This CVPR paper is the Open Access version, provided by the Computer Vision Foundation.
Except for this watermark, it is identical to the accepted version;
the final published version of the proceedings is available on IEEE Xplore.

Sphere Generative Adversarial Network Based on Geometric Moment Matching

Sung Woo Park and Junseok Kwon
School of Computer Science and Engineering, Chung-Ang University, Seoul, Korea
pswkiki@gmail.com jskwon@cau.ac.kr

Abstract

We propose sphere generative adversarial network (GAN), a novel integral probability metric (IPM)-based GAN. Sphere GAN uses the hypersphere to bound IPMs in the objective function. Thus, it can be trained stably. On the hypersphere, sphere GAN exploits the information of higher-order statistics of data using geometric moment matching, thereby providing more accurate results. In the paper, we mathematically prove the good properties of sphere GAN. In experiments, sphere GAN quantitatively and qualitatively surpasses recent state-of-the-art GANs for unsupervised image generation problems with the CIFAR-10, STL-10, and LSUN bedroom datasets. Source code is available at https://github.com/pswkiki/SphereGAN.

1. Introduction

Since the seminal work by Goodfellow *et al.* [8], generative adversarial networks (GANs) have attracted much research interest, and they have been used to achieve outstanding performance in a wide range of computer vision applications including in image generation [17, 26], super resolution [14], video prediction [19], style transfer [5, 12, 34], image inpainting [39], image editing [14], visual tracking [28], 3D reconstruction [1], segmentation [7], object detection [35], reinforcement learning [10], and medical imaging [40].

Conventional GANs try to minimize the distribution divergence between fake and real data [8]. For this purpose, the generator tries to produce desired samples that look like real data, and the discriminator tries to differentiate them from real data. Although GANs have been successfully applied to various tasks, it is very difficult to train them, in turn making it difficult to use them to solve more complex problems. For example, training dynamics frequently become unstable, and the generated samples easily collapse to a few modes.

While a lot of GANs and their applications have been proposed recently, in this paper, we focus on GANs based on integral probability metrics (IPMs) [2, 9, 24, 37] for overcoming the aforementioned problems. IPM-based GANs insert a gradient penalty term or soft consistent term into objective functions for achieving stable learning, resulting in a remarkable improvement in performance. However, these additional terms inevitably introduce additional hyper-parameters that need to be tuned, thereby incurring higher computation cost. In addition, many IPM-based GANs suffer from the unstable behavior of the sample-based constrain strategy, and WGAN uses only first-order statistics with a dual form of the 1-Wesserstein distance.

In this paper, we develop **sphere GAN**, a novel IPM-based GAN. Sphere GAN uses geometric moment matching and exploits the information of higher-order statistics of data, thus obtaining accurate results. Because moment matching is performed on the hypersphere, IPMs of sphere GAN can be bounded. We show that the geometric constraint induced by the hypersphere makes GAN training more stable. Sphere GAN affords these advantages without relying on the heuristics of conventional IPM-based GAN, namely, virtual sampling techniques and additional gradient penalty terms. Instead, sphere GAN utilizes Riemannian manifolds (*i.e.*, hypersphere) supported by the mathematical theory.

This paper makes three main contributions:

- We propose sphere GAN, a novel concept that afford several advantages over IPM-based GANs. To the best of our knowledge, our proposed sphere GAN is the first attempt to use Riemannian manifolds to define IPMs in GAN objective functions. In addition, it is the first IPM-based GAN that does not use the gradient penalty or virtual data sampling techniques.
- The good properties of sphere GAN are mathematically proven. In Section 4, we show that sphere GAN is closely related to IPMs and that minimizing the proposed distance amounts to minimizing the multiple Wesserstein distances of probability measures on the n-dimensional *hypersphere* $\mathbb{S}^n$.
- Sphere GAN outperforms recent state-of-the-art methods including IPM-based GAN variants for unsu-

pervised image generation problems with CIFAR-10, STL-10, and LSUN bedroom datasets. Sphere GAN significantly improves the accuracy by efficiently matching higher order moments in feature spaces.

2. Related Work

It is difficult to measure the distance between two non-overlapping probability distributions with low variances when we utilize discrepancy measures based on Kullback-Leibler (KL) divergence [2]. To overcome this problem, IPM-based GAN variants [23] have been recently proposed as alternatives for KL-divergence-based GANs. In IPMs, the distance between two probability distributions is measured by the largest discrepancy in expectation over a certain class of functions, making it crucial to select a proper class of functions in IPM-based GANs. In this section, we discuss the advantages and disadvantages of several IPM-based GAN variants.

Wesserstein distance: WGAN and its variants in [2, 9, 24, 37] use Wesserstein metrics to compare the probability measures of real images with those of fake images. In these methods, discriminators are modeled as a real-valued 1-Lipschitz function, which output a one-dimensional Euclidean space. To enforce the Lipschitz condition, WGAN clips the weights of discriminators such that they lie in a compact interval $[-\mathbf{c}, \mathbf{c}]$ [2]. However, weight clipping leads to unstable learning and produces sub-optimal results [9]. To solve this problem, WGAN with a gradient penalty (WGAN-GP) was proposed. However, the training time of WGAN-GP is almost two times that of other methods because it needs to calculate the gradient norm in every iteration. WGAN-CT [37] avoided this constraint by combining the gradient penalty term with the soft consistent term that penalizes violations of the 1-Lipschitz condition. WGAN-GP and WGAN-CT showed remarkable performance; however, both methods need additional penalty terms that can lead to sub-optimal solutions when penalty weights are wrongly selected.

WGAN-CT trains networks with good heuristics; by contrast sphere GAN does not sample virtual data points. Unlike WGAN-GP [9], WGAN-CT [37], and WGAN-LP [24], sphere GAN does not have an additional penalty term [20], making its training time much shorter. We experimentally demonstrate that sphere GAN achieves state-of-the-art results without gradient constraints.

