2. Fig. 5 a) and b) show the source space is divided by two SDOT maps. An optimal SDOT algorithm should divide the source space into 5 regions with equal area. In a), the source space is only divided in 3 regions, and thus 2 target points do not have regions in the source space that are mapped to them. The MRE in a) is $1.8 > 1$. In b), the source space is divided into 5 regions, with each region having approximately the same area. The MRE is $0.05 < 1$. b) is much better than a). This shows an example of using MRE to measure the qualities of SDOT maps.

is only divided in 3 regions, and thus 2 target points do not have regions in the source space that are mapped to them. The MRE in a) is $1.8 > 1$. In b), the source space is divided into 5 regions, with each region having approximately the same area. The MRE is $0.05 < 1$. b) is much better than a). This shows an example of using MRE to measure the quality of SDOT maps.

A.2 Proofs of Theorems

Next we introduce Theorem 4 in Chan et al. (2014); Diakonikolas (2014) and propose Lemma 1 that is important to prove Theorem 1.

Theorem 4. *(Chan et al., 2014; Diakonikolas, 2014) Let p be a distribution over $[n]$. Let $\hat{p}$ be an empirical distribution over n obtained by drawing K samples from p, then*
1)

$$\mathbb{E}[\|\hat{p} - p\|_1] \leq 2\sqrt{n/K} \tag{7}$$

and
2) for each $i \in [n]$, we have

$$\mathbb{E}[|\hat{p}_i - p_i|] \leq \sqrt{p_i(1 - p_i)/K} \tag{8}$$

Lemma 1. *Let p be a distribution over $[n]$. Let $\hat{p}$ be an empirical distribution over n obtained by drawing K samples from p. Then with probability of at least $1 - \delta$, $0 < \delta < 1$, each of the following holds*
1)

$$\|\hat{p} - p\|_1 \leq 2\sqrt{n/K} + \sqrt{2\log(1/\delta)/K} \tag{9}$$

2) for each $i \in [n]$,

$$|\hat{p}_i - p_i| \leq \kappa\sqrt{p_i(1 - p_i)/K} \tag{10}$$

where $\kappa = 1/\sqrt{\delta}$.

Proof. 1) The proof is motivated by the proof in Lemma 3.1 Acharya et al. (2015). Let $Y = \|\hat{p} - p\|_1$. We can write $Y = g(s_1, ..., s_K)$, where g is a function, $s_j \sim p$ and samples $s_j, j \in [K]$, are mutually independent. As the L_1 distance between any two probability measures are upper bounded by 2, the function g satisfies the following Lipschitz property:

$$|g(s_1, ..., s_j, ..., s_K) - g(s_1, .., s_j', ..., s_K)| \leq \frac{2}{K}$$

for any $j \in [K]$. Using McDiarmid's inequality (McDiarmid, 1989), we have $\mathbb{P}(Y > \mathbb{E}[Y] + \eta) \leq \exp(-\eta^2 K/2), \forall \eta > 0$. Therefore, $\mathbb{P}(Y > 2\sqrt{n/K} + \eta) \leq \mathbb{P}(Y > \mathbb{E}[Y] + \eta) \leq \exp(-\eta^2 K/2)$. Let $\delta = \exp(-\eta^2 K/2)$, we have Eq. (9).

2) Each i in Eq. (10) can be proved independently. For a specific i, let us consider Poisson trials $X_j, j = 1, ..., K$, where each X_j is a Bernoulli random variable with success probability being p_i, i.e., $\mathbb{P}(X_j = 1) = p_i$ and $\mathbb{P}(X_j = 0) = 1 - p_i, \forall j = 1, ..., K$. Therefore, the expectation of each X_j is $\mathbb{E}(X_j) = p_i$, and the variance is $Var(X_j) = p_i(1 - p_i)$. $\mathbb{P}(X_j = 1) = p_i$ means that, in the j-th Poisson trial, the probability of drawing the j-th sample falls in the i-th histogram bin is p_i. Let $X = \frac{1}{K}\sum_{j=1}^{K} X_j$, and thus, X follows a Poisson binomial distribution. For the Poission binomial distribution, $\mathbb{E}(X) = p_i$ and $Var(X) = p_i(1 - p_i)/K$. So, the standard deviation of X is $\sigma = \sqrt{p_i(1 - p_i)/K}$. Eq. (10) is to bound $\mathbb{P}(|X - p_i| \leq \kappa\sigma)$. Using the Chebyshev bound we have $\mathbb{P}(|X - p_i| \leq \kappa\sigma) \geq 1 - \frac{1}{\kappa^2}$. Let $\delta = \frac{1}{\kappa^2}$ we obtain Eq. (10). □

A.2.1 Proof of Theorem 1

We restate Theorem 1 below:

Theorem 1. *Suppose Assumption 1 is satisfied, given a continuous source distribution μ, a discrete target distribution ν, target data points $\hat{Y} = \{y_i \in \mathbb{R}^d\}_{i=1}^n$, SDOT output v, a precision ϵ_* and a confidence $1 - \delta$, $0 < \delta < 1$, let $\xi = \frac{\epsilon_*^2}{4(\sqrt{1+\epsilon_*}+1)^2}$, and let p be the discrete transported distribution using v. Draw $\lceil 1/(4\delta\xi\nu_{\min})\rceil$ i.i.d. samples from μ and compute the empirical discrete transported distribution $\hat{p}$ using v. Let $g = p - \nu$ be the gradient and $\hat{g} = \hat{p} - \nu$ be the empirical gradient of the SDOT objective at v. Let $\hat{d} \overset{def}{=} \texttt{MRE}(\nu, \hat{p}) = \max_{i=1}^n\{|\hat{p}_i - \nu_i|/\nu_i\}$ be the empirical MRE, $d_1 = \|\hat{p} - \nu\|_1$ be the empirical L_1 distance between $\hat{p}$ and ν. Let $\Delta = 4\sqrt{\delta\xi n\nu_{\min}} + \sqrt{8\delta\log(1/\delta)\xi\nu_{\min}}$, and $\omega = \sqrt{\xi^2 + \hat{d}\xi + \xi}$. Then, with probability of at least $1 - \delta$, each of the following holds:*
1) the lower bound of $\texttt{MRE}(\nu, p)$ is $\epsilon_{lb} = \max(\hat{d} - 2\omega + 2\xi, 0)$.
2) the upper bound of $\texttt{MRE}(\nu, p)$ is $\epsilon_{ub} = \min(\hat{d} + 2\omega + 2\xi, (1 - \nu_{\min})/\nu_{\min})$.
3) the L_1 distance between $\hat{p}$ and p is bounded as $\|\hat{p} - p\|_1 \leq \Delta$.
4) the L_1 distance between $\hat{g}$ and g is bounded as $\|\hat{g} - g\|_1 \leq \Delta$.
5) the L_1 distance between p and ν is bounded as $\max(d_1 - \Delta, 0) \leq \|p - \nu\|_1 \leq \min(d_1 + \Delta, 2)$.

Proof. We prove 1) and 2) together. The trivial lower bound in 1) is 0 and the trivial upper bound in 2) is $(1 - \nu_{\min})/\nu_{\min}$. Let $\kappa = 1/\sqrt{\delta}$. Suppose we draw $\lceil n/\epsilon\rceil$ samples from the source distribution μ, and use v to compute the empirical transported distribution $\hat{p}$. Without loss of generality, suppose $\hat{d}$ is achieved at dimension j, i.e.,

$$\hat{d} = |\hat{p}_j - \nu_j|/\nu_j \geq |\hat{p}_k - \nu_k|/\nu_k, \qquad \forall k \in [n],\ k \neq j \tag{11}$$

Let $d^* \overset{\text{def}}{=} \texttt{MRE}(\nu, p) = \max_{i=1}^n\{|p_i - \nu_i|/\nu_i\}$. Without loss of generality, suppose d^* is achieved at dimension i, i.e.,

$$d^* = |p_i - \nu_i|/\nu_i \geq |p_k - \nu_k|/\nu_k, \qquad \forall k \in [n],\ k \neq i \tag{12}$$

Let $\nu_{\min}$ denote the minimum value in ν. Hence,

$$\begin{aligned}
d^* &= |p_i - \nu_i|/\nu_i \\
&= \left|\left(\frac{p_i}{\nu_i} - 1\right) - \left(\frac{\hat{p}_i}{\nu_i} - 1\right) + \left(\frac{\hat{p}_i}{\nu_i} - 1\right)\right| \\
&\leq \frac{|p_i - \hat{p}_i|}{\nu_i} + \left|\frac{\hat{p}_i}{\nu_i} - 1\right| \\
&\leq \frac{\kappa\sqrt{\epsilon p_i/n}}{\nu_i} + \hat{d} \text{ with probability of at least } 1 - \delta, \text{ by applying } \textbf{Lemma } 1 \text{ and Eq. } (11) \\
&= \kappa\sqrt{\frac{\epsilon}{n\nu_i}} \cdot \sqrt{\frac{p_i}{\nu_i}} + \hat{d} \\
&\leq \kappa\sqrt{\frac{\epsilon}{n\nu_{\min}}} \cdot \sqrt{\frac{p_i}{\nu_i}} + \hat{d}
\end{aligned} \tag{13}$$

Therefore, with probability of at least $1 - \delta$, we have

$$d^* = \left|\frac{p_i}{\nu_i} - 1\right| \leq \kappa\sqrt{\frac{\epsilon}{n\nu_{\min}}} \cdot \sqrt{\frac{p_i}{\nu_i}} + \hat{d} \tag{14}$$

To find the upper bound d^*, we can find the upper bound of $\sqrt{\frac{p_i}{\nu_i}}$. Let $a = \kappa\sqrt{\frac{\epsilon}{n\nu_{\min}}}$ and $x = \sqrt{\frac{p_i}{\nu_i}}$. According to Eq. (14), we have $|x^2 - 1| \le ax + \hat{d}$. If $x \ge 1$, the right hand side gets larger values than the case $x < 1$. Since we are computing the upper bound, we just need to consider the case $x \ge 1$. We need to consider the maximum x that satisfies

$$x^2 - 1 \le ax + \hat{d} \tag{15}$$

The maximum x that satisfies the above inequality is

$$\frac{a + \sqrt{a^2 + 4\hat{d} + 4}}{2} \tag{16}$$

Therefore, d^* has an upper bound of

$$\hat{d} + \frac{\sqrt{a^4 + 4\hat{d}a^2 + 4a^2} + a^2}{2} \tag{17}$$

for any $a > 0$. If $\hat{d} < \epsilon_*$, to ensure the precision ϵ_* can be achieved, we want to choose an ϵ such that the upper bound is less than or equal to ϵ_*, i.e.,

$$\hat{d} + \frac{\sqrt{a^4 + 4\hat{d}a^2 + 4a^2} + a^2}{2} \le \epsilon_* \tag{18}$$

To satisfy Eq. (18), let $\hat{d} = \beta\epsilon_*,\ 0 < \beta < 1$. Substituting $\hat{d}$ by $\beta\epsilon_*$ and a by $\kappa\sqrt{\frac{\epsilon}{n\nu_{\min}}}$ in Eq. (18) , we have

$$\frac{\epsilon}{n\nu_{\min}} \le \left(\frac{1-\beta}{\kappa}\right)^2 \frac{\epsilon_*^2}{1+\epsilon_*} \tag{19}$$

