AWGAN: Empowering High-Dimensional Discriminator Output for Generative Adversarial Networks

Mengyu Dai, 1 Haibin Hang, 2 Anuj Srivastava 3

1 Microsoft
2 University of Delaware
3 Florida State University, Department of Statistics
mendai@microsoft.com, haibin@udel.edu, anuj@stat.fsu.edu

Abstract
Empirically multidimensional discriminator (critic) output can be advantageous, while a solid explanation for it has not been discussed. In this paper, (i) we rigorously prove that high-dimensional critic output has advantage on distinguishing real and fake distributions; (ii) we also introduce an square-root velocity transformation (SRVT) block which further magnifies this advantage. The proof is based on our proposed maximal $p$-centrality discrepancy which is bounded above by $p$-Wasserstein distance and perfectly fits the Wasserstein GAN framework with high-dimensional critic output. We have also showed when $n = 1$, the proposed discrepancy is equivalent to 1-Wasserstein distance. The SRVT block is applied to break the symmetric structure of high-dimensional critic output and improve the generalization capability of the discriminator network. In terms of implementation, the proposed framework does not require additional hyper-parameter tuning, which largely facilitates its usage. Experiments on image generation tasks show performance improvement on benchmark datasets.

1 Introduction
Generative Adversarial Networks (GAN) have led to numerous success stories in various tasks in recent years (Niemeyer and Geiger 2021; Chan et al. 2021; Han, Chen, and Liu 2021; Karras et al. 2020; Nauata et al. 2020; Heim 2019; Zhan, Zhu, and Lu 2019). The goal in a GAN framework is to learn a distribution (and generate fake data) that is as close to real data distribution as possible. This is achieved by playing a two-player game, in which a generator and a discriminator compete with each other and try to reach a Nash equilibrium (Goodfellow et al. 2014). Arjovsky et al. (Arjovsky and Bottou 2017; Arjovsky, Chintala, and Bottou 2017) pointed out the shortcomings of using Jensen-Shannon Divergence in formulating the objective function, and proposed using the 1-Wasserstein distance instead. Numerous promising frameworks (Li et al. 2017; Mroueh, Sercu, and Goel 2017; Mroueh and Sercu 2017; Mroueh et al. 2017; Wu et al. 2019; Deshpande et al. 2019; Ansari, Scarlett, and Soh 2020) based on other discrepancies were developed afterwards. Although some of these works use dimension of critic output $n = 1$, empirical evidence can be found that using multiple dimension $n$ could be advantageous. For examples, in (Li et al. 2017) authors pick different $n$s (16, 64, 128) for different datasets; In Sphere GAN (Park and Kwon 2019) their ablation study shows the best performance with $n = 1024$. However, the reason for this phenomenon has not been well explored yet.

One of the contribution of this paper is to rigorously prove that high-dimensional critic output is advantageous in the revised WGAN framework. Particularly, we propose a new metric on the space of probability distributions, called maximal $p$-centrality discrepancy. This metric is closely related with $p$-Wasserstein distance (Theorem 3.8) and can serve as a replacement of the objective function in WGAN framework especially when the discriminator has high-dimensional output. In this revised WGAN framework we are able to prove that using high-dimensional critic output makes discriminator more sensitive on distinguishing real and fake distributions (Proposition 3.10). In classical WGAN with only one critic output, the discriminator push-forwards (or project) real and fake distributions to 1-dimensional space, and then look at their maximal mean discrepancy. This 1-dimensional push-forward may hide significant differences of distributions in the shadow. Even though ideally there exists a “perfect” push-forward which reveals any tiny differences, practically the discriminator has difficulties to reach that global optimal push-forward. However, using $p$-centrality allows to push-forward distributions to higher dimensional space. Since even an average high-dimensional push-forward may reveal more differences than a good 1-dimensional push-forward, this reduces the burden on discriminator.

Another novelty of our framework is to break the symmetry structure of the discriminator network by compositing with an asymmetrical square-root velocity transformation (SRVT). In the general architecture of GAN, people assume that the output layer of discriminator is fully connected. This setup puts all output neurons in equal and symmetric positions. As a result, any permutation of the high-dimensional output vector will leave the value of objective function unchanged. This permutation symmetry implies that the weights connected to output layer are somehow correlated and this would undermine the generalization power of the discriminator network (Liang et al. 2019;
After adding the asymmetrical SRVT block, each output neuron would be structurally unique (Proposition 3.13). Our intuition is that the structural uniqueness of output neurons would imply their functionality uniqueness. This way, different output neurons are forced to reflect distinct features of input distribution. Hence SRVT serves as an magnifier which takes full use of high-dimensional critic output. Since the proposed framework utilizes an asymmetric transformation SRVT in a general WGAN framework, we name it as AWGAN.

In terms of implementation, our experiments show performance improvement on unconditional and conditional image generation tasks. The novelty of our work is summarized as follows:

1. We propose a metric between probability distributions as follows:
2. We utilize an asymmetrical (square-root velocity) transformation which breaks the symmetric structure of the discriminator network. We have conducted experiments to show its effectiveness in our work;
3. The proposed framework improves stability of training in the setting of high-dimensional critic output without introducing additional hyper tuning parameters.