Maximum mean discrepancy (MMD) distance: WGAN matches only first-order moments in discriminator networks [2]. By contrast, MMD GAN matches infinite-order moments defined on unit ball in Hilbert space [16]. MMD GAN affords several advantages through the use of higher-order statistics; however, it uses autoencoders to satisfy the injectivity of networks and performs weight clipping to bound the gradients for stable learning. Therefore, the objective functions in [16] considerably reduce the network capacity. The MMD distance cannot handle complex natural images well because the pixel space is high dimensional. In this case, the MMD distance produces low-quality samples and loses the diversity of representations.

Other IPMs: Squared MMD with a specific kernel is well known to be equivalent to the energy distance. The Cramér GAN used this energy distance to train GANs [3]. The critic function was parameterized by neural networks, and then, the energy distance was maximized [30, 41]. Like MMD GAN, Cramér GAN imposed the Lipschitz constraint on critic functions for achieving stable learning. By contrast, fisher GAN [22] and Sobolev GAN [21] defined function classes on the Lebesgue ball and the Sobolev space, respectively, to avoid the Lipschitz constraint; however, they need to solve an augmented Lagrangian to impose theoretical constraints on discriminators.

Like MMD GAN and Fisher GAN [22], sphere GAN uses information of higher-order statistics in GAN objective functions. However, MMD GAN and Fisher GAN require expensive penalty terms to satisfy theoretical assumptions. By contrast, the objective function of sphere GAN in (4) is simple and straightforward but also robust because it is mathematically equivalent to using multiple Wesserstein distances defined on a hypersphere. Section 4 provides mathematical proofs of the fact that the objective function (4) is closely related to IPMs.

3. Sphere GAN

This section introduces the novel **sphere GAN** and shows that is has several advantages compared to state-of-the-art IPM-based GANs.

3.1. Objective Function

The objective function based on the Wasserstein metric directly matches the first moment in the one-dimensional feature space as follows.

$$\min_G \max_D E_{x\sim\mathcal{P}}[D(x)] - E_{z\sim\mathcal{N}}[D(G(z))], \qquad (1)$$

where G and D denote the generator and discriminator, respectively, and $\mathcal{P}$ and $\mathcal{N}$ represent real data and latent code distributions, respectively. In (1), the discriminator D maps data x to a real number $\mathbb{R}$:

$$D : x \in \mathcal{X} \to \mathbb{R}, \qquad (2)$$

where D should satisfy the 1-Lipschitz condition $D \in \mathbf{Lip}_1$, and $\mathcal{X} \subset \mathbb{R}^n$ is the n-dimensional Euclidean image space. As in conventional IPM-based GANs, the objective function of our sphere GAN is based on (1). Unlike in existing GANs that directly match the first moment

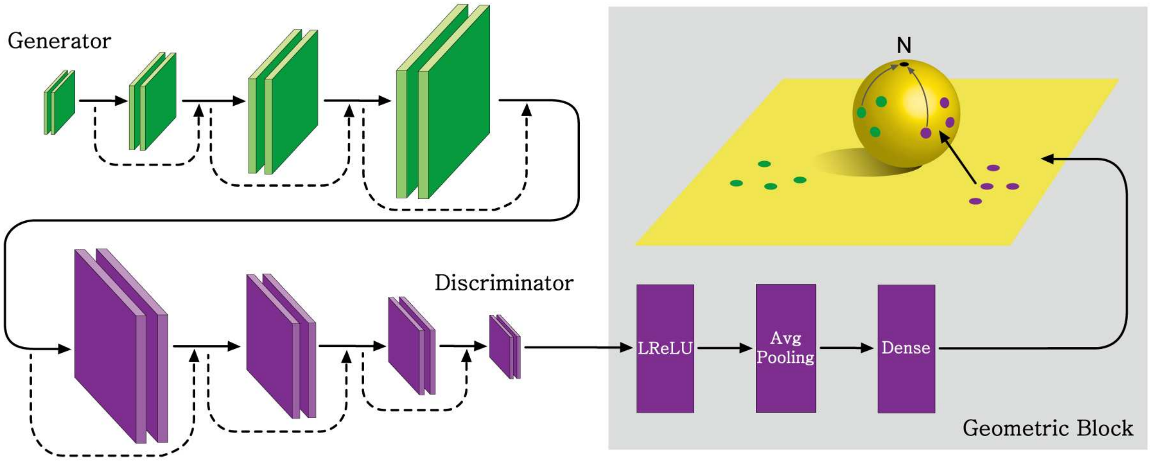

Figure 1. **Pipeline of sphere GAN.** Fake data is generated from noise inputs by a generator. Then, real and fake data are fed to a discriminator, which maps the output to an n-dimensional Euclidean feature space (*i.e.*, yellow plane). The green and purple circles on the plane denote feature points of fake and real samples, respectively. By geometric transformation, these feature points are re-mapped into the n-dimensional hypersphere (*i.e.*, yellow sphere). By using theses mapped points, sphere GAN calculates geometric moments centered at the north pole of the hypersphere. The discriminator of sphere GAN tries to maximize the moment differences of probability measures between real and fake samples, while the generator tries to interfere with the discriminator by minimizing the moment differences. By using the geometric moments defined on the hypersphere, the generator and discriminator enhance their performance through a two-player minmax game.

of one-dimensional feature spaces, sphere GAN matches higher-order and multiple moments defined on the hypersphere. For this purpose, the discriminator outputs an n-dimensional hypersphere $\mathbb{S}^n$:

$$D : x \in \mathcal{X} \rightarrow \mathbb{S}^n. \tag{3}$$

Then, the objective function of sphere GAN is defined as

$$\min_G \max_D \sum_r E_x[d_s^r(\mathbf{N}, D(x)] - \sum_r E_z[d_s^r(\mathbf{N}, D(G(z)))], \tag{4}$$

for $r = 1, \cdots, R$, where the function d_s^r in (8) measures the r-th moment distance between each sample and the north pole of the hypersphere, $\mathbf{N}$. Note that the subscript s indicates that d_s^r is defined on $\mathbb{S}^n$. Fig.1 shows the pipeline of sphere GAN.