To get the lower bound,

$$\begin{aligned}
d^* &= |p_i - \nu_i|/\nu_i \\
&\ge |p_j - \nu_j|/\nu_j \\
&= \left|\left(\frac{p_j}{\nu_j} - 1\right) - \left(\frac{\hat{p}_j}{\nu_j} - 1\right) + \left(\frac{\hat{p}_j}{\nu_j} - 1\right)\right| \\
&\ge \left|\left(\frac{\hat{p}_j}{\nu_j} - 1\right)\right| - \left|\left(\frac{p_j}{\nu_j} - 1\right) - \left(\frac{\hat{p}_j}{\nu_j} - 1\right)\right| \\
&\ge \hat{d} - \frac{\kappa\sqrt{\epsilon p_j/n}}{\nu_j} \ \text{ with probability of at least } 1-\delta, \text{ by applying } \textbf{Lemma } 1 \\
&= \hat{d} - \kappa\sqrt{\frac{\epsilon}{n\nu_j}} \cdot \sqrt{\frac{p_j}{\nu_j}} \\
&\ge \hat{d} - \kappa\sqrt{\frac{\epsilon}{n\nu_{\min}}} \cdot \sqrt{\frac{p_j}{\nu_j}}
\end{aligned} \tag{20}$$

Therefore, with probability of at least $1 - \delta$, we have

$$d^* \ge \left|\frac{p_j}{\nu_j} - 1\right| \ge \hat{d} - \kappa\sqrt{\frac{\epsilon}{n\nu_{\min}}} \cdot \sqrt{\frac{p_j}{\nu_j}} \tag{21}$$

To find the lower bound of d^*, we can find the minimum $|p_i/\nu_i - 1|$ that satisfies Eq. (21). Let $y = \sqrt{\frac{p_j}{\nu_j}}$. According to Eq. (21), we have $|y^2 - 1| \ge \hat{d} - ay$. To analyze the minimum value of $|y^2 - 1|$, we need to analyze two cases: i) $\hat{d} < a$, ii) $\hat{d} \ge a$.

i) $\hat{d} < a$.

There exists an $y_0 = 1$ and $|y_0^2 - 1| \ge \hat{d} - ay_0$ holds. Therefore, the minimum value of $|y^2 - 1|$ is 0 in the case $\hat{d} < a$. Thus, the lower bound of d^* is 0 in this case.

ii) $\hat{d} \ge a$.

In this case, we consider $y \geq 1$ and $y < 1$.

For $y \geq 1$, we consider the following inequality:

$$y^2 - 1 \geq \hat{d} - ay \tag{22}$$

Let

$$y_{\geq 1} = \frac{-a + \sqrt{a^2 + 4\hat{d} + 4}}{2} \tag{23}$$

Since $\hat{d} \geq a$, we can derive from Eq. (23) and obtain $y_{\geq 1}$ greater than or equal to 1. It can be verfied that $y_{\geq 1}^2 - 1 = \hat{d} - ay_{\geq 1}$, and $y_{\geq 1}$ is the minimum y that is ≥ 1 and satisfies Eq. (22). Therefore, $y_{\geq 1}^2 - 1$ is the lower bound of d^* for $y \geq 1$ in the case $\hat{d} \geq a$.

For $y < 1$, we consider the following inequality:

$$1 - y^2 \geq \hat{d} - ay \tag{24}$$

If there is no y that satisfies Eq. (24), we just need to consider $y \geq 1$ for the case $\hat{d} \geq a$. Otherwise, denote the maximum y that is less than 1 and satisfies Eq. (24) by $y_{<1}$. So, we have $1 - y_{<1}^2 = \hat{d} - ay_{<1}$. Since $\hat{d} - ay_{<1} > \hat{d} - ay_{\geq 1}$, the lower bound of d^* is always achieved when $y \geq 1$ for the case $\hat{d} \geq a$. Therefore, the lower bound of d^* in the case $\hat{d} \geq a$ is $\hat{d} - ay_{\geq 1}$ which is equivalent to:

$$\hat{d} - \frac{\sqrt{a^4 + 4\hat{d}a^2 + 4a^2} - a^2}{2} \tag{25}$$

It is easy to verify that if $\hat{d} < a$ then Eq. (25) is less than 0, and vice versa. Therefore, we can combine case i) $\hat{d} < a$, and case ii) $\hat{d} \geq a$ as:

$$\max\left(\hat{d} - \frac{\sqrt{a^4 + 4\hat{d}a^2 + 4a^2} - a^2}{2}, 0\right) \tag{26}$$

Eq. (26) is the lower bound of d^*. To make the lower bound non-trivial, i.e. Eq. (25) ≥ 0, we want to choose an ϵ that achieves $\hat{d} \geq a$. Substituting a by $\kappa\sqrt{\frac{\epsilon}{n\nu_{\min}}}$ and $\hat{d}$ by $\beta\epsilon_*$, we obtain

$$\frac{\epsilon}{n\nu_{\min}} \leq \left(\frac{\beta\epsilon_*}{\kappa}\right)^2 \tag{27}$$

We should choose an ϵ that is large enough for efficiency consideration and at the same time satisfies both Eq. (19) and Eq. (27). This leads to optimizing the following problem

$$\max_{\beta} \quad \min\left(\left(\frac{1-\beta}{\kappa}\right)^2 \frac{\epsilon_*^2}{1+\epsilon_*}, \frac{\beta^2\epsilon_*^2}{\kappa^2}\right) \tag{28}$$

The solution to the above problem is

$$\beta = \frac{1}{\sqrt{1+\epsilon_*}+1} \tag{29}$$

Therefore, the optimal ϵ is

$$\epsilon = \frac{\epsilon_*^2}{\kappa^2(\sqrt{1+\epsilon_*}+1)^2} \cdot n\nu_{\min} \tag{30}$$

and the optimal a is

$$a = \frac{\epsilon_*}{\sqrt{1+\epsilon_*}+1} \tag{31}$$

Let $\xi = \frac{\epsilon_*^2}{4(\sqrt{1+\epsilon_*}+1)^2}$. Thus, $\xi = a^2/4$ and $\epsilon = 4\delta\xi n\nu_{\min}$. Substituting a^2 in Eq. (25) by 4ξ and considering the trivial lower bound 0, we obtain the lower bound in 1). Substituting a^2 in Eq. (17) by 4ξ and considering the trivial upper bound $(1 - \nu_{\min})/\nu_{\min}$, we obtain the upper bound in 2)..

3) As we draw the samples from μ, we are essentially drawing samples from the real target distribution p achieved by v. The number of samples is $\lceil n/\epsilon \rceil$. Applying Lemma 1 1), we obtain the desired bound of $\|\hat{p} - p\|_1$.

4) $\|\hat{g} - g\|_1 = \|(\hat{p} - \nu) - (p - \nu)\|_1 = \|\hat{p} - p\|_1 \leq \Delta$ by applying 3).

5) $\|p - \nu\|_1 = \|p - \hat{p} + \hat{p} - \nu\|_1$. Using the triangle inequality on the upper bound we have $\|p - \nu\|_1 \leq \|p - \hat{p}\|_1 + \|\hat{p} - \nu\|_1 \leq d_1 + \Delta$. As p and ν are two probability measures, $\|p - \nu\|_1$ is upper bounded by 2. Using the triangle inequality on the lower bound we have $\|p - \nu\|_1 \geq \|\hat{p} - \nu\|_1 - \|p - \hat{p}\|_1 \geq d_1 - \Delta$. As any norm is greater than or equal to 0, we have the desired lower bound. □

A.2.2 Proof of Theorem 2

Before proving Theorem 2, it is necessary to give the bound of the distance between the initial SDOT output v_0, and the optimal SDOT solution v_*. The bound of the distance will be used in the convergence analysis in Theorem 2 and 3. The following theorem shows the range of an optimal SDOT solution v_*.

Theorem 5. *There exists at least one optimal solution, denoted by v_*, in $[-C_\infty/2, C_\infty/2]^n$, that maximizes Eq. (2).*

Proof. Maximizing Eq. (2) is equivalent to solving the following problem:

Problem 2. *Given two bounded domains X and $\hat{Y} = \{y_i\}_{i=1}^n$ and their probability measures $\mu \in \mathbb{P}(X)$, $\nu \in \mathbb{P}(\hat{Y})$, respectively, find a function u and a vector v to solve*

$$\begin{aligned} \max_{u,v} \quad & W(u,v) = -\int u(x)d\mu(x) + \sum_{i\in\mathcal{I}} v_{(i)}\nu_i \\ s.\ t. \quad & \qquad -u(x) + v_{(i)} \leq c(x, y_i) \\ & \qquad \forall x \in X,\ \forall y_i \in \hat{Y} \end{aligned} \tag{32}$$

If (u_*, v_*) is the optimal solution to Problem 2, then $\forall r \in \mathbb{R}$, $(u_* - r, v_* - r)$ is also an optimal solution to Problem 2, because $(u_* - r, v_* - r)$ does not change the Laguerre cell decomposition computed by (u_*, v_*). Let r be the minimum value in (u_*, v_*), and let $u_* \leftarrow u_* - r$ and $v_* \leftarrow v_* - r$. So, the minimum value in (u_*, v_*) is 0.

Next, we show that

$$\inf_{i\in\mathcal{I}} v_{*,i} \geq \inf_{x\in X} u_*(x) \qquad \text{and} \qquad \sup_{i\in\mathcal{I}} v_{*,i} \geq \sup_{x\in X} u_*(x) \tag{33}$$

We prove Eq. (33) by contradiction. If $\inf_{i\in\mathcal{I}} v_{*,i} < \inf_{x\in X} u_*(x)$. Without loss of generality, let $v_{*,k} < \inf_{x\in X} u_*(x)$. We can construct a new v_{**}, in which $v_{**,i} = v_{*,i}, i \neq k$ and $v_{**,k} = \inf_{x\in X} u_*(x)$. The new solution (u_*, v_{**}) does not violate the constraints in Eq. (32), and increases the objective. Therefore, (u_*, v_*) is not the optimal solution to Problem 2 which is a contradiction. Thus, $\inf_{i\in\mathcal{I}} v_{*,i} \geq \inf_{x\in X} u_*(x)$. $\sup_{i\in\mathcal{I}} v_{*,i} \geq \sup_{x\in X} u_*(x)$ can be proved in a similar way (by decreasing the maximum value in $u_*(x)$ to the maximum value in v_*). Recall the minimum value in (u_*, v_*) equals 0, we have $\inf_{x\in X} u_*(x) = 0$. Recall the constraint $-u(x) + v_{(i)} \leq c(x, y_i)$, $\forall x \in X$, $\forall y_i \in \hat{Y}$, we have $\sup_{i\in\mathcal{I}} v_{*,i} \leq C_\infty$, where C_∞ is the maximum transport cost. Therefore, every element in (u_*, v_*) lies between 0 and C_∞. Let $u_{***} = u_* - C_\infty/2$ and $v_{***} = v_* - C_\infty/2$. Thus, v_{***} is an optimal solution that maximizes Eq. (2) and v_{***} is in $[-C_\infty/2, C_\infty/2]^n$.