2 Related work

Wasserstein Distance and Other Discrepancies Used in GAN: Arjovsky et al. (Arjovsky, Chintala, and Bottou 2017) applied Kantorovich-Rubinstein duality for 1-Wasserstein distance as loss function in GAN objective. WGAN makes great progress toward stable training compared with previous GANs, and marks the start of using Wasserstein distance in GAN. However, sometimes it still may converge to sub-optimal optima or fail to converge due to the raw realization of Lipschitz condition by weight clipping. To resolve these issues, researchers proposed sophisticated ways (Gulrajani et al. 2017; Wei et al. 2018; Miyato et al. 2018) to enforce Lipschitz condition for stable training. Recently, people come up with another way to involve Wasserstein distance in GAN (Wu et al. 2019; Kolouri et al. 2019; Deshpande, Zhang, and Schwing 2018; Lee et al. 2019). They use the Sliced Wasserstein Distance (Rabin et al. 2011; Kolouri, Zou, and Rohde 2016) to estimate the Wasserstein distance from samples based on a summation over the projections along random directions. Either of these methods rely on pushforwards of real and fake distributions through Lipschitz functions or projections on to 1-dimensional space. In our work, we attempt to distinguish two distributions by looking at their pushforwards in high dimensional space. This would add a lot of flexibility to convergence path which may prevents the minimizer getting stuck on poor local optimum.

Another way people used to distinguish real data and fake data distributions in generative network is by moment matching (Li, Swersky, and Zemel 2015; Dziugaite, Roy, and Ghahramani 2015). Particularly, in (Li et al. 2017) the authors used the kernel maximum mean discrepancy (MMD) in GAN objective, which aims to match infinite order of moments. In our work we propose to use the maximum discrepancy between p-centrality functions to measure the distance of two distributions. The p-centrality function (Definition 3.1) is exactly the p-th root of the p-th moment of a distribution. Hence, the maximal p-centrality discrepancy distance we propose can be viewed as an attempt to match the p-th moment for any given p ≥ 1.

p-Centrality Functions: The mean or expectation of a distribution is a basic statistic. Particularly, in Euclidean spaces, it is well known that the mean realizes the unique minimizer of the so-called Fréchet function of order 2 (cf. (Grove and Karcher 1973; Bhattacharya and Patrangenaru 2003; Arnaudon, Barbaresco, and Yang 2013)). Generally speaking, a Fréchet function of order p summarizes the p-th moment of a distribution with respect to any base point. A topological study of Fréchet functions is carried out in (Hang, Memoli, and Mio 2019) which shows that by taking p-th root of a Fréchet function, the p-centrality function can derive topological summaries of a distribution which is robust with respect to p-Wasserstein distance. In our work, we propose using p-centrality functions to build a nice discrepancy distance between distributions, which would benefit from its close connection with p-Wasserstein distance.

Asymmetrical Networks: Symmetries occur frequently in deep neural networks. By symmetry we refer to certain group actions on the weight parameter space which keep the objective function invariant. These symmetries would cause redundancy in the weight space and affects the generalization capacity of network (Liang et al. 2019; Badrinarayanan, Mishra, and Cipolla 2015). There are two types of symmetry: (i) permutation invariant; (ii) rescaling invariant. A straight forward way to break symmetry is by random initialization (cf. (Glorot and Bengio 2010; He et al. 2015)). Another way to break symmetry is via skip connections to add extra connections between nodes in different layers (He et al. 2015a,b; Huang et al. 2017). In our work, we attempt to break the permutation symmetry of the output layer in the discriminator using a nonparametric asymmetrical transformation specified by square-root velocity function (SRVF) (Srivastava et al. 2011; Srivastava and Klassen 2016). The simple transformation that converts functions into their SRVFs changes Fisher-Rao metric into the L2 norm, enabling efficient analysis of high-dimensional data. Since the discretised formulation of SRVF is equivalent with an non-fully connected network (as depicted in Fig. 2), our approach can be viewed as breaking symmetry by deleting specific connections from the network.

3 Proposed Framework

In this section we introduce the proposed GAN framework in which the objective function is built on the maximal p-centrality discrepancy and the discriminator is compositized with an SRVT block.

3.1 Objective Function

The objective function of the proposed GAN is as follows:

$$\min_G \max_D \left( E_x [ \| D(x) \|_p ]^{1/p} - E_z [ \| D(G(z)) \|_p ]^{1/p} \right)$$
For any two Borel probability measures \( \mu, \nu \) on \((M, d)\) and \( p \geq 1 \), the \( p \)-centrality function is defined as

\[
\sigma_{\mu, \nu}(x) := \left( \int_M d^p(x, y) d\nu(y) \right)^{\frac{1}{p}} = \left( \mathbb{E}_{y \sim \mu} [d^p(x, y)] \right)^{\frac{1}{p}}.
\]

Particularly, the value of \( p \)-centrality function at \( x \) is the \( p \)-th root of the \( p \)-th moment of \( \mu \) with respect to \( x \). As we know it, the \( p \)-th moments are important statistics of a probability distribution. After taking the \( p \)-th root, the \( p \)-centrality function retains those important information in \( p \)-th moments, and it also shows direct connection with the \( p \)-Wasserstein distance \( W_p \):

**Lemma 3.2.** For any \( x \in M \), let \( \delta_x \) be the Dirac measure centered at \( x \). Then \( \sigma_{\mu, \nu}(x) = W_p(\mu, \delta_x) \).