With the new objective function in (4), sphere GAN affords advantages. First, by defining IPMs on the hypersphere, it can alleviate several constraints that should be imposed on the discriminator. As mentioned above, conventional discriminators based on the Wesserstein distance require Lipschitz constraints, which forces the discriminators to be a member of 1-Lipschitz functions. However, constraints with incorrect weight parameters λ considerably reduce the network capacity and overly reflect sampled points. For example, WGAN-GP, WGAN-CT, and WGAN-LP in [9, 24, 37] require additional constraint terms in the objective function for updating discriminators:

$$\mathcal{L}_{disc} = E_z[D(G^*(z))] - E_x[D(x)] + \lambda \mathbf{C}(x), \tag{5}$$

Table 1. **Gradient penalty terms** used in conventional GANs based on the Wesserstein distance. GP, CT, and LP denote gradient penalty, soft consistency, and Lipschitz penalty terms, respectively. $\hat{x}$ denotes the feature points that are uniformly sampled from straight lines from real to fake data points. x', x'' denote virtual data points which are perturbed by dropout units.

	Additional constraint term
GP	$E_{\hat{x}}\left[(\lVert\nabla_{\hat{x}} D(\hat{x})\rVert_2 - 1)^2\right]$
CT	GP + $E_{x',x''}\left[\max\left(0, d(D(x'), D(x''))\right) - Const\right]$
LP	$E_{\hat{x}}\left[\max\left(0, \lVert\nabla_{\hat{x}} D(\hat{x})\rVert_2 - 1\right)^2\right]$

where G^* denotes the fixed generator and $\mathbf{C}$ denotes additional constraint terms that are defined in Table 1. In (5), the gradient norm should be calculated at every iteration; this increases the computational complexity. Unlike in conventional approaches, sphere GAN does not need any additional constraints that forces discriminators to lie in a desired function space. By using *geometric transformation*, sphere GAN ensures that distance functions lie in a desired function space. Then, our new objective function for updating the weights of the discriminator is

$$\mathcal{L}_{disc} = \sum_r E_z[d_s^r(\mathbf{N}, D(G^*(z)))] - \sum_r E_x[d_s^r(\mathbf{N}, D(x)], \tag{6}$$

where there are no additional constraint terms. Algorithm 1 show the pseudo-code of sphere GAN.

Algorithm 1 Sphere GAN

```
Input: Real data distribution $\mathcal{P}$.
Output: Discriminator and generator parameters: $w, \theta$
while $\theta$ has not converged do
    for $r = 1$ to $R$ do
        Sample real data $x$ from $\mathcal{P}$.
        Sample random noise $z$ from $\mathcal{N}(0, I)$.
        $\mathcal{L}_{disc}^{(r)} \leftarrow d_s^r(\mathbf{N}, D_w(G_\theta(z))) - d_s^r(\mathbf{N}, D_w(x))$
    end for
    for $r = 1$ to $R$ do
        Sample real data $x$ from $\mathcal{P}$.
        Sample random noise $z$ from $\mathcal{N}(0, I)$.
        $\mathcal{L}_{gen}^{(r)} \leftarrow -d_s^r(\mathbf{N}, D_w(G_\theta(z)))$
    end for
    $w \leftarrow \text{Adam}(\nabla_w \sum_{r=1}^{R} \mathcal{L}_{disc}^{(r)}, w)$
    $\theta \leftarrow \text{Adam}(\nabla_\theta \sum_{r=1}^{R} \mathcal{L}_{gen}^{(r)}, \theta)$
end while
```

3.2. Hypersphere

As in (4), sphere GAN matches multiple moments over the feature space defined on the hypersphere $\mathbb{S}^n$. Sphere GAN uses the hypersphere instead of arbitrary Riemannian manifolds $\mathcal{M}$ because doing so affords the following three advantages.

1. The distance function d_s^r of the hypersphere is bounded and becomes very easy to implement.

2. The gradient norm behaves well with this distance function, which is crucial for stable learning.

3. The Riemannian structure of the hypersphere is suitable for defining GAN objectives.

Conventional GANs typically consider the Euclidean space $\mathbb{R}^n$ with the Euclidean distance. These GANs can be extended by modeling arbitrary Riemannian manifolds. These manifolds are not compact and the distance function is not bounded, which may cause gradient explosion and unstable learning. To solve this problem, sphere GAN uses a geometric-aware transformation function, which transforms the Euclidean space $\mathbb{R}^n$ to the hypersphere $\mathbb{S}^n$. Note that this function is implemented by the last dense layer of the discriminator. Our transformation function is designed by a *diffeomorphism*[1] from $\mathbb{R}^n$ to $\mathbb{S}^n$. Thus, the transformation function is differentiable and can preserve dimensionality at every point of the feature space. The next section introduces stereographic projection as a geometric transformation function.

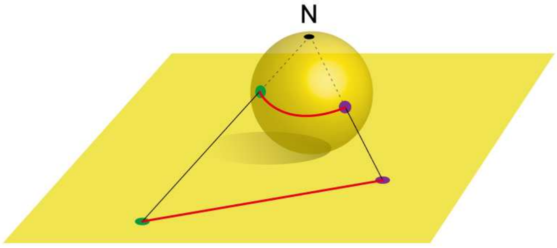

Figure 2. **Inverse of stereographic projection on Euclidean plane $\mathbf{\Pi}^{-1} : \mathbb{R}^2 \rightarrow \mathbb{S}^2/\{\mathbf{N}\}$.** Each red line denote the geodesics on $\mathbb{R}^2$ and $\mathbb{S}^2$.