□

Let v_0 be 0, and v_* be the optimal solution that maximizes Eq. (2) and lies in $[-C_\infty/2, C_\infty/2]^n$. Then,

$$\|v_0 - v_*\| \leq \frac{\sqrt{n}C_\infty}{2} \tag{34}$$

Theorem 2 is to establish the optimal step size γ given a fixed number of iterations T. Before proving Theorem 2, we introduce the following Proposition in Bach (2014).

Proposition 1. *(Bach, 2014) Assume 1) f is convex and three-times differentiable, 2) f has a global minimum attained at $v_* \in \mathcal{V}$, 3) all gradients of f and f_t, are bounded by R, i.e., $\|f'(v)\| \leq R$, and $\|f_t'(v_{t-1})\| \leq R$ almost surely. 4) $\forall t \geq 1$, f_t is $\mathcal{F}$-measurable, 5) $\mathbb{E}[f_t'(v_{t-1})|\mathcal{F}_{t-1}] = f'(v_{t-1})$ and 6) $v_t = v_{t-1} - \gamma_t f_t'(v_{t-1})$, where $(\gamma_t)_{t\geq 1}$ is a deterministic sequence. Let $\bar{v} = \frac{1}{T}\sum_{t=1}^{T} v_{t-1}$. With constant step size equal to γ, for any $T \geq 0$, we have*

$$\mathbb{E}\left[2\gamma T(f(\bar{v}) - f(v_*)) + \|v_T - v_*\|^2\right] \leq \|v_0 - v_*\|^2 + T\gamma^2 R^2 \tag{35}$$

and

$$\mathbb{E}\left[2\gamma T(f(\bar{v}) - f(v_*)) + \|v_T - v_*\|^2\right]^2 \leq \left(\|v_0 - v_*\|^2 + 9T\gamma^2 R^2\right)^2 \tag{36}$$

We restate Theorem 2 below:

Theorem 2. *With the constant step size*

$$\gamma = \frac{\sqrt{n}C_\infty + 1}{12\sqrt{T}} \tag{37}$$

the expectation of the gradient norm of function f at $\bar{v}$ is bounded by:

$$\mathbb{E}\left\|f'(\bar{v})\right\|^2 \leq \frac{4}{T}\left(\frac{4}{\sqrt{T}} + 10 + 6\sqrt{n}C_\infty\right)^2 \tag{38}$$

The proof of Theorem 3 is similar to the proof proposed in Bach (2014), but uses a tighter bound introduced in the Proposition 1.

Proof. **First, we bound $\frac{1}{T}\sum_{t=1}^{T} f'(v_{t-1})$.**

For fixed step size update $v_t = v_{t-1} - \gamma f_t'(v_{t-1})$, we have

$$f_t'(v_{t-1}) = \frac{1}{\gamma}(v_{t-1} - v_t) \tag{39}$$

Summing from 1 to the total number of iterations T, we get

$$\frac{1}{T}\sum_{t=1}^{T} f'(v_{t-1}) = \frac{1}{T}\sum_{t=1}^{T}[f'(v_{t-1}) - f_t'(v_{t-1})] + \frac{1}{\gamma T}(v_0 - v_*) + \frac{1}{\gamma T}(v_* - v_T) \tag{40}$$

Similar to Bach (2014), we apply the Burkholder-Rosenthal-Pinelis inequality (Pinelis, 1994) [Theorem 4.1] to bound

$$\left[\mathbb{E}\left\|\frac{1}{T}\sum_{t=1}^{T}[f'(v_{t-1}) - f_t'(v_{t-1})]\right\|^2\right]^{1/2} \leq \frac{4R}{T} + \frac{2\sqrt{2}R}{\sqrt{T}} \tag{41}$$

Using Proposition 1 and Minkowski's inequality:

$$\begin{aligned}
\left[\mathbb{E}\left\|\frac{1}{T}\sum_{t=1}^{T} f'(v_{t-1})\right\|^2\right]^{1/2} &\leq \left[\mathbb{E}\left\|\frac{1}{T}\sum_{t=1}^{T}[f'(v_{t-1}) - f_t'(v_{t-1})]\right\|^2\right]^{1/2} \\
&\quad + \frac{1}{\gamma T}\|v_0 - v_*\| + \frac{1}{\gamma T}\left[\mathbb{E}\|v_T - v_*\|^2\right]^{1/2} \\
&\leq \frac{4R}{T} + \frac{2\sqrt{2}R}{\sqrt{T}} + \frac{1}{\gamma T}\|v_0 - v_*\| + \frac{1}{\gamma T}\sqrt{\|v_0 - v_*\|^2 + T\gamma^2 R^2} \\
&\leq \frac{4R}{T} + \frac{2\sqrt{2}R}{\sqrt{T}} + \frac{1}{\gamma T}\|v_0 - v_*\| + \frac{\|v_0 - v_*\|}{\gamma T} + \frac{\sqrt{T}\gamma R}{\gamma T} \\
&\leq \frac{4R}{T} + \frac{4R}{\sqrt{T}} + \frac{2\|v_0 - v_*\|}{\gamma T}
\end{aligned} \tag{42}$$

Next, we bound $\frac{1}{T}\sum_{t=1}^{T} f'(v_{t-1}) - f'(\frac{1}{T}\sum_{t=1}^{T} v_{t-1})$

By using the self-concordance property in Bach (2014) [D.2], we have $\|\frac{1}{T}\sum_{t=1}^{T} f'(v_{t-1}) - f'(\frac{1}{T}\sum_{t=1}^{T} v_{t-1})\| \le 2R(\frac{1}{T}\sum_{t=1}^{T} f(v_{t-1}) - f(v_*))$.

Using Proposition 1, we have

$$\begin{aligned}\left[\mathbb{E}\left\|\frac{1}{T}\sum_{t=1}^{T} f'(v_{t-1}) - f'\left(\frac{1}{T}\sum_{t=1}^{T} v_{t-1}\right)\right\|^2\right]^{1/2} &\le 2R\left(\mathbb{E}\left[\frac{1}{T}\sum_{t=1}^{T} f(v_{t-1}) - f(v_*)\right]^2\right)^{1/2} \\ &\le \frac{R}{\gamma T}\left(\|v_0 - v_*\|^2 + 9T\gamma^2 R^2\right)\end{aligned} \tag{43}$$

Summing Eq. (42) and Eq. (43), we get

$$\begin{aligned}\left[\mathbb{E}\left\|f'\left(\frac{1}{T}\sum_{t=1}^{T} v_{t-1}\right)\right\|^2\right]^{1/2} &\le \frac{4R}{T} + \frac{4R}{\sqrt{T}} + \frac{2\|v_0 - v_*\|}{\gamma T} + \frac{R\|v_0 - v_*\|^2}{\gamma T} + 9\gamma R^3 \\ &\le \frac{R}{\sqrt{T}}\left[\frac{4}{\sqrt{T}} + 4 + 9R^2\gamma\sqrt{T} + \frac{(\|v_0 - v_*\| + 1/R)^2}{\gamma\sqrt{T}}\right]\end{aligned} \tag{44}$$

If

$$\gamma = \frac{\|v_0 - v_*\| + 1/R}{3R\sqrt{T}} \tag{45}$$

the upper bound of $\left[\mathbb{E}\left\|f'\left(\frac{1}{T}\sum_{t=1}^{T} v_{t-1}\right)\right\|^2\right]^{1/2}$ is minimized. For optimal transport, the gradient is the difference between two distributions. Thus, the gradient norm is upper bounded by 2. Therefore, we choose $R = 2$. Substituting $\|v_0 - v_*\|$ in Eq. (45) by its upper bound in Eq. (34), we get

$$\gamma = \frac{\sqrt{n}C_\infty + 1}{12\sqrt{T}} \tag{46}$$

Take γ, R and the upper bound of $\|v_0 - v_*\|$ into Eq. (44), we obtain Eq. (38). □

A.2.3 Proof of Theorem 3

Theorem 3 establishes an optimal total number of iterations T and the corresponding optimal step size γ, such that we can achieve an expected value of the MRE for any given $\epsilon > 0$.

Theorem 3. *Suppose Assumption 1 is satisfied, $\forall \epsilon > 0$, with constant step size*

$$\gamma = \frac{\epsilon \nu_{\min}}{24} \cdot \frac{(1 + \sqrt{n}C_\infty)}{(14 + 6\sqrt{n}C_\infty)} \tag{47}$$

and number of iterations

$$T = \frac{4(14 + 6\sqrt{n}C_\infty)^2}{\epsilon^2 \nu_{\min}^2} \tag{48}$$

we have

$$\mathbb{E}\left[\mathit{MRE}\left(\nu, p\right)\right] \le \epsilon \tag{49}$$

	1D toy data	2D toy data	256D toy data	256D real face data
MC-Adam	1e-5	1e-4	1e-2	1e-4
Vanilla-EGD	1e-2	1e0	1e+2	1e+3

Table 2: Parameter settings of MC-Adam and Vanilla-EGD for different data.

Proof. Let p be the probability density function achieved by $\bar{v}$.

$$\begin{aligned}
\mathbb{E}[\mathtt{MRE}(\nu, p)] &= \mathbb{E}\left[\max_{i\in\mathcal{I}} \frac{|p_i - \nu_i|}{\nu_i}\right] \\
&\leq \mathbb{E}\left[\max_{i\in\mathcal{I}} \frac{|p_i - \nu_i|}{\nu_{\min}}\right] \\
&= \mathbb{E}\left[\frac{\|f'(\bar{v})\|_\infty}{\nu_{\min}}\right] \\
&\leq \mathbb{E}\left[\frac{\|f'(\bar{v})\|}{\nu_{\min}}\right] \\
&\leq \frac{1}{\nu_{\min}}\sqrt{\mathbb{E}\|f'(\bar{v})\|^2} \qquad \text{Jensen's inequality}
\end{aligned} \tag{50}$$

Therefore, if $\mathbb{E}\|f'(\bar{v})\|^2 \leq \epsilon^2\nu_{\min}^2$, then we have $\mathbb{E}[\mathtt{MRE}(\nu, p)] \leq \epsilon$. According to Eq. (38), we have

$$\begin{aligned}
\mathbb{E}\left\|f'(\bar{v})\right\|^2 &\leq \frac{4}{T}\left(\frac{4}{\sqrt{T}} + 10 + 6\sqrt{n}C_\infty\right)^2 \\
&\leq \frac{4}{T}\left(14 + 6\sqrt{n}C_\infty\right)^2
\end{aligned} \tag{51}$$

Let

$$\frac{4}{T}\left(14 + 6\sqrt{n}C_\infty\right)^2 = \epsilon^2\nu_{\min}^2 \tag{52}$$

We obtain Eq. (48). Combining Eq. (37) and Eq. (48), we obtain Eq. (47). With γ in Eq. (47) and T in Eq. (48), we have $\mathbb{E}[\mathtt{MRE}(\nu, p)] \leq \epsilon$. Hence, we ensure the convergence of MRE for any given ϵ. □

A.3 Experiments

In the experiments, we compare SDOT-EGD against ASGD (Peyré et al., 2019; Aude et al., 2016), 2-layer ASGD (Leclaire & Rabin, 2020), MC-Adam (An et al., 2020a) and Vanilla-EGD (Hazan & Kale, 2014). MC-Adam uses the Monte-Carlo sampling to estimate the gradient, and uses the Adam optimizer to optimize the SDOT dual objective. The Vanilla-EGD uses an epoch gradient descent strategy with a fixed number of iterations in each epoch, and decreases the learning rate by half in each epoch.