**Lemma 3.3.** For any two Borel probability measures \( \mu \) and \( \nu \) on \((M, d)\), we have

\[
\|\sigma_{\mu, \nu} - \sigma_{\nu, \mu}\|_{\infty} \leq W_p(\mu, \nu) \leq \|\sigma_{\mu, \nu} + \sigma_{\nu, \mu}\|_{\infty}.
\]

**Proof.** For any \( x \in M \), by Lemma 3.2 and triangle inequality we have

\[
|\sigma_{\mu, \nu}(x) - \sigma_{\nu, \mu}(x)| \leq W_p(\mu, \nu) \leq |\sigma_{\mu, \nu}(x) + \sigma_{\nu, \mu}(x)|.
\]

The result follows by letting \( x \) run over all \( M \).

Let \( \mathcal{P}(M) \) be the set of all probability measures on \( M \) and let \( C_0(M) \) be the set of all continuous functions on \( M \). We define an operator \( \Sigma_p : \mathcal{P}(M) \to C_0(M) \) s.t. \( \Sigma_p(\mu) = \sigma_{\mu, \mu} \).

The above lemma implies that \( \Sigma_p \) is 1-Lipschitz, which makes \( \Sigma_p \) a powerful indicator of a probability measure. Specifically, since \( p \)-Wasserstein distance \( W_p \) metrizes weak convergence when \((M, d)\) is compact, we have:

**Proposition 3.4.** If \((M, d)\) is compact and \( \mathcal{P} \) weakly converges to \( \mu \), then \( \sigma_{\mu, \nu} \) converges to \( \sigma_{\mu, \nu} \) with respect to \( L^\infty \) distance.

Recall that in WGAN, the discriminator is viewed as a \( K \)-Lipschitz function. In our understanding, this requirement is enforced to prevent the discriminator from distorting input distributions too much. More precisely, in the more general setting, the following is true:

**Proposition 3.5.** Given any \( K \)-Lipschitz map \( f : (M, d_M) \to (N, d_N) \) and Borel probability distributions \( \mu, \nu \in \mathcal{P}(M) \). Then the pushforward distributions \( f_*\mu, f_*\nu \in \mathcal{P}(N) \) satisfy

\[
W_p(f_*\mu, f_*\nu) \leq K : W_p(\mu, \nu).
\]

**Proof.** By the Kantorovich-Rubinstein duality, we have

\[
\inf_{\gamma \in \mathcal{P}(\mu, \nu)} \int_{M \times N} d_N^p(x, y) \gamma(dx, dy) = \sup_{f \in Lip_k(M \to N)} \mathbb{E}_{x \sim \mu} |f(x) - f(y)|.
\]
Lemma 3.6. The definition of $L_{p,n,K}$ is independent of the choice of the base point. Or simply

$$L_{p,n,K}(P, Q) = \sup_{f \in Lip(K)} \left( \int \|f\|^p dP \right)^{\frac{1}{p}} - \left( \int \|f\|^p dQ \right)^{\frac{1}{p}}.$$

Proof. Let $\phi$ be the translation map on $\mathbb{R}^n$ with $\phi(y) = y + x_0$. Then $g := \phi^{-1} \circ f \in Lip(K)$ iff. $f \in Lip(K)$ and

$$L_{p,n,K}(P, Q) = \sup_{f \in Lip(K)} |\sigma_{f, \phi, p}(\phi(0)) - \sigma_{f, \phi, p}(\phi(0))|$$

$$= \sup_{\phi \in Lip(K)} |\sigma_{\phi^{-1} \circ f, \phi, p}(\phi(0)) - \sigma_{\phi^{-1} \circ f, \phi, p}(\phi(0))|$$

$$= \sup_{\gamma \in Lip(K)} |\sigma_{\phi, \gamma, p}(\gamma(0)) - \sigma_{\phi, \gamma, p}(\gamma(0))|$$

$$= \sup_{\gamma \in Lip(K)} \left( \int \|f\|^p d\gamma \right)^{\frac{1}{p}} - \left( \int \|f\|^p d\gamma \right)^{\frac{1}{p}}.$$

The following proposition implies that $L_{n,p,K}$ is a direct generalization of Wasserstein distance:

Proposition 3.7. If $supp(P)$ and $supp(Q)$ are both compact, then

$$L_{1,1,K}(P, Q) = K \cdot W_1(P, Q).$$

Proof. Since $f \in Lip(K)$ implies $|f| \in Lip(K)$, we easily have $L_{1,1,K} \leq K \cdot W_1$.