3.3. Geometric-aware transformation function

The inverse of the stereographic projection is a diffeomorphism from the Euclidean space $\mathbb{R}^n$ to the hypersphere $\mathbb{S}^n$. Intuitively, the inverse of the stereographic projection can be considered a way of projecting the hyperplane onto the hypersphere. Let $p = (p_1, \ldots, p_n)$ be a coordinate system of $\mathbb{R}^n$ and $\mathbf{N} = (0, \ldots, 1)$ be a north pole of the hypersphere. Then, the inverse of the stereographic projection $\Pi^{-1} : \mathbb{R}^n \rightarrow \mathbb{S}^n/\{\mathbf{N}\}$ is defined as follows:

$$\mathbf{\Pi}^{-1}(p) = \left(\frac{2p}{||p||^2 + 1}, \frac{||p||^2 - 1}{||p||^2 + 1} \right). \tag{7}$$

After projecting two points $p, q \in \mathbb{R}^n$ through the inverse of the stereographic projection, we measure the distance between two points, in terms of hypersphere metrics:

$$\begin{aligned} & d_s(\mathbf{\Pi}^{-1}(p), \mathbf{\Pi}^{-1}(q)) \\ & = \arccos \left(\frac{||p||^2||q||^2 - ||p||^2 - ||q||^2 + 4p \cdot q + 1}{(||p||^2 + 1)\,(||q||^2 + 1)} \right), \end{aligned} \tag{8}$$

where d_s is the distance function defined on $\mathbb{S}^n$.

Geometrically, d_s can be considered a geodesic distance. As shown in Fig.2, the geodesic distance between two points on the hypersphere is much shorter than the Euclidean distance and is bounded on the hypersphere (*i.e.*, yellow sphere), thus implementing geometric transformation is equivalent to impose global constraint to hyperplane. As a result, it enables stable training when using sphere GAN with the objective function in (4).

Lemma 1. *The distance function in* (8) *is differentiable and is bounded.*

The distance function in (8) satisfies non-negativity, symmetry, and triangle inequality and is differentiable. The distance between any two points is bounded, because the hypersphere is a compact manifold. For example, the Euclidean distance between two points $\mathbf{0} = (0, \ldots, 0)$ and

[1] The diffeomorphism is a bijective and differentiable function, which preserves the dimensionality of the tangent space of the domain and image smooth manifolds.

$q = (t, \ldots, t)$ diverges: $\sqrt{nt^2} \to \infty$ as $t \to \infty$. By contrast, the distance defined on the hypersphere in (8) converges: $d(\mathbf{\Pi}^{-1}(\mathbf{0}), \mathbf{\Pi}^{-1}(q)) = \arccos\left(\frac{-nt^2+1}{nt^2+1}\right) \to \pi$ as $t \to \infty$. The geometric-aware transformation function of sphere GAN makes the distribution divergence of the discriminator outputs bounded, thereby enforcing stable training dynamics. In addition, the function preserves the dimensionality of the feature spaces and maintains differentiability.

4. Analysis of Sphere GAN

This section presents a mathematical analysis of sphere GAN.

4.1. Link to IPMs

We first prove that minimizing the objective function in (4) amounts to minimizing IPMs. For this purpose, we define geometric central moments on the Riemannian manifold. Let $\mathcal{M}$ be the compact, connected, and geodesically complete Riemannian manifold with Borel σ-algebra, Σ. Both $p \sim \mathbb{P}$ and $q \sim \mathbb{Q}$ are probability measures defined on the measurable space $(\mathcal{M}, \Sigma)$. Then, the IPM is defined as follows:

Definition 1. *The IPM is a distance measure between two probability measures $\mathbb{P}$ and $\mathbb{Q}$:*

$$\gamma(\mathbb{P}, \mathbb{Q}) = \sup_{f \in \mathcal{F}} \left| \int_{\mathcal{M}} f d\mathbb{P} - \int_{\mathcal{M}} f d\mathbb{Q} \right|, \tag{9}$$

where $\mathcal{F}$ is a class of real-valued bounded measurable functions on $\mathcal{M}$.

We can define the geometric moments on $\mathcal{M}$:

Definition 2. *The r-th central moment of $\mathbb{P}$ on $(\mathcal{M}, \Sigma)$ for given a point p_0 is*

$$\mathbf{m}_{\mathbb{P}}^{\mathbf{r}} = \int_{\mathcal{M}} d^r(p_0, p) \mathbf{d}\mathbb{P}(p), \tag{10}$$

where $1 \leq r < \infty$ and $\mathbf{m}_{\mathbb{P}}^r < \infty$. d^r is the Riemannian distance function on $\mathcal{M}$.

In sphere GAN, we define a new IPM between $\mathbb{P}$ and $\mathbb{Q}$:

Definition 3. *The IPM based on the moment difference is*

$$\gamma_{\mathcal{M}}(\mathbb{P}, \mathbb{Q}) = \sup_{d \in \mathcal{C}_{p_0}(\mathcal{M})} \sum_r \left| \mathbf{m}_{\mathbb{P}}^{\mathbf{r}} - \mathbf{m}_{\mathbb{Q}}^{\mathbf{r}} \right|, \tag{11}$$

where $\mathcal{C}_{p_0}(\mathcal{M})$ is a class of bounded distance functions from a given point p_0 to another point on $\mathcal{M}$.

When we compare Definition 1 with Definition 3, we note relations between conventional IPMs and the IPM of sphere GAN. While $\mathbf{m}_{\mathbb{P}}^r$ in (11) corresponds to $\mathbb{E}_x[d_s^r(p_0, D(x))]$ in (4), $\mathcal{M}$ can be replaced by $\mathbb{S}^n$[2] and x_0 can be set to north pole $\mathbf{N}$. Then, we obtain the same equation as (4), which implies that minimizing the objective function in (4) amounts to minimizing IPMs in (11).

However, there are several differences between conventional IPMs and the IPMs of sphere GAN. The function space of our IPM is the set of bounded distance functions on $\mathcal{M}$ centered at p_0, $\mathcal{C}_{p_0}(\mathcal{M})$. Thus, sphere GAN parameterizes *distance functions*:

$$\mathbb{E}_x[d_s^r(p_0, D(x))] \simeq \frac{1}{N} \sum_{i=1}^{N} d_s^r(p_0, D(x_i)), \tag{12}$$

where $\{x_i\}$ is the set of images. By contrast, the function space of the IPM of WGAN is the set of 1-Lipschitz discriminators. Thus, it parameterizes *discriminators*.

$$\mathbb{E}_x[D(x)] \simeq \sum_{i=1}^{N} D(x_i), \tag{13}$$

where $D \in \mathbf{Lip}_1$.