A.3.1 Experimental Settings

In all the experiments, we set $\epsilon_* = 0.2$ and the confidence $1 - \delta$ to 0.9 in SDOT-EGD, except in the ablation study part. In all the experiments, we use the quadratic transport cost for all the methods, i.e., $c(x, y) = \|x - y\|^2$. For ASGD, 2-layer ASGD, we use the default parameter settings, because the initial learning rate 1.0 is widely used in the literature (Leclaire & Rabin, 2019; 2020). we tuned the learning rate for the Adam optimizer in MC-Adam and the initial learning rate for Vanilla-EGD. The learning rate for Adam in MC-Adam is tuned in {1e-5, 1e-4, 1e-3, 1e-2, 5e-2, 1e-1} and the initial learning rate for Vanilla-EGD is tuned in {1e-3, 1e-2, 1e-1, 1e0, 1e+1, 1e+2, 1e+3}. We list the best learning rates for both methods in Table 2.

In the GAN experiments, we use the same network architecture as in Mescheder et al. (2018) for 256×256 size images with the maximum number of filters in each layer set to 512 for all the methods. We train all the models for 400K iterations with a batch size of 16 on the CelebA-HQ and the COVOC datasets. We use the RMSProp optimizer (Tieleman & Hinton, 2012) with a learning rate of $1e$-4 for all models. We use the uniform distribution in $[-1, 1]^{256}$ as the simple distribution to train all the

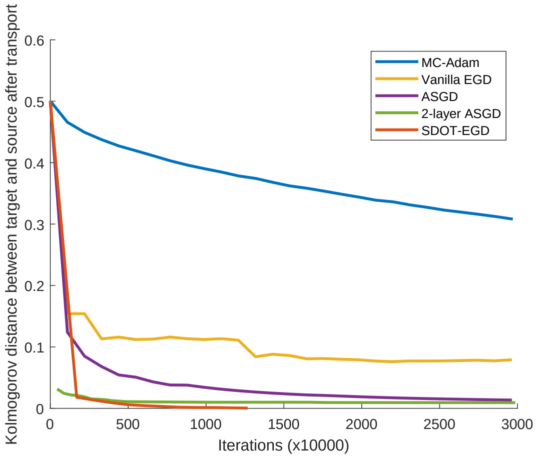


Figure 6: The Kolmogorov distance for different methods.

GANs. For WGAN-GP (SDOT-EGD) and GAN-0GP (SDOT-EGD), we use the method described in Sec. 5.3 in the main paper to obtain the 256D feature for each image. We compute the SDOT map using SDOT-EGD to map each noise point in $[-1, 1]^{256}$ to a 256D image feature. As each 256D image feature is extracted from each image, we have the matching between each noise point and each real image. This matching is fixed and is used throughout the WGAN-GP (SDOT-EGD) and GAN-0GP (SDOT-EGD) training process. The procedure of how we train a GAN with SDOT-EGD is listed below:

1 Given a dataset of images, we compute the features of these images.

2 We compute the SDOT map using SDOT-EGD between a uniform distribution and the features in the latent space.

3 Suppose batch size is m in GAN training. We randomly sample m noise points from the uniform distribution. Using the SDOT map, we find the corresponding m image features. Using these m image features, we find the corresponding m images. These m noise points and the m corresponding images form a batch.

4 We use the batch of m noise points and the m images to compute the discriminator and generator losses in a GAN, and then update the discriminator and generator weights.

5 Go to Step 3 if the training is not converged.

All the experiments are executed on a cluster of machines. Each machine has 4 Intel Xeon CPUs and 128GB RAM. All the GAN experiments are executed on Quadro RTX 8000 GPUs.

A.3.2 1D Experiments

In the 1D experiments, we use the same data as used in the main paper. Similar to Leclaire & Rabin (2020), we adopt the Kolmogorov distance[3] between target and the transported distribution to evaluate the performance of different methods. Fig. 6 plots the Kolmogorov distances of the various methods. Compared to other methods, SDOT-EGD converges much faster. It also converges to a much lower value, almost 0. The 2-layer ASGD decreases faster than MC-Adam, ASGD and Vanilla-EGD, but remains constant at around 0.01.

To investigate the reproducibility of SDOT-EGD, we rerun SDOT-EGD 10 times and plot the mean `MRE`s and the standard deviations in Fig. 7 a). As the `MRE` decreases, the standard deviation decreases, meaning SDOT-EGD becomes more and more stable. When SDOT-EGD terminates, the standard deviation is very small.

[3] The Kolmogorov distance is the L_∞ distance between two cumulative distribution functions.

Number of iterations/million	5	10	14
ASGD	0.333026	0.333480	0.333549
MC-Adam	0.004680	0.008253	0.011736
Vanilla-EGD	0.333501	0.333580	0.333591
SDOT-EGD	0.333659	**0.333667**	**0.333667**

Table 3: The SDOT dual objectives of different methods w.r.t number of iterations. Ground truth is 0.333667.

Time/seconds	250	500	700
ASGD	0.333314	0.333563	0.333610
MC-Adam	0.023410	0.051604	0.066314
Vanilla-EGD	0.333444	0.333551	0.333580
SDOT-EGD	0.333659	**0.333667**	**0.333667**

Table 4: The SDOT dual objectives of different methods w.r.t time. Ground truth is 0.333667.

We provide a benchmark in the 1D case. In the 1D case, the Ground Truth (GT) SDOT dual objective value is 0.333667. In the Tables 3 and 4, we report the SDOT dual objective values of different methods w.r.t. number of iterations and time, respectively. The 2-layer ASGD is not included because its objective is neither an SDOT primal nor dual objective. In both tables, SDOT-EGD achieves the ground truth of 0.333667 faster than all the other methods.

A.3.3 2D Experiments

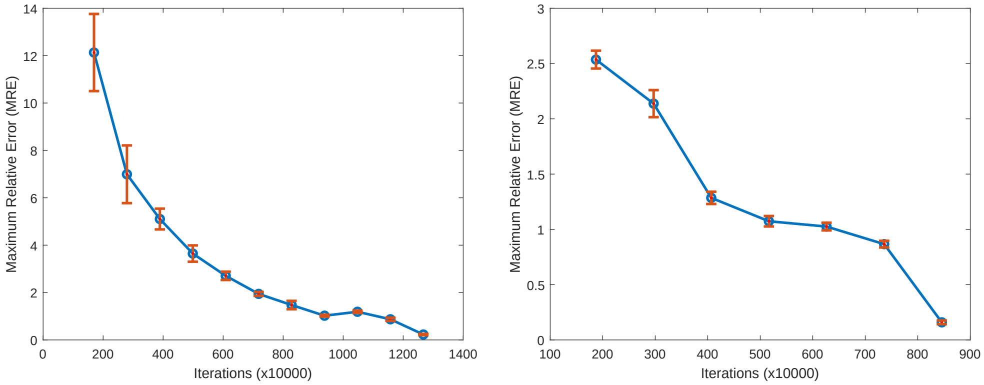
To investigate the reproducibility of SDOT-EGD, we plot the error bars of SDOT-EGD using 2D data. The source distribution μ is uniformly distributed in $[0, 1] \times [0, 1]$. We randomly sample 1000 points in $[0, 1] \times [0, 1]$ as target points, with each point having probability value of 1e-3. We rerun SDOT-EGD 10 times and plot the mean estimated $\texttt{MRE}$s and the standard deviations in Fig. 7 b). As we can see in this figure, SDOT-EGD has very small standard deviations across multiple runs. Fig. 7 a) and b) together show that SDOT-EGD is a stable approach.

To make it easier to assess the quality of the cell decomposition results, we use a 40×40 grid of 1600 points within $[0, 1] \times [0, 1]$ as target points shown in Fig. 8 f). We call it 2D grid data. They are equally spaced horizontally and vertically. Each data point has a probability of $1/1600$. So, an optimal SDOT algorithm should divide the $[0, 1] \times [0, 1]$ space into a grid of 40×40 square cells.

Fig. 8 shows the cell decomposition results. Cells are colored per their area from small (blue) to large (red). Cell decomposition from our method SDOT-EGD look more similar to a regular grid and is more uniform compared to other methods. MC-Adam and ASGD have some cells that are obviously large. The vanilla-EGD has small cells in the center and large cells in the corners. The 2-layer ASGD has a high cell area variance and many cells are not square. The Vanilla-EGD and the ASGD have the number of cells that are smaller than 1600, indicating that some target points do not have area in the noise space that are mapped to them. $\texttt{MRE}$ est and L_1 est are estimated $\texttt{MRE}$ and L_1 distance computed according to our theoretical bounds. The $\texttt{MRE}$ est corresponds very well with the quality of the Laguerre cell decomposition. SDOT-EGD has much lower $\texttt{MRE}$ est and L_1 est, suggesting it is better than other methods.

A.3.4 256D Experiments

We show the performance of different methods for 256D toy data. The source distribution μ is uniformly distributed in $[0, 1]^{256}$. We randomly sample 1000 points in $[0, 1]^{256}$, and assign the probability value of $1/1000$ to each point. Fig. 9 a) shows that SDOT-EGD terminated at around 2M iterations achieving an $\texttt{MRE}$ near 0, which is considerably faster than all the other methods. Fig. 9 b) shows that SDOT-EGD converges to a much lower L_1 distance efficiently compared to all the other methods.



a) b)

Figure 7: The error bar plot for a) 1D data and b) 2D data. The caps mark the standard deviations.

A.3.5 Comparison In Terms of Running Time

We also compare the performance of different methods in terms of running time in Fig. 10 and 11. From these figures we can observe that the proposed SDOT-EGD can reach lowest MREs or the L_1 distances in shorter time compared to other methods. Note that the shaded area in Fig. 10 (a) is too thin to be seen because the average upper bound and lower bound gap is smaller than 0.2.

It is worth mentioning that estimating the MRE is not expensive. It only involves sampling and matrix computations, which can be sped up on GPUs. In our 256D real face data experiments, $\nu_{\min} = 3.3e-5$ and $\epsilon_* = 0.2$. The MRE estimation process in SDOT-EGD only takes up approximately 5% of the total time.