On the other hand, for any $\epsilon > 0$, there exists a $K$-Lipschitz map $f : M \to \mathbb{R}$ s.t. $\int f dP - \int f dQ > K \cdot W_1(P, Q) - \epsilon$. Let $D = supp(P) \cup supp(Q)$ and $c = \min_{x \in D} f(x)$, then $f - c \geq 0$ and $\int f dP - \int f dQ = \int (f - c) dP - \int (f - c) dQ = \int |f - c| dP - \int |f - c| dQ \leq L_{1,1,K}(P, Q)$. Hence $L_{1,1,K}(P, Q) \geq K \cdot W_1(P, Q) - \epsilon$ for any $\epsilon > 0$ which implies $L_{1,1,K} \geq K \cdot W_1$.

More generally, $L_{n,p,K}$ is closely related with $p$-Wasserstein distance:

Theorem 3.8. For any $K$-Lipschitz map $f : M \to \mathbb{R}^n$,

$$L_{p,n,K}(P, Q) \leq K \cdot W(p, Q).$$

Proof. By Lemma 3.2, we have

$$L_{p,n,K}(P, Q) = \sup_{f \in Lip(K)} W_p(f, P, \delta_0) - W_p(f, Q, \delta_0).$$

Applying triangle inequality and Proposition 3.5, we have

$$L_{p,n,K}(P, Q) \leq \sup_{f \in Lip(K)} W_p(f, P, f, Q) \leq K \cdot W(p, Q).$$

Also $L_{n,p,K}$ is closely related with an $L^\infty$ distance:

Proposition 3.9. For any $K$-Lipschitz map $f : M \to \mathbb{R}^n$,

$$\|\sigma_{f, p, \epsilon} - \sigma_{f, \phi, p}\|_\infty \leq \max \{ L_{p,n,K}(P, Q), L_{p,n,K}(Q, P) \}.$$

Proof.

$$\|\sigma_{f, p, \epsilon} - \sigma_{f, \phi, p}\|_\infty$$

$$= \sup_{x_0 \in \mathbb{R}^n} |\sigma_{f, p, \epsilon}(x_0) - \sigma_{f, \phi, p}(x_0)|$$

$$\leq \sup_{x_0 \in \mathbb{R}^n} \sup_{f \in Lip(K)} |\sigma_{f, \phi, p}(x_0) - \sigma_{f, \phi, p}(x_0)|$$

$$= \sup_{x_0 \in \mathbb{R}^n} \max \{ L_{p,n,K}(P, Q), L_{p,n,K}(Q, P) \}.$$

The lower bound implies that, when we feed two distributions into the discriminator $f$, as long as some differences retained in the pushforwards $f_\epsilon$ and $f_\phi$, they would be detected by $L_{p,n,K}$. The upper bound implies that, if $P$ and $Q$ only differ a little bit under distance $W_p$, then $L_{p,n,K}(P, Q)$ would not change too much. Furthermore,

Proposition 3.10. If integers $n < n'$, then for any $P, Q \in \mathcal{P}(\mathbb{R}^n)$, we have $L_{p,n,K}(P, Q) \leq L_{p,n',K}(P, Q)$.

Proof. For any $n < n'$ we have natural embedding $\mathbb{R}^n \hookrightarrow \mathbb{R}^{n'}$. Hence any $K$-Lipschitz function with domain $\mathbb{R}^n$ can also be viewed as a $K$-Lipschitz function with domain $\mathbb{R}^{n'}$. Hence larger $n$ gives larger candidate pool for searching the maximal discrepancy and the result follows.

Hence the maximal $p$-centrality discrepancy $L_{p,n,K}$ becomes more sensitive to the differences between distributions with larger $n$.

3.3 Square Root Velocity Transformation

Proposition 3.10 suggests us to choose high-dimensional discriminator output to improve the performance of GAN. However, if the last layer of discriminator is fully connected, then all output neurons are in symmetric positions and the loss function is permutation invariant. Thus the generalization power of discriminator only depends on the equivalence class obtained by identifying each output vector with its permutations (Badrinarayanan, Mishra, and Cipolla 2015; Liang et al. 2019). Correspondingly the advantage of high-dimensional output vector would be significantly undermined. In order to further improve the performance of our proposed framework, we consider adding an SRVT block to the discriminator to break the symmetric structure. SRVT usually is used in shape analysis to define a distance between curves or functional data.

Particularly, we view the high-dimensional discriminator output $(x_1, x_2, \ldots, x_n)$ as an ordered sequence.

Definition 3.11. The signed square root function $Q : \mathbb{R} \to \mathbb{R}$ is given by $Q(x) = sgn(x) \sqrt{|x|}$.