4.2. Link to Wesserstein distance

$\gamma_{\mathbb{S}^n}$ is the IPM of sphere GAN defined in (11), where $\mathcal{M} = \mathbb{S}^n$. The generator of sphere GAN aims to reduce $\gamma_{\mathbb{S}}$, which is equivalent to matching higher-order central moments between two probability measures $\mathbb{P}$ and $\mathbb{Q}$ defined on $\mathbb{S}^n$:

Proposition 1. *As $\mathbb{P}$ weakly converges to $\mathbb{Q}$,*

- $\gamma_{\mathbb{S}^n} \to 0$
- $\mathbf{m}_{\mathbb{P}}^{\mathbf{r}} \to \mathbf{m}_{\mathbb{Q}}^{\mathbf{r}}$ *for all* $\mathbf{r}$

Let $W_{\mathbb{S}^n}^r$ be the r-Wesserstein distance of probability measures defined on $\mathbb{S}^n$. Then, minimizing $\gamma_{\mathbb{S}^n}$ is equivalent to minimizing the summation of r-Wesserstein distances over all r.

Proposition 2. *As $\gamma_{\mathbb{S}^n}$ converges to* 0,

$$\sum_r W_{\mathbb{S}^n}^r(\mathbb{P}, \mathbb{Q}) \to 0. \tag{14}$$

The result of Proposition 2 is not surprising because weak convergence is strongly related to the Wesserstein distance [33]. In conventional GANs based on the Wesserstein distance [2, 9, 24, 37], objective functions are designed as a dual form by the Kantorovich-Rubinstein duality theorem. In the dual form, only the 1-Wesserstein distance can be implemented for achieving efficient learning of GANs. Contrary to conventional GANs, sphere GAN can use more general r-Wesserstein distances, and thus, the function space is much broader.

[2] Note that the hypersphere satisfies all assumptions mentioned earlier in this section.

4.3. Gradient Analysis

By using $\gamma_{\mathbb{S}^n}$ over other IPMs, sphere GAN can compute the gradients of loss functions by choosing different moments of $\gamma_{\mathbb{S}^n}$. The selection of different moments leads to different learning behaviors as the gradients differ. We found that any moment enables stable learning using sphere GAN.

Lemma 2. $\mathbb{E}_{x\sim\mathcal{P}}\left[||\nabla_x d_s^r(\mathbf{N}, D(x))||_2\right] < \infty$ *for all* r.

Lemma 2 tells us that using the hypersphere is a reasonable choice for stably learning GANs, where the norm of gradient is bounded during the training. But our sphere GAN can have large gradients because no penalty is imposed on the discriminator. Thus, it has a potential risk of gradient explosion. However, in experiments, we observed that the average magnitude of the norm of gradients at each iteration is affordable when using the Adam optimizer.

5. Experiments

5.1. Implementation Details

Hyper-parameters: The network was trained with batch size of 64. In all experiments, we used the Xavier initialization and Adam optimizer for both the generator and the discriminator. We fixed the hyper-parameters of the Adam optimizer for the generator and discriminator to $\alpha = 1\text{E}-4, \beta_1 = 0, \beta_2 = 0.9$. In experiments using ConvNet, we set the moment modes to $\sum_1^5 d^r$. In other experiments, we set the dimension of the hypersphere to $\mathbb{S}^{1024}$ and the moment modes to $\sum_1^3 d^r$. In conventional IPM-based GANs, the discriminator was updated multiple times and the generator, one time, per iteration. Contrary to these GANs, in sphere GAN, both networks were updated one time per iteration[3].

Geometric Block: We added the geometric block to the last convolutional layer of the discriminator for geometric-aware transformation. The discriminator (**D**) and geometric block (**GB**) were designed as follows:

$$\begin{aligned}\mathbf{D}&: \mathcal{X} \to \text{ConvBlocks} \to \text{GB}\\ \mathbf{GB}&: \text{ReLU} \to \text{AverageMeanPooling}\\ &\to \text{DenseLayer}(\mathbb{R}^n) \to \text{ISGP}(\mathbb{S}^n \subset \mathbb{R}^{n+1}),\end{aligned}$$

where $\mathcal{X} \subset \mathbb{R}^n$ is an input and ISGP denotes the inverse of stereo-graphic projection. The pseudo code for ISGP and detailed network structures are provided in the supplementary materials.

Baseline Network: We conducted unsupervised image generation tasks using two baseline networks: ConvNet and ResNet. For ConvNet, we followed the network architecture proposed in [20] to build both the generator and the discriminator. It consists of transposed convolutional blocks in the generator and convolutional blocks in the discriminator, in which each blocks consists of two convolutional layers. For ResNet, we followed the network architectures proposed in [9]. In both discriminator networks, we used layer normalization [15] for the normalization unit suggested in [22], and we attached the geometric block **GB** to the last convolutional block for geometric transformation. Details of the network architectures are provided in the supplementary materials.

Environments: All experiments were conducted using a single GTX Titan GPU. Sphere GAN was implemented using `Keras-2.2.4` with `Tensorflow-1.11.0` backend.

5.2. Dataset and Evaluation Metrics

Dataset: We conducted experiments on CIFAR-10 [13], STL-10 [6], and LSUN [38] datasets. CIFAR-10 and STL-10 contains around 50K and 100K natural images of size 32 × 32 and 96 × 96 with 10 different classes, respectively. For STL-10, we downsized original images to a size of 48 × 48. For LSUN, we used around 3M bedroom images that were resized to 64 × 64.

Evaluation Metrics: To quantitatively evaluate the networks, we used two metrics for image generation tasks: inception score (IS) [27] and Fréchet inception distance (FID) [11]. By using these metrics, we compared sphere GAN against other IPM-based GANs with various datasets. In all experiments, we generated 50K images to evaluate GANs in terms of IS and FID. For implementation, we used open source code provided by the authors[4].