A.3.6 Ablation Study

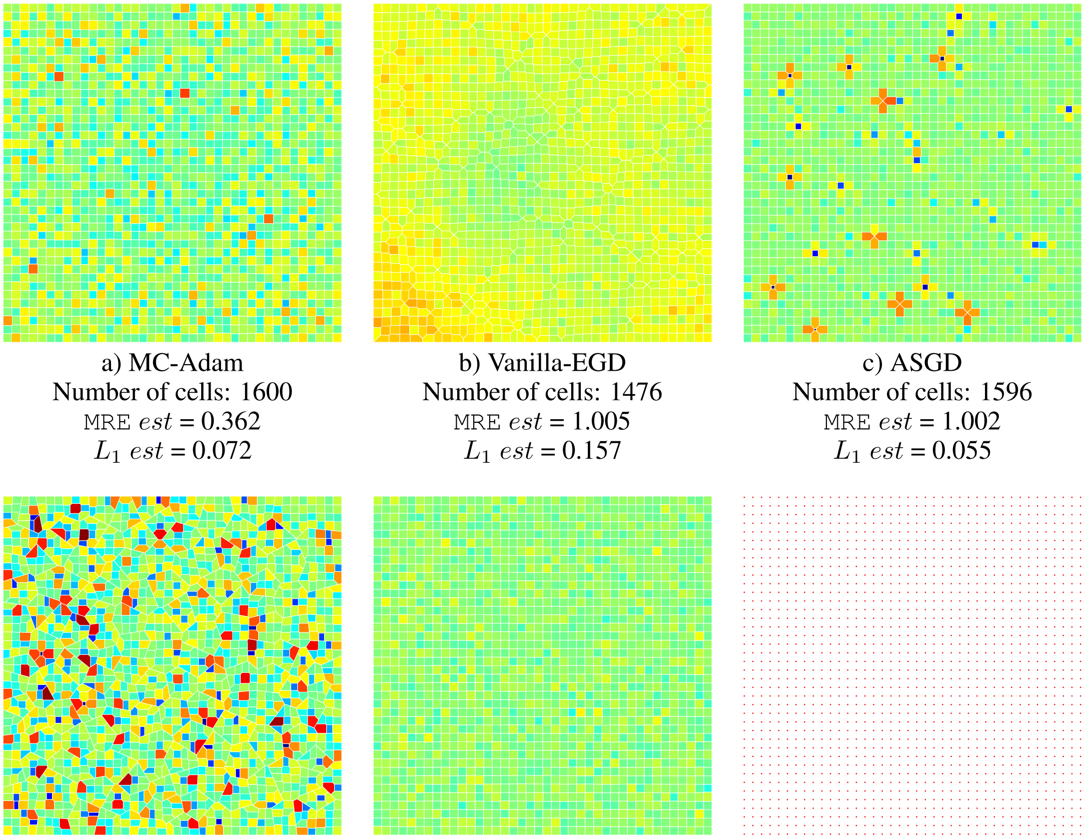
In this experiment, we investigate how different ϵ_* and confidence $(1-\delta)$ affect the performance of SDOT-EGD. We perform experiments using 2D data with 1000 target points.

Fig. 12 a) and b) show the performance of SDOT-EGD with ϵ_* fixed to 0.2 and various confidence values. SDOT-EGD can always converge to 0.2 under different confidence values. When the confidence is higher, SDOT-EGD requires a higher number of iterations. Fig. 12 c) and d) show the performances of SDOT-EGD with the confidence fixed to 0.9 and various ϵ_*. SDOT-EGD can always achieve a preset precision ϵ_* ranging from 0.1 to 0.9. The smaller ϵ_*, the higher the number of iterations required. As SDOT-EGD can always converge under different parameter settings, SDOT-EGD is not sensitive to parameter selection. In all the other experiments, we simply set ϵ_* to 0.2 and the confidence $1-\delta$ to 0.9.

A.3.7 GAN Experiments

In this section, we show more generated images by different methods. We conduct experiments on the CelebA-HQ dataset (Karras et al., 2018) and the COVID-19 Outcome (COVOC) dataset (Konwer et al., 2021). The CelebA-HQ dataset (CC BY-NC 4.0 License) contains 29970 unique images. The original image size is 1024×1024. In our experiments, similar to Liu et al. (2019), we resize the images to 256×256 to train GANs. The COVOC dataset contains 248 COVID-19 X-ray images acquired upon disease presentation. We also use 256×256 image size on the COVOC dataset to train GANs. We use the code as used in Mescheder et al. (2018) (MIT License).

Even though the number of images of the COVOC dataset is small, it is still okay to test on this dataset. We want to test how much SDOT can benefit GAN training when only a small number of training images are available. Generating images on the X-ray image dataset could benefit downstream tasks such as COVID-19 diagnosis (Rasheed et al., 2021), pneumonia diagnosis (Srivastav et al., 2021), chest pathology classification (Salehinejad et al., 2018), and so on. Training a good GAN on this



a) MC-Adam
Number of cells: 1600
MRE est = 0.362
L_1 est = 0.072

b) Vanilla-EGD
Number of cells: 1476
MRE est = 1.005
L_1 est = 0.157

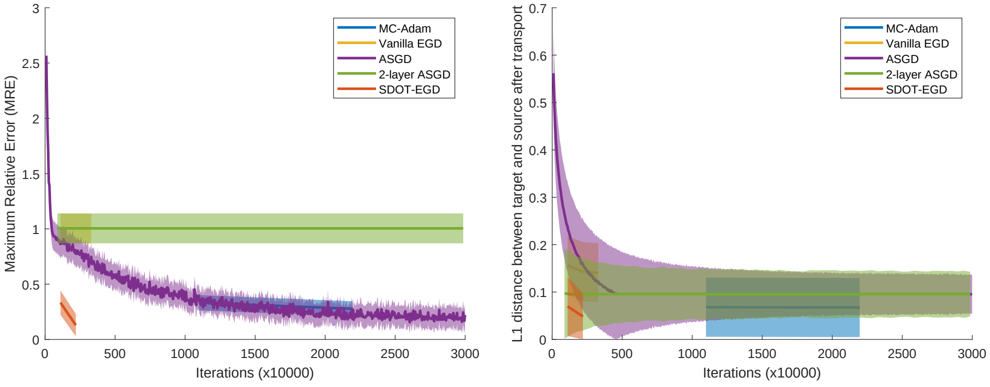
c) ASGD
Number of cells: 1596
MRE est = 1.002
L_1 est = 0.055

d) 2-layer ASGD
Number of cells: 1600
MRE est = 0.863
L_1 est = 0.131

e) SDOT-EGD
Number of cells: 1600
MRE est = 0.183
L_1 est = 0.050

f) Target distribution.
A 40×40 grid of 1600 points.

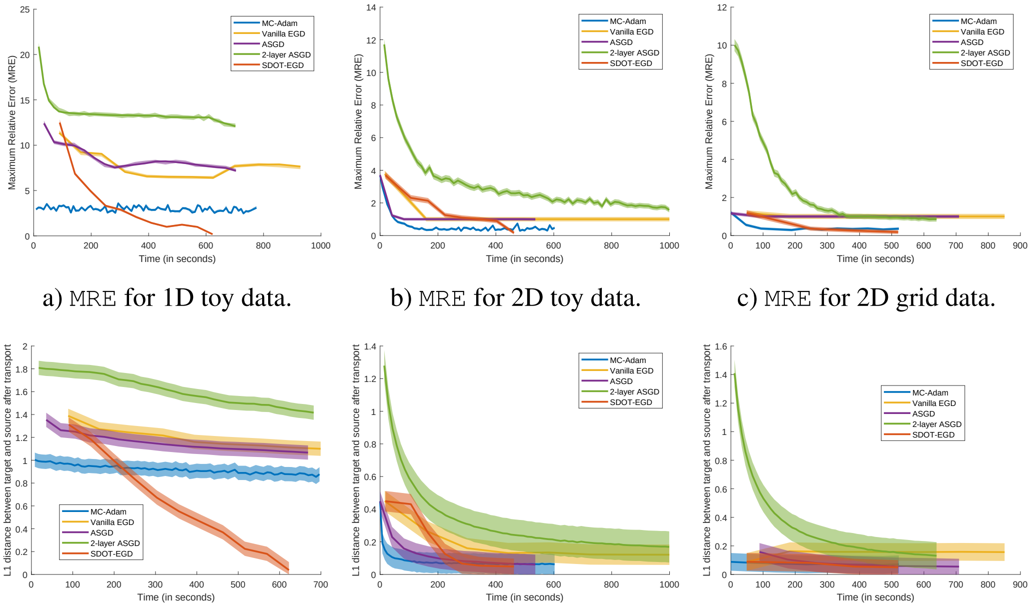
Figure 8: Laguerre cell decomposition for a 40×40 grid of 1600 data points. Each data point has probability of $1/1600$. Cells are colored per their area from small (blue) to large (red). Cell decomposition from our method SDOT-EGD look more similar to a regular grid and is more uniform compared to other methods. MC-Adam and ASGD have some cells that are obviously large. The vanilla-EGD has small cells in the center and large cells in the corners. The 2-layer ASGD high cell area variance and many cells are not square. The Vanilla-EGD and the ASGD have the number of cells that are smaller than 1600, indicating that some target points do not have area in the noise space that are mapped to them. MRE est and L_1 est are estimated MRE and L_1 distance computed according to our theoretical bounds. The MRE est corresponds very well with the quality of the Laguerre cell decomposition. SDOT-EGD has much lower MRE est and L_1 est, suggesting it is better than other methods.



a) MRE

b) L_1 distance

Figure 9: Comparison of MRE and the L_1 distance for 256D toy data.



a) MRE for 1D toy data.

b) MRE for 2D toy data.

c) MRE for 2D grid data.

d) L_1 distance for 1D toy data.

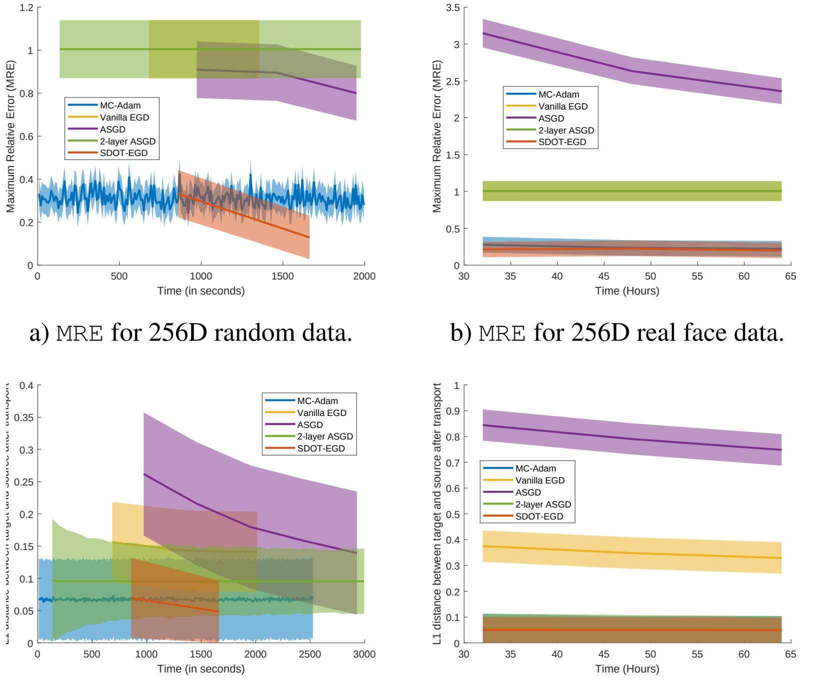
e) L_1 distance for 2D toy data.

f) L_1 distance for 2D grid data.