Given any differentiable function $f : [0, 1] \to \mathbb{R}$, its SRVT is a function $q : [0, 1] \to \mathbb{R}$ with

$$q := Q \circ f' = sgn(f') \sqrt{|f'|}.$$ (2)

SRVT is invertible. Particularly, from $q$ we can recover $f$:

Lemma 3.12.

$$f(t) = f(0) + \int_0^t q(s) |q(s)| ds.$$ (3)
By assuming \( x_0 = 0 \), a discretized SRVT
\[
S : (x_1, x_2, \ldots, x_n) \in \mathbb{R}^n \mapsto (y_1, \ldots, y_n) \in \mathbb{R}^n
\]
is given by
\[
y_i = \text{sgn}(x_i - x_{i-1}) \sqrt{|x_i - x_{i-1}|}, i = 1, 2, 3, \ldots, n.
\]
Similarly, \( S^{-1} : \mathbb{R}^n \rightarrow \mathbb{R}^n \) is given by
\[
x_i = \sum_{j=1}^{i} y_j |y_j|, i = 1, 2, 3, \ldots, n.
\]

With this transformation, the pullback of \( L^2 \) norm gives
\[
\|(x_1, \ldots, x_n)\|_Q = \sqrt{\sum_{i=1}^{n} |x_i - x_{i-1}|}
\]
Applying SRVT on a high-dimensional vector results in an ordered sequence which captures the velocity difference at each consecutive position. The discretized SRVT can be represented as a neural network with activation function to be signed square root function \( Q \) as depicted in Fig 2. Particularly, for the purpose of our paper, each output neuron of SRVT is structurally unique:

**Proposition 3.13.** Any (directed graph) automorphism of the SRVT block leaves each output neuron fixed.

*Proof.* View the SRVT block as a directed graph, then all output neurons has out-degree 0. By the definition of discretized SRVT, there is a unique output neuron \( v_0 \) with in-degree 1 and any two different output neurons have different distance to \( v_0 \). Since any automorphism of directed graph would preserve in-degrees, out-degrees and distance, it has to map each output neuron to itself. \( \square \)

Also, the square-root operation has smoothing effect which forces the magnitudes of derivatives to be more concentrated. Thus, values at each output neuron would contribute more similarly to the overall resulting discrepancy. It reduces the risk of over-emphasizing features on certain dimensions and ignoring the rest ones.

![Figure 2: A representation of the SRVT block.](image)

Algorithm 1: Our proposed framework

**Input:** Real data distribution \( \mathbb{P}_r \) and a prior distribution \( p(z) \)

**Output:** Generator and discriminator parameters \( \theta, w \)

**while** \( \theta \) has not converged **do**

**for** \( i = 1 \) to \( k \) **do**

Sample real data \( x_i \) from \( \mathbb{P}_r \); sample random noise \( z_i \) from \( p(z) \);
\[
a_i \leftarrow \|D_w \circ G_\theta(z_i)\|; b_i \leftarrow \|D_w(x_i)\|;
\]
**end for**

\[
\mathcal{L}_D^{(p)} \leftarrow (\sum_{i=1}^{k} a_i^{1/p} - (\sum_{i=1}^{k} b_i^{1/p})^{1/p};
\]
\[
w \leftarrow \text{Adam}(\nabla_w \mathcal{L}_D^{(p)}, w);
\]
**for** \( i = 1 \) to \( k \) **do**

Sample random noise \( z_i \) from \( p(z) \);
\[
c_i \leftarrow \|D_w \circ G_\theta(z_i)\|;
\]
**end for**

\[
\mathcal{L}_G^{(p)} \leftarrow - (\sum_{i=1}^{k} c_i^{1/p})^{1/p};
\]
\[
\theta \leftarrow \text{Adam}(\nabla_\theta \mathcal{L}_G^{(p)}, \theta);
\]
**end while**

4 Experiments

In this section we provide experimental results supporting the proposed framework. We explore various setups to study characteristics of the proposed blocks. Since the proposed framework is closely related to WGAN, we also make a comparison between the two approaches in ablation study. Final evaluation results on benchmark datasets are presented afterwards.

4.1 Implementation Details

We applied the proposed framework in unconditional and conditional image generation tasks. For unconditional generation task, the generator and discriminator architectures were built following the ResNet architecture provided in (Miyato et al. 2018). The SRVT block and \( p \)-block were added to the last dense layer in the discriminator consecutively. Spectral normalization was utilized to ensure Lipschitz condition. Adam optimizer was used with learning rate \( 1e-4 \), \( \beta_1 = 0 \) and \( \beta_2 = 0.9 \). The length of input noise vector \( z \) was set to 128, and batch size was fixed to 64 in all experiments. Dimension of output from the last dense layer in discriminator was set to \( n = 1024 \) except for ablation study. For conditional generation task, we adopted the network architectures in BigGAN (Brock, Donahue, and Simonry 2019) and used their default parameter settings except for necessary changes for the proposed framework. All training tasks were conducted on Tesla V100 GPU.

4.2 Datasets and Evaluation Metrics

For unconditional image generation task, we implemented experiments on CIFAR-10 (Krizhevsky, Nair, and Hinton 2010), STL-10 (Coates, Ng, and Lee 2011) and LSUN bedroom (Yu et al. 2015) datasets. We used 60K images including 50K training images and 10K test images in CIFAR-10, 100K unlabelled images in STL-10, and 3M images
in LSUN bedroom, respectively. For conditional generation task we used CIFAR-10 and CIFAR-100 datasets. For each dataset, we center-cropped and resized the images, where images in STL-10 were resized to $48 \times 48$ and images in LSUN bedroom were resized to $64 \times 64$. Results were evaluated with Frechet Inception Distance (FID) (Heusel et al. 2017), Kernel Inception Distance (KID) (Binkowski et al. 2018a) and Precision and Recall (PR) (Sajjadi et al. 2018). Lower FID and KID scores and higher PR indicate better performance. In ablation study we generated 10K images for fast evaluation. For final evaluation on three datasets we used 50K generated samples. Please refer to the supplementary material for more details.