IS is strongly correlated to human judgment and inception. The generated images were applied to an inception convolutional network [29] to obtain the conditional distribution $p(y|x)$, and IS was calculated as follows: $\exp\left(\mathbb{E}\left[D_{KL}[p(y|\mathbf{x})||p(y)]\right]\right)$ where $p(y)$ is approximated by $\frac{1}{N}\sum_{n=1}^{N} p(y|\mathbf{x}_n)$. On the other hand, FID overcomes the problems of IS by estimating the 2-Wesserstein distance of Gaussian distributions induced by the outputs of hidden activation (*pool*3 of inception model). FID is consistent with increasing disturbances and human judgment. FID between two image distributions $\mathbb{P}_1, \mathbb{P}_2$ is defined as follows:

$$\mathbf{FID}(\mathbb{P}_1, \mathbb{P}_2) = ||\mathbf{m_1} - \mathbf{m_2}||_2^2 + Tr(\mathbf{C_1} + \mathbf{C_2} - 2(\mathbf{C_1}\mathbf{C_2})^{\frac{1}{2}}), \tag{15}$$

where $\mathbf{m}_i$ and $\mathbf{C}_i$ are the Gaussian mean and covariance matrix obtained from $\mathbb{P}_i$, respectively.

5.3. Ablation Study

This section aims to answer the following three questions:

[3] One study has investigated the dynamics of learning GANs [11]. However, it is difficult to perform direct comparisons and analyses.

[4] IS: https://github.com/openai/improved-gan, FID: https://github.com/bioinf-jku/TTUR.

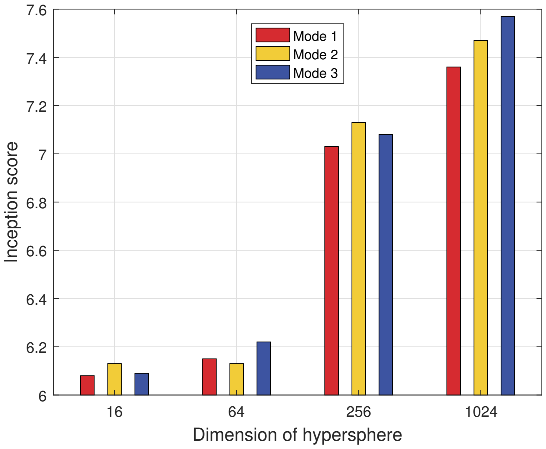

Figure 3. **Inception scores (IS) on CIFAR-10** with ConvNet according to different moment matching modes and different dimensions of hypersphere. Red, yellow, and blue bars denote moment modes: $\sum_1 d^r$, $\sum_1^3 d^r$, and $\sum_1^5 d^r$, respectively. The horizontal axis denotes the dimensions of hypersphere $\mathbb{S}^n : n = 16, 64, 256, 1024$.

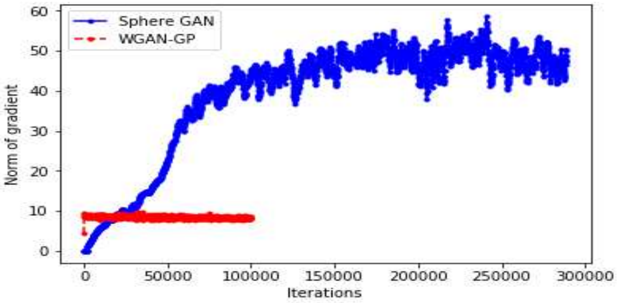

Figure 4. **Norm of gradient for Sphere GAN and WGAN-GP discriminator networks.**

Q1: Does training GANs with higher moments improve the quality of samples?

Q2: Does training GANs with higher dimensions of the hypersphere improve performance?

Q3: Does the norm of gradients behave well during training?

Answer to Q1: We conducted unsupervised image generation tasks with different moments to show that higher moments help to improve performance. In this experiment, various summation modes were used for the GAN objective. As shown in Fig.3, matching higher moments in the feature space considerably improves performance. We observed that higher than fifth-order moments deteriorate the performance in the CIFAR-10 dataset, because higher moments induces high magnitude of gradients, and this may cause unstable learning. However, in experiments, $\sum_{i=1}^3 d^r$ was suitable for large networks in general. Conventional GANs based on the Wesserstein distance did not improve their accuracy as higher-order moments were used.

Table 2. **Unsupervised image generation results on CIFAR-10**. IS : higher is better, FID : lower is better. For networks with $\star$, we used the results reported in [20].

Method	IS	FID
CIFAR-10 (real)	11.24 ± .12	7.8
MMD GAN [16]	6.17 ± .07	-
Weight clipping*	6.41 ± .11	42.6
WGAN-GP*	6.68 ± .06	40.2
Spectral Norm-WD*	7.20 ± .08	32.0
Sphere GAN-Conv	7.57 ± .05	-
WGAN-GP-ResNet [9]	7.86 ± .07	-
χ^2 GAN [31]	7.88 ± .10	-
Fisher GAN [22]	7.90 ± .05	-
Coulomb GAN [32]	-	27.3
Spectral Norm-WD*	7.96 ± .06	22.5
WGAN-LP [24]	8.02 ± .08	-
WGAN-CT [37]	8.12 ± .12	-
Spectral Norm [20]	8.22 ± .05	21.7
Sphere GAN-ResNet	**8.39 ± .08**	**17.1**

Answer to Q2: We observed that the dimensions of the hypersphere should be large enough to ensure that the information contained in the feature space is meaningful in using geometric moments. In other methods where the feature space is one dimension (*e.g.*, Wesserstein distance), the dimension of the feature space was not enough to deliver the information of higher-order statistics. As demonstrated in Fig.3, higher dimensions of the hypersphere significantly improved the accuracy of sphere GAN.