Figure 10: Comparison of MRE and L_1 distance in 1D and 2D toy data, and 256D real face data in terms of running time.

dataset is not easy because a lot of X-ray images look similar, and thus it is a fine-grained image generation problem. FID is used because it is a commonly used measure to evaluate the quality of the images generated by GANs (Heusel et al., 2017; Brock et al., 2019). Also, a thorough study was done by Segal et al. (2021) showing the evidence that the FID could be used to measure the quality of the chest X-ray images.

The only difference between WGAN-GP and WGAN-GP (SDOT-EGD) is that in WGAN-GP the noise and real images are randomly matched in a GAN training iteration, but in WGAN-GP (SDOT-EGD) the noise and the real images are matched using the SDOT matched GAN training mechanism described in the main paper. The only difference between GAN-0GP and GAN-0GP (SDOT-EGD) is the similar to the difference between WGAN-GP and WGAN-GP (SDOT-EGD).



a) MRE for 256D random data.

b) MRE for 256D real face data.

c) L_1 distance for 256D random data.

d) L_1 distance for 256D real face data.

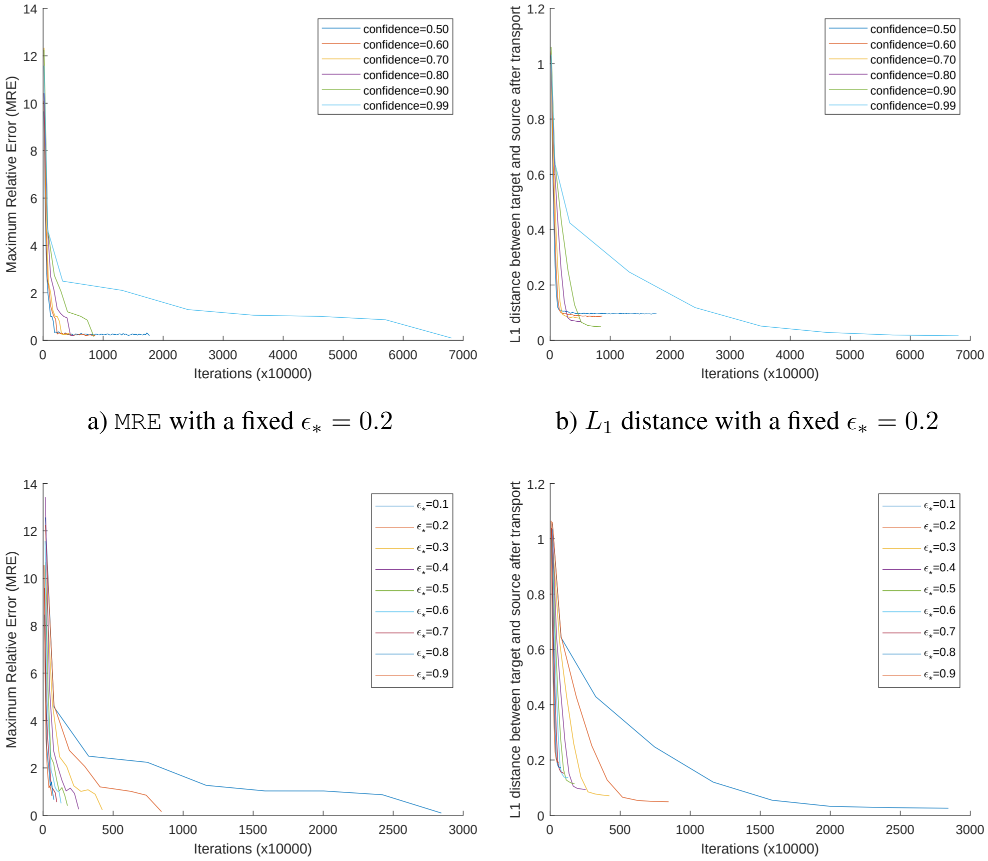
Figure 11: Comparison of MRE and L_1 distance in 256D toy data, and 256D real face data in terms of running time.

Fig. 13 a) and b) show the randomly generated images by WGAN-GP, and WGAN-GP (SDOT-EGD), respectively. WGAN-GP generates several faces with bad quality, whereas WGAN-GP (SDOT-EGD) generates better faces compared to WGAN-GP. Fig. 14 a) and b) show the randomly generated faces by GAN-0GP and GAN-0GP (SDOT-EGD), respectively. GAN-0GP (SDOT-EGD) generates much better faces than GAN-0GP, with every face having good quality. Fig. 15 shows the interpolation results between faces. Transitions between faces look realistic and smooth.

To show that ignoring samples in SDOT does lead to mode collapse in practice, we design a simple experiment. We use eight real face images, shown in Fig. 16 a), as training images and compute an SDOT map between the noise distribution and the eight 16×16 downsampled gray images using SDOT-EGD. To mimic the situation that one target image does not have a source area that maps to it, we took the dual variable $v_{(0)}$ for image 0 (The first image in Fig. 16 a)) to be $-\infty$ such that image 0 does not have an area in the source domain that maps to it. We train GAN-0GP using this new SDOT map. After the model is trained, we randomly generate 64 images, shown in Fig. 16 b). We checked these 64 images and found that, for all other 7 images in the training set, we can find that there are generated images that resemble them, and there is no generated image that resembles image 0. This indicates that mode collapse happens. Note that due to a small number of training samples, 8 samples in Fig. 16 a), a GAN is expected to be overfitting.

Fig. 17 shows the Covid-19 X-ray images randomly generated by WGAN-GP and WGAN-GP (SDOT-EGD). The images generated by both methods appear comparable. Fig. 18 shows the Covid-19 X-ray images randomly generated by GAN-0GP and GAN-0GP (SDOT-EGD). In Fig. 18 a), GAN-0GP generates several images with bad quality (row 2 column 3, and row 3 column 3). GAN-0GP (SDOT-EGD) generates much better images compared to GAN-0GP.

Qualitative results on both face images and Covid-19 X-ray images show the effectiveness of using SDOT to boost GAN training performance.



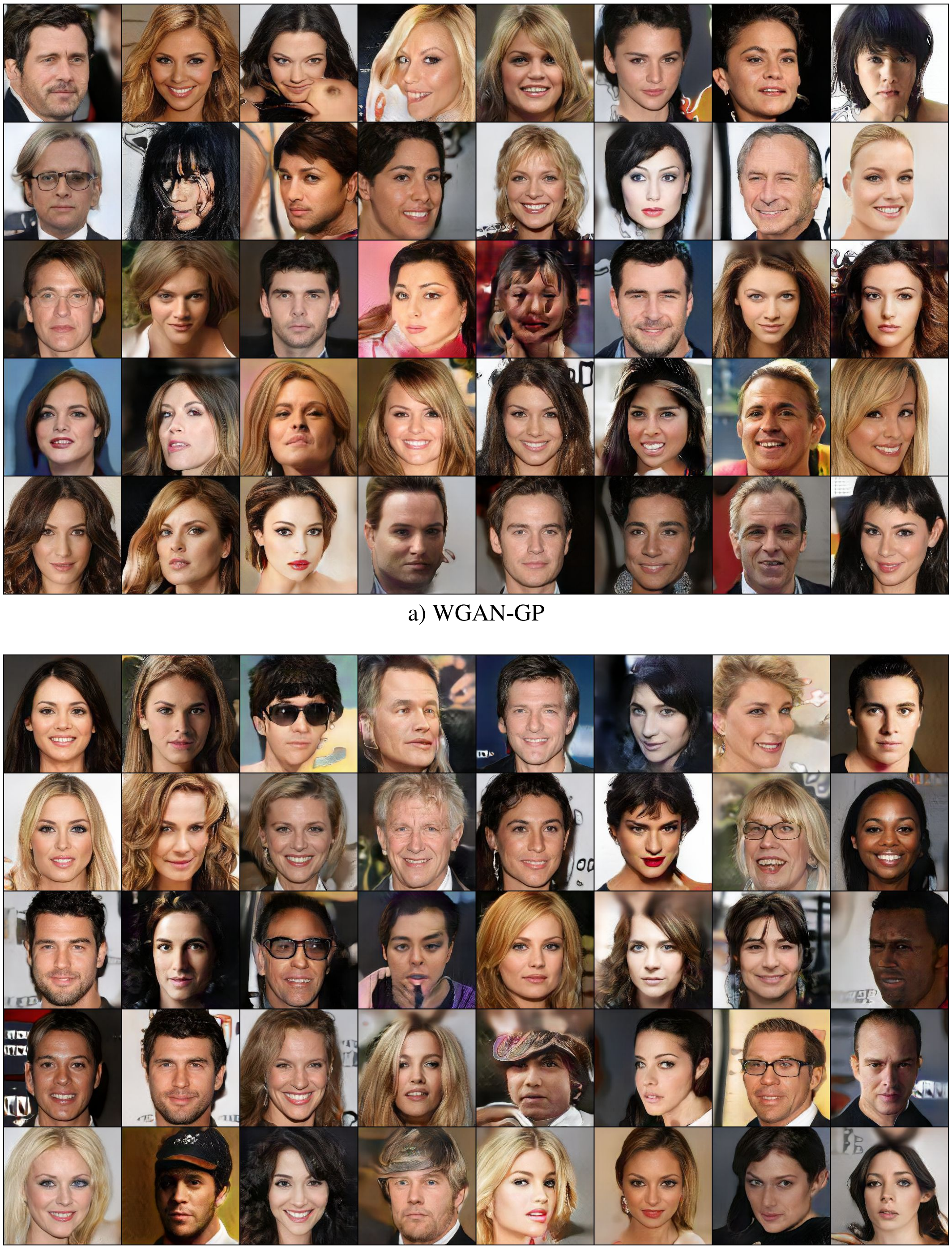
a) `MRE` with a fixed $\epsilon_* = 0.2$