4.3 Results

In the following sections we first present ablation experimental results on CIFAR-10 with analysis, and then report final evaluation scores on all datasets. We display randomly generated samples in supplementary material.

**Ablation Study:**

We first conducted experiments under different settings to explore the effects of $p$-centrality function and SRVT used in our framework. Since our approach is tightly related to WGAN, we also include results from WGAN-GP and WGAN-SN for comparison. For each setup we trained 100K generator iterations on CIFAR-10 dataset, and reported average FID scores calculated from 5 runs in Fig 3. For this experiment we used 10K generated samples for fast evaluation. One can see without the use of SRVT (three green curves), settings with higher dimensional critic output resulted in better evaluation performances. The pattern is the same when comparing cases with SRVT (three blue curves). These observations are consistent with our Proposition 3.10. Furthermore, the results shows the asymmetric transformation boosts performances for different choices of $n$s, especially when $n = 1024$ (blue vs green). Our settings with high dimensional critic output outperform both WGAN-GP and WGAN-SN. In fact, sample qualities can further be improved with more training iterations, and we observe our training session can lead to a better convergence point. In Fig 4 we also present plots of precision and recall from these settings. Note that for WGAN-GP we obtained $(0.850, 0.943)$ recall and precision. Since the scores are far behind those from other methods, for display convenience we did not include WGAN-GP in the figure. As we see our default setting with the highest dimensional critic output $n = 1024$ and with the use of SRVT outperforms results from other settings.

We further present comparisons using KID under different settings in Fig 5. Results in Fig 5(a) are aligned with previous evaluations which shows the advantage of using higher dimensional critic output. Performance was further boosted with SRVT. Fig 5(b) shows KID evaluations under different choices of $p$s, where SRVT was used with fixed $n = 1024$. We observe using $p = 1$ only, or both $p = 1$ and 2 resulted in better performance compared with using $p = 2$ only. In practice one can customize $p$ for usage.

To keep a stable training session on CIFAR-10, WGAN-GP requires $n_{\text{critic}} = 5$, WGAN-SN requires $n_{\text{critic}} = 2$, and the proposed framework is able to train stably with $n_{\text{critic}} = 1$ with spectral normalization. For the same number of generator iterations, our approach can produce better sample qualities, with nearly the same amount of network parameters and training time. In experiments we observe generally a higher dimensional critic output $n$ requires less $n_{\text{critic}}$ to result in a stable training session. This is consistent with our theoretical results that a bigger $n$ leads to a “stronger” discriminator, and to result in a balanced game for the two networks, a smaller $n_{\text{critic}}$ can be used to ensure the training session going forward stably.

We further conducted experiments to validate the effect of SRVT with MMD-GAN objective (Li et al. 2017). For implementation we used the authors’ default hyper-parameter settings and network architectures. From Table 1 one can see SRVT significantly boosts performance for different $n$s. The best result was obtained with $n = 128$ (default setup in [28]). We also notice for MMD-GAN, higher $n$ (1024) did not improve performance (Binkowski et al. 2018b), while we have shown our framework can take advantage of higher
Table 1: Evaluation of KID(x10^3)(\downarrow) on the effect of SRVT with MMD-GAN objective and DCGAN architectures.

| Dimension of critic output | 16   | 128  | 1024 |
|----------------------------|------|------|------|
| w/o SRVT (Default)        | 17(1)| 16(1)| 20(1)|
| w/ SRVT                   | 14(1)| 13(1)| 16(1)|

Figure 5: KID evaluation under different settings. (a) Left: without SRVT; Right: default setting with SRVT. (b) Evaluation with SRVT under different ps with fixed n = 1024.

Table 2: FIDs(\downarrow) from unconditional generation experiments with ResNet architectures.

| Method  | CIFAR-10   | STL-10   | LSUN     |
|---------|------------|----------|----------|
| WGAN-GP | 19.0(0.8)  | 55.1     | 26.9(1.1)|
| SNGAN   | 14.1(0.6)  | 40.1(0.5)| 31.3(2.1)|
| MMD GAN-qq | -        | -        | 32.0     |
| CTGAN   | 17.6(0.7)  | -        | 19.5(1.2)|
| Sphere GAN | 17.1     | 31.4     | 16.9     |
| SWGAN   | 17.0(1.0)  | -        | 14.9(1.0)|
| CRGAN   | 14.6       | -        | -        |
| DGflow  | 9.6(0.1)   | -        | -        |
| Ours    | 8.5(0.3)   | 26.1(0.4)| 14.2(0.2)|

CIFAR-10: As we see in Table 2, the proposed method outperforms other comparable approaches under the same network architectures. It is able to generate high quality samples from different classes of objects using the simple architectures as presented in supplementary material.