Answers to Q3: We evaluated the norm of gradients at each iteration to show that GANs can be trained stably with the proposed metric. As shown in Fig.4, the norm of gradients started to converge after 100K iterations, while WGAN-GP easily attained the convergence. In sphere GAN, the norm of gradients was smoothly bounded with the proposed metric.

5.4. Quantitative and Qualitative Results

CIFAR-10 : Table 2 summarizes quantitative results. Sphere GAN-ResNet achieved state-of-the-art scores for both IS and FID with a large margin. Sphere GAN-Conv also outperformed WGAN-GP and MMD GAN.

STL-10 : In experiments with STL-10, we used approximately one-half the number of network parameters compared to the original network used in [20]. Despite the small number of network parameters, sphere GAN-ResNet significantly outperformed SN-GAN and other IPM-based GANs, as shown in Table 3.

LSUN Bedroom : In this experiment, we reported FID only because IS was not meaningful, as noted in [4]. The results in Table 4 indicate that sphere GAN-ResNet outperformed state-of-the-art GANs.

Table 3. **Unsupervised image generation results on STL-10**. For networks with $\star$, we used the results reported in [20].

Method	IS	FID
STL-10 (real)	26.08 ± .26	7.9
Weight clipping*	7.57 ± .10	64.2
WGAN-GP*	8.42 ± .13	55.1
Sphere GAN-Conv	8.43 ± .09	44.1
Warde-Farley [36]	8.51 ± .13	-
Spectral Norm*	9.10 ± .04	40.1
Sphere GAN-ResNet	**9.55 ± .11**	**31.4**

Table 4. **Unsupervised image generation results on LSUN Bedroom**. For networks with $\star$, we used the results reported in [4].

Method	FID
LSUN Bedroom (real)	2.36
Cramér GAN*	54.2
WGAN-GP*	41.4
MMD-GAN-rq*	32.0
Sphere GAN	**16.9**

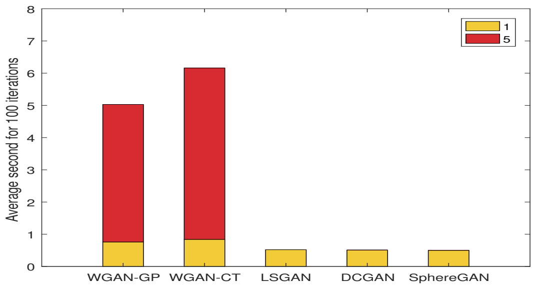

Figure 5. **Averaged computation time over 100 iterations for different GAN variants**. The yellow and red bars denote the averaged computation time when the updating ratio of the generator and discriminator is $1 : 1$ and $1 : 5$, respectively.

Training Time: In Fig.5, we calculated the averaged training time over 100 iterations for different methods. WGAN-CT and WGAN-GP were clearly much slower than other methods (around 40% slower than DCGAN) because they calculate the norm of gradients $||\nabla_{\hat{x}} D(\hat{x})||_2$ at every iteration. The training time of sphere GAN is much shorter than that of other IPM-based GANs and almost the same as that of vanilla DCGAN [25] and LSGAN [18].

We qualitatively evaluated sphere GAN using three datasets. Figs.6 and 7 show the qualitative results of sphere GAN for the LSUN-bedroom and STL-10 datasets, respectively. The qualitative results indicate that sphere GAN was trained stably and hardly suffered from mode collapse problems. Most generated images are photo-realistic.

6. Conclusion

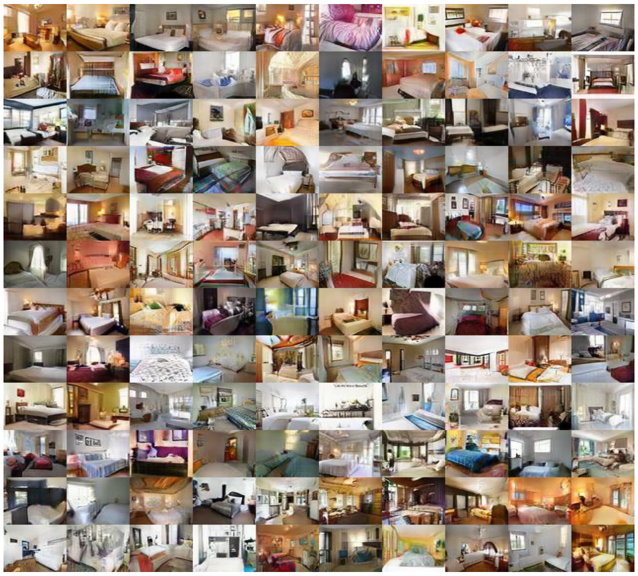

Figure 6. **Qualitative results** of sphere GAN for LSUN-bedroom dataset

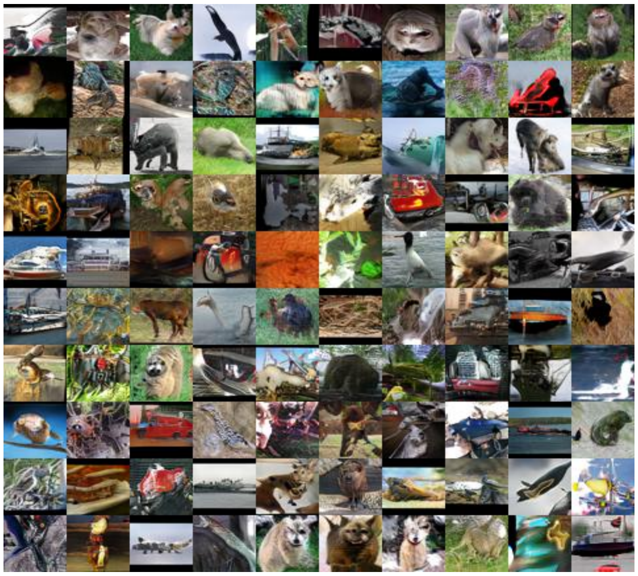

Figure 7. **Qualitative results** of sphere GAN for STL-10 dataset

This paper proposes sphere GAN, a novel IPM-based GAN. Sphere GAN defines IPMs on the hypersphere (*i.e.*, a type of Riemannian manifolds), and therefore, it can be trained stably using bounded IPMs. High-order moment matching enables sphere GAN to exploit useful information about data and to provide accurate results. Experimental results demonstrate that sphere GAN shows state-of-the-art performance compared to IPM-based GANs for the LSUN, STL-10, and CIFAR-10 datasets.