b) L_1 distance with a fixed $\epsilon_* = 0.2$

c) `MRE` with a fixed confidence $= 0.9$

d) L_1 distance with a fixed confidence $= 0.9$

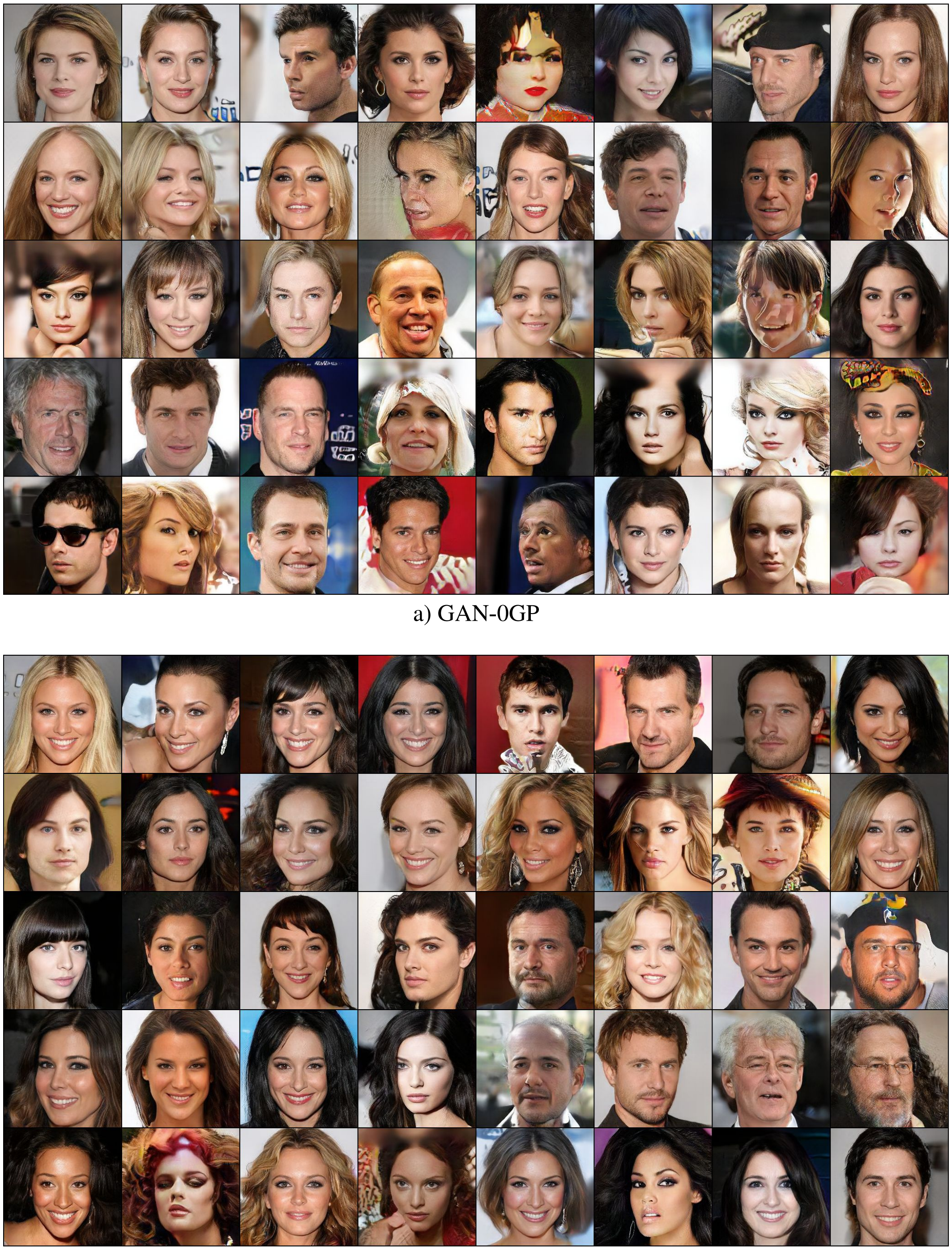
Figure 12: a) and b) show the performance of SDOT-EGD with ϵ_* fixed to 0.2, and various confidence values. SDOT-EGD can always converge to 0.2 under different confidence values. When the confidence is higher, SDOT-EGD requires a higher number of iterations. c) and d) show the performance of SDOT-EGD with the confidence fixed to 0.9, and various ϵ_*. SDOT-EGD can always achieve a preset precision ϵ_* ranging from 0.1 to 0.9. The smaller ϵ_*, the higher the number of iterations required. As SDOT-EGD can always converge under different parameter settings, SDOT-EGD is not sensitive to parameter selection. In all the other experiments, we simply set ϵ_* to 0.2 and the confidence $1 - \delta$ to 0.9.



a) WGAN-GP

b) WGAN-GP (SDOT-EGD)

Figure 13: Images randomly generated by a) WGAN-GP and b) WGAN-GP (SDOT-EGD). WGAN-GP generates several faces with bad quality, whereas WGAN-GP (SDOT-EGD) generates better faces compared to WGAN-GP.



a) GAN-0GP

b) GAN-0GP (SDOT-EGD)

Figure 14: Images randomly generated by a) GAN-0GP and b) GAN-0GP (SDOT-EGD). GAN-0GP (SDOT-EGD) generates much better faces than GAN-0GP, with every face having good quality.

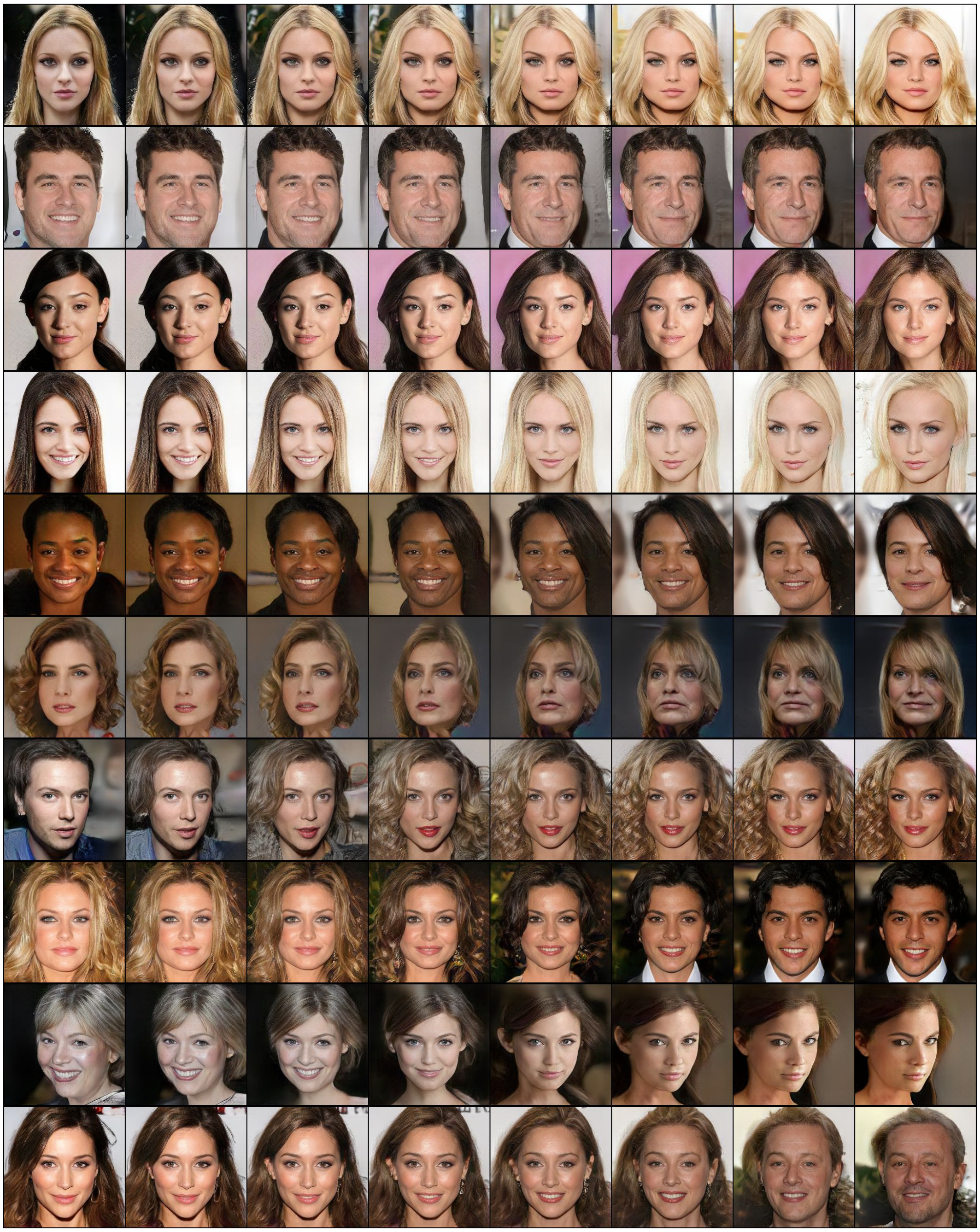
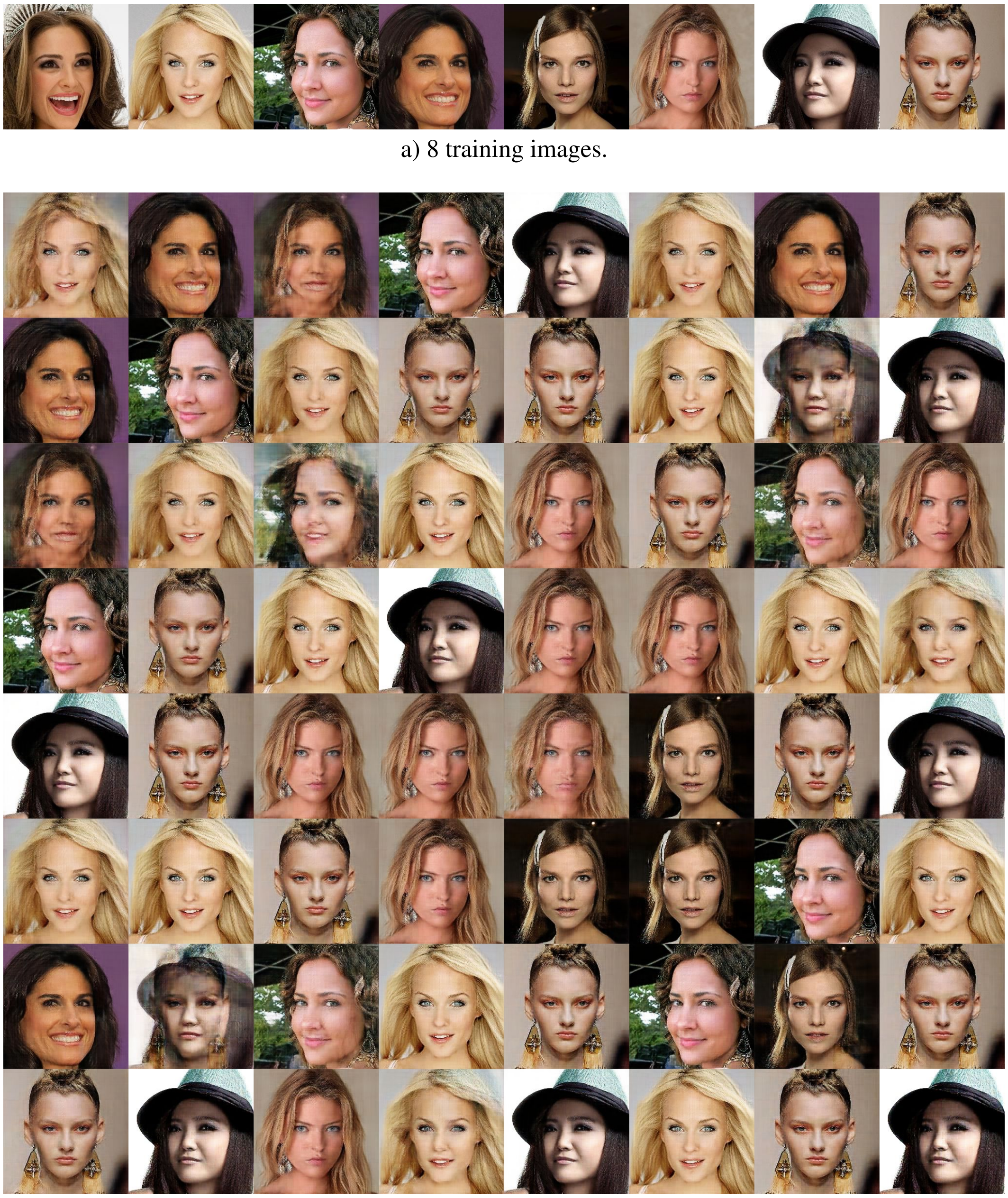