STL-10: Data distribution of STL-10 is more diverse and complicated compared to CIFAR-10. Despite the difficulty of training on the dataset using simple network architectures, our method was able to obtain competitive results. We also conducted experiments with original 96 x 96 images on the dataset, and display randomly generated samples in supplementary material. While STL-10 is harder to train using ResNet architectures compared with the other two datasets, we observe our method manages to generate visually distinguishable samples from different classes for the diverse and complicated data distribution.

LSUN bedroom: LSUN bedroom dataset has relatively simpler data distribution, and most of the methods listed in Table 2 are able to generate high quality samples. Our competitive results also show the capability.

For conditional generation, we show evaluation results from the original BigGAN setting and the proposed objective in Table 3. The results indicate the proposed framework can also be applied in the more sophisticated training setting and obtain competitive performance.

Overall the proposed method is capable of obtaining competitive performances with different network architectures on the four datasets. In addition, compared to some classic approaches (Li et al. 2017; Wu et al. 2019; Ansari, Scarlett, and Soh 2020), the proposed framework does not require additional parameter tuning, which greatly facilitates implementation. In our experiments we did not see evidence of mode collapse.

Table 3: FIDs(\downarrow) from conditional generation experiments with BigGAN architectures.

| Objective | CIFAR-10  | CIFAR-100 |
|-----------|-----------|-----------|
| Hinge     | 9.7(0.1)  | 13.6(0.1) |
| Ours      | 8.9(0.1)  | 12.3(0.1) |

5 Conclusion and Discussion

In this paper we have proposed the maximal p-centrality discrepancy as a nice metric on the space of probability distributions, and used it in GAN objectives. The proposed metric fits well in the framework of WGAN especially when critic has multidimensional output. We have also proved that when n = 1, maximal p-centrality discrepancy is equivalent to 1-Wasserstein distance. We have further utilized an asymmetrical (square-root velocity) transformation added to discriminator to break the symmetric structure of its network output. The use of the nonparametric transformation takes advantage of multidimensional features and improves the generalization capability of critic network. In terms of implementation, the proposed framework does not require additional parameter tuning, which greatly facilitates implementation. In our experiments we did not see evidence of mode collapse.
References

Ansari, A. F.; Ang, M. L.; and Soh, H. 2021. Refining Deep Generative Models via Discriminator Gradient Flow. In International Conference on Learning Representations.

Ansari, A. F.; Scarlett, J.; and Soh, H. 2020. A Characteristic Function Approach to Deep Implicit Generative Modeling. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR).

Arjovsky, M.; and Bottou, L. 2017. Towards Principled Methods for Training Generative Adversarial Networks. 1

Arjovsky, M.; Chintala, S.; and Bottou, L. 2017. Wasserstein Generative Adversarial Networks. In Proceedings of the 34th International Conference on Machine Learning, volume 70, 214–223.

Arnaudon, M.; Barbaresco, F.; and Yang, L. 2013. Medians and means in Riemannian geometry: existence, uniqueness and computation. In Matrix Information Geometry, 169–197. Springer.

Badrinarayanan, V.; Mishra, B.; and Cipolla, R. 2015. Un-}

Brock, A.; Donahue, J.; and Simonyan, K. 2019. Large Scale GAN Training for High Fidelity Natural Image Synthesis. In International Conference on Learning Representations.

Chan, E. R.; Monteiro, M.; Kellnhofer, P.; Wu, J.; and Wetzstein, G. 2021. Pi-GAN: Periodic Implicit Generative Adversarial Networks for 3D-Aware Image Synthesis. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 5799–5809.

Coates, A.; Ng, A.; and Lee, H. 2011. An Analysis of Single-Layer Networks in Unsupervised Feature Learning. In Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics, volume 15, 215–223.

Deshpande, I.; Hu, Y.-T.; Sun, R.; Pyros, A.; Siddiqui, N.; Koyejo, S.; Zhao, Z.; Forsyth, D.; and Schwing, A. G. 2019. Max-Sliced Wasserstein Distance and Its Use for GANs. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR).

Deshpande, I.; Zhang, Z.; and Schwing, A. G. 2018. Generative Modeling Using the Sliced Wasserstein Distance. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR).

Dziugaite, G. K.; Roy, D. M.; and Ghahramani, Z. 2015. Training generative neural networks via maximum mean discrepancy optimization. In Proceedings of the Thirty-First Conference on Uncertainty in Artificial Intelligence, 258–267.

Glorot, X.; and Bengio, Y. 2010. Understanding the difficulty of training deep feedforward neural networks. In Proceedings of the thirteenth international conference on artificial intelligence and statistics, 249–256.