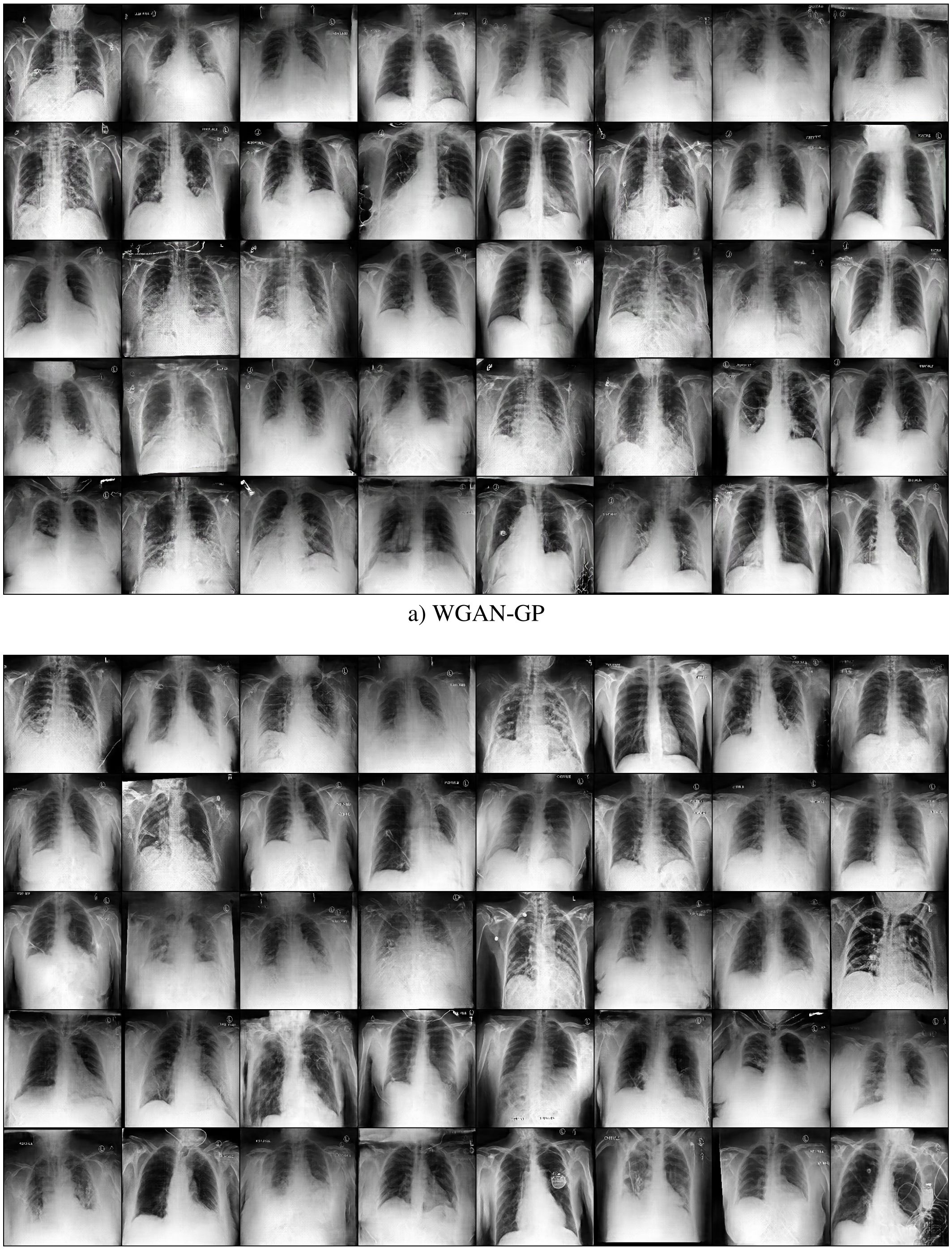
Figure 15: Face interpolation by GAN-0GP (SDOT-EGD). Transitions between faces appear good.



a) 8 training images.

b) Randomly sampled images.

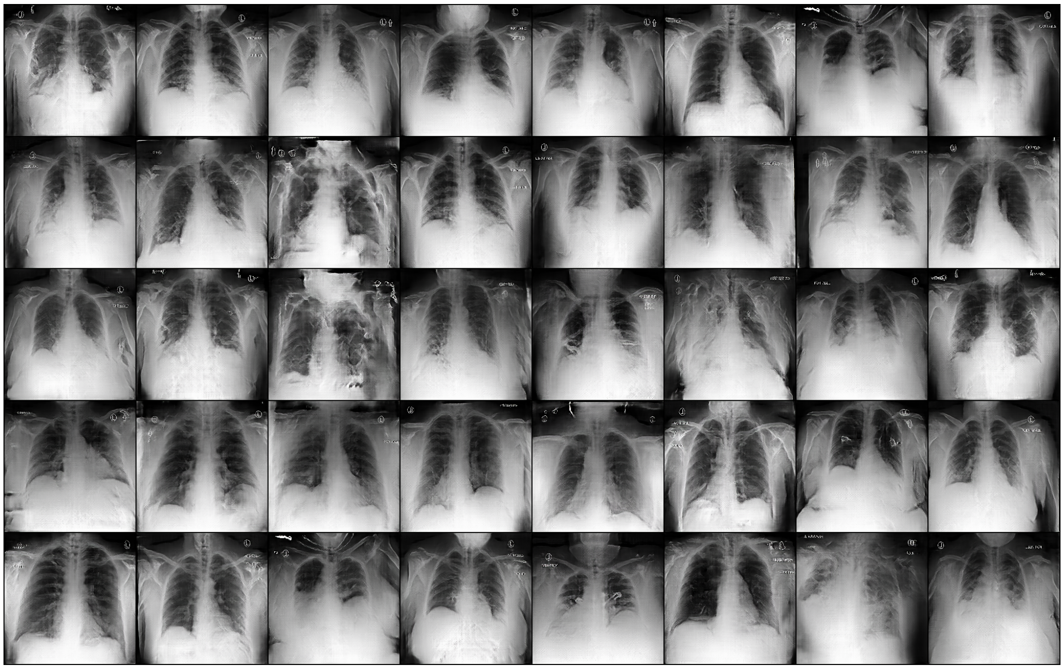
Figure 16: A simple experiment showing mode collapse happens in a GAN with SDOT if there is an image, the first image in a), that does not have any area in the noise space that maps to it in SDOT. In b), there is no generated image that resembles the first image in a). Note that due to a small number of training samples, 8 samples in a), a GAN is expected to be overfitting.



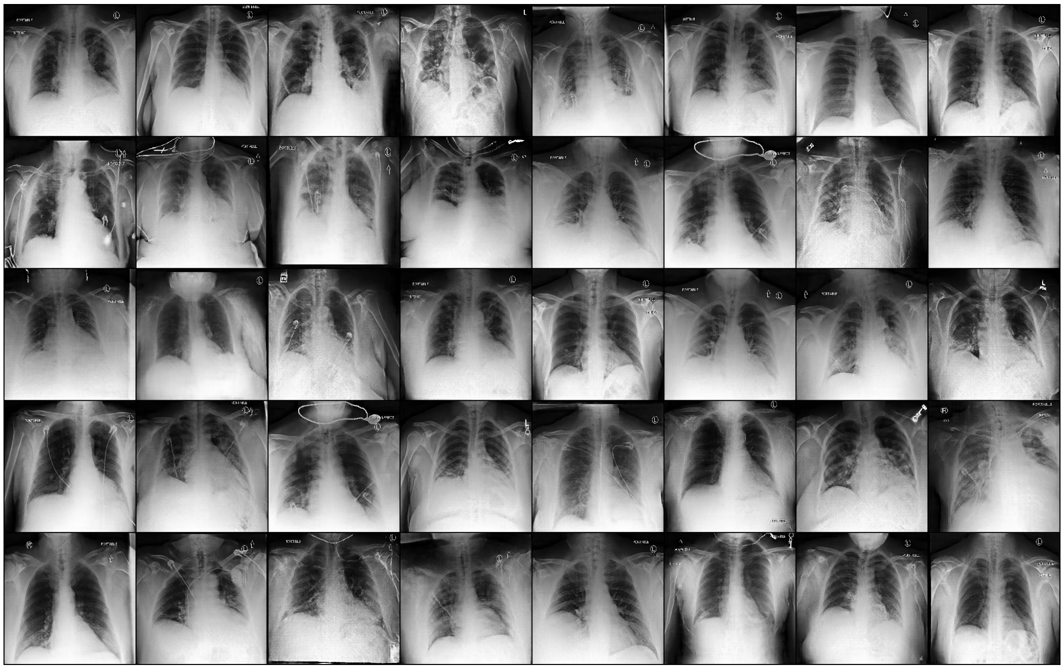
a) WGAN-GP

b) WGAN-GP (SDOT-EGD)

Figure 17: Images randomly generated by a) WGAN-GP and b) WGAN-GP (SDOT-EGD). The images generated by both methods appear comparable.



a) GAN-0GP



b) GAN-0GP (SDOT-EGD)

Figure 18: Images randomly generated by a) GAN-0GP and b) GAN-0GP (SDOT-EGD). GAN-0GP generates several images with bad quality (row 2 column 3, and row 3 column 3 in a)). GAN-0GP (SDOT-EGD) generates much better images compared to GAN-0GP.

A.4 DISCUSSION

In the SDOT GAN application part, we use the downsized images as features in the latent space. This kind of features may not be optimal to represent the data manifold. Interestingly, even with such a simple mapping, “downsampling + RGB to gray”, we achieved significantly better results compared to GANs without using the SDOT matching mechanism. An alternative approach could be to use an Auto-Encoder to compute the image features. How to find a better mapping for GAN with SDOT-EGD to further improve the GAN training performance is very interesting future work. It is worth noting that such a mapping is only used to find image features. It is well-accepted that the images lie in a low-dimensional space, a.k.a. the latent space. Given a dataset of images, there exists a mapping that maps the images to the latent space. Once the mapping is computed on the dataset, it should be fixed and thus it is deterministic.

When using SDOT-EGD, the source domain X should be bounded such that C_∞ is well-defined. In practice, using a Gaussian as a continuous distribution as the source distribution for SDOT-EGD is possible. We can truncate the Gaussian at 3 or 6 standard deviations from the mean according to the precision we need. Using the truncated is a reasonable approximation to train GANs. For example, it has been shown in BigGAN (Brock et al., 2019), that using a truncated Gaussian can generate high fidelity images compared to using the original un-truncated Gaussian.