Goodfellow, I.; Pouget-Abadie, J.; Mirza, M.; Xu, B.; Wardefarley, D.; Ozair, S.; Courville, A.; and Bengio, Y. 2014. Generative adversarial nets. In Advances in neural information processing systems, 2672–2680.
Miyato, T.; Kataoka, T.; Koyama, M.; and Yoshida, Y. 2018. Spectral Normalization for Generative Adversarial Networks. In International Conference on Learning Representations. 2, 5, 7, 9
Mroueh, Y.; Li, C.; Sercu, T.; Raj, A.; and Cheng, Y. 2017. Sobolev GAN. CoRR, abs/1711.04894. 1
Mroueh, Y.; and Sercu, T. 2017. Fisher gan. In Advances in Neural Information Processing Systems, 2513–2523. 1
Mroueh, Y.; Sercu, T.; and Goel, V. 2017. McGan: Mean and Covariance Feature Matching GAN. In Proceedings of the 34th International Conference on Machine Learning, volume 70, 2527–2535. 1
Nauata, N.; Chang, K.-H.; Cheng, C.-Y.; Mori, G.; and Furukawa, Y. 2020. House-gan: Relational generative adversarial networks for graph-constrained house layout generation. In European Conference on Computer Vision, 162–177. Springer. 1
Niemeyer, M.; and Geiger, A. 2021. GIRAFFE: Representing Scenes As Compositional Generative Neural Feature Fields. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). 11453–11464. 1
Park, S. W.; and Kwon, J. 2019. Sphere Generative Adversarial Network Based on Geometric Moment Matching. In The IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 1, 7
Rabin, J.; Peyré, G.; Delon, J.; and Bernt, M. 2011. Wasserstein barycenter and its application to texture mixing. In International Conference on Scale Space and Variational Methods in Computer Vision, 435–446. Springer. 2
Sajjadi, M. S. M.; Bachem, O.; Lučić, M.; Bousquet, O.; and Gelly, S. 2018. Assessing Generative Models via Precision and Recall. In Advances in Neural Information Processing Systems (NeurIPS). 6
Srivastava, A.; Klassen, E.; Joshi, S. H.; and Jermyn, I. H. 2011. Shape Analysis of Elastic Curves in Euclidean Spaces. IEEE Transactions on Pattern Analysis and Machine Intelligence, 33(7): 1415–1428. 2
Srivastava, A.; and Klassen, E. P. 2016. Functional and shape data analysis, volume 1. Springer. 2
Wei, X.; Liu, Z.; Wang, L.; and Gong, B. 2018. Improving the Improved Training of Wasserstein GANs. In International Conference on Learning Representations. 2, 7
Wu, J.; Huang, Z.; Acharya, D.; Li, W.; Thoma, J.; Paudel, D. P.; and Gool, L. V. 2019. Sliced Wasserstein Generative Models. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). 1, 2, 7
Yu, F.; Zhang, Y.; Song, S.; Seff, A.; and Xiao, J. 2015. LSUN: Construction of a Large-scale Image Dataset using Deep Learning with Humans in the Loop. arXiv preprint arXiv:1506.03365. 5
Zhan, F.; Zhu, H.; and Lu, S. 2019. Spatial Fusion GAN for Image Synthesis. In The IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 1
Zhang, H.; Zhang, Z.; Odena, A.; and Lee, H. 2020. Consistency Regularization for Generative Adversarial Networks. In International Conference on Learning Representations. 7

6 Appendix

Network architectures for unconditional generation task: We employed ResNet architectures in (Miyato et al. 2018). Detailed generator and discriminator architectures are shown in Table 4 and 5.

Table 4: Generator architecture for 32 × 32 images.

| $z \in \mathbb{R}^{128} \sim \mathcal{N}(0, I)$ |
| dense, 4 × 4 × 256 |
| ResBlock up 256 |
| ResBlock up 256 |
| ResBlock up 256 |
| BN, ReLu, 3 × 3 conv, 3 Tanh |

Table 5: Discriminator architecture for 32 × 32 images.

| $x \in \mathbb{R}^{32 \times 32 \times 3}$ |
| ResBlock down 128 |
| ResBlock down 128 |
| ResBlock down 128 |
| ResBlock down 128 |
| LReLU |
| Global avg pooling |
| dense \rightarrow 1024 |
| SRVT block |
| p-block |

For STL-10 with image size $x \in \mathbb{R}^{48 \times 48 \times 3}$ and LSUN with image size $x \in \mathbb{R}^{64 \times 64 \times 3}$, we changed the number of units of the dense layer in generator to 6 × 6 × 256 and 8 × 8 × 256 respectively. All other setups were the same as above. Table 6 and 7 show network architectures implemented on 96 × 96 images.

Table 6: Generator architecture for 96 × 96 images.

| $z \in \mathbb{R}^{128} \sim \mathcal{N}(0, I)$ |
| dense, 6 × 6 × 512 |
| ResBlock up 512 |
| ResBlock up 256 |
| ResBlock up 128 |
| ResBlock up 64 |
| BN, ReLu, 3 × 3 conv, 3 Tanh |

Table 7: Discriminator architecture for 96 × 96 images.

| $x \in \mathbb{R}^{96 \times 96 \times 3}$ |
| ResBlock down 64 |
| ResBlock down 128 |
| ResBlock down 256 |
| ResBlock down 512 |
| ResBlock 512 |
| LReLU |
| Global avg pooling |
| dense \rightarrow 1024 |
| SRVT block |
| p-block |
Evaluation setup details: We used 50000 randomly generated samples comparing against real sets for testing. For each dataset we randomly sampled 50000 images and computed FID using 10 bootstrap resamplings. Features were extracted from the pool3 layer of a pre-trained Inception network. We display randomly generated examples from unconditional tasks in Fig 6.