Acknowledgements

This work was supported by Institute for Information & communications Technology Planning & evaluation(IITP) grant funded by the Korea government(MSIT) (No.2017-0-01780).

References

[1] P. Achlioptas, O. Diamanti, I. Mitliagkas, and L. Guibas. Learning representations and generative models for 3d point clouds. In *ICLR*, 2018. 1

[2] M. Arjovsky, S. Chintala, and L. Bottou. Wasserstein generative adversarial networks. In *ICML*, 2017. 1, 2, 5

[3] M. G. Bellemare, I. Danihelka, W. Dabney, S. Mohamed, B. Lakshminarayanan, S. Hoyer, and R. Munos. The cramer distance as a solution to biased wasserstein gradients. *arXiv preprint arXiv:1705.10743*, 2017. 2

[4] M. Bikowski, D. J. Sutherland, M. Arbel, and A. Gretton. Demystifying MMD GANs. In *ICLR*, 2018. 7, 8

[5] Y. Chen, Y.-K. Lai, and Y.-J. Liu. Cartoongan: Generative adversarial networks for photo cartoonization. In *CVPR*, 2018. 1

[6] A. Coates, A. Ng, and H. Lee. An analysis of single-layer networks in unsupervised feature learning. In *AISTATS*, 2011. 6

[7] K. Ehsani, R. Mottaghi, and A. Farhadi. Segan: Segmenting and generating the invisible. In *CVPR*, 2018. 1

[8] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative adversarial nets. In *NIPS*, 2014. 1

[9] I. Gulrajani, F. Ahmed, M. Arjovsky, V. Dumoulin, and A. C. Courville. Improved training of wasserstein GANs. In *NIPS*, 2017. 1, 2, 3, 5, 6, 7

[10] P. Henderson, W.-D. Chang, P.-L. Bacon, D. Meger, J. Pineau, and D. Precup. Optiongan: Learning joint reward-policy options using generative adversarial inverse reinforcement learning. In *AAAI*, 2018. 1

[11] M. Heusel, H. Ramsauer, T. Unterthiner, B. Nessler, and S. Hochreiter. GANs trained by a two time-scale update rule converge to a local nash equilibrium. In *NIPS*, 2017. 6

[12] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros. Image-to-image translation with conditional adversarial nets. In *CVPR*, 2017. 1

[13] A. Krizhevsky. Learning multiple layers of features from tiny images. Technical report, Citeseer, 2009. 6

[14] C. Ledig, L. Theis, F. Huszar, J. Caballero, A. Cunningham, A. Acosta, A. P. Aitken, A. Tejani, J. Totz, Z. Wang, and W. Shi. Photo-realistic single image super-resolution using a generative adversarial network. In *CVPR*, 2017. 1

[15] J. Lei Ba, J. R. Kiros, and G. E. Hinton. Layer normalization. *arXiv preprint arXiv:1607.06450*, 2016. 6

[16] C.-L. Li, W.-C. Chang, Y. Cheng, Y. Yang, and B. Poczos. Mmd gan: Towards deeper understanding of moment matching network. In *NIPS*, 2017. 2, 7

[17] J. Lin, Y. Xia, T. Qin, Z. Chen, and T.-Y. Liu. Conditional image-to-image translation. In *CVPR*, 2018. 1

[18] X. Mao, Q. Li, H. Xie, R. Y. Lau, Z. Wang, and S. P. Smolley. Least squares generative adversarial networks. *arXiv preprint arXiv:1611.04076*, 2017. 8

[19] M. Mathieu, C. Couprie, and Y. LeCun. Deep multi-scale video prediction beyond mean square error. In *ICLR*, 2016. 1

[20] T. Miyato, T. Kataoka, M. Koyama, and Y. Yoshida. Spectral normalization for generative adversarial networks. In *ICLR*, 2018. 2, 6, 7, 8

[21] Y. Mroueh, C.-L. Li, T. Sercu, A. Raj, and Y. Cheng. Sobolev GAN. In *ICLR*, 2018. 2

[22] Y. Mroueh and T. Sercu. Fisher gan. In *NIPS*, 2017. 2, 6, 7

[23] A. Müller. Integral probability metrics and their generating classes of functions. *Advances in Applied Probability*, 29(2):429–443, 1997. 2

[24] H. Petzka, A. Fischer, and D. Lukovnikov. On the regularization of wasserstein GANs. In *ICLR*, 2018. 1, 2, 3, 5, 7

[25] A. Radford, L. Metz, and S. Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks. *arXiv preprint arXiv:1511.06434*, 2015. 8

[26] S. Reed, Z. Akata, L. X. Yan, B. Logeswaran, Schiele, and H. Lee. Generative adversarial text to image synthesis. In *ICML*, 2016. 1

[27] T. Salimans, I. Goodfellow, W. Zaremba, V. Cheung, A. Radford, and X. Chen. Improved techniques for training GANs. In *NIPS*, 2016. 6

[28] Y. Song, C. Ma, X. Wu, L. Gong, L. Bao, W. Zuo, C. Shen, R. Lau, and M.-H. Yang. Vital: Visual tracking via adversarial learning. In *CVPR*, 2018. 1

[29] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna. Rethinking the inception architecture for computer vision. In *CVPR*, 2016. 6

[30] G. J. Székely and M. L. Rizzo. Energy statistics: A class of statistics based on distances. *Journal of statistical planning and inference*, 143(8):1249–1272, 2013. 2

[31] C. Tao, L. Chen, R. Henao, J. Feng, and L. C. Duke. Chi-square generative adversarial network. In *ICML*, 2018. 7

[32] T. Unterthiner, B. Nessler, C. Seward, G. Klambauer, M. Heusel, H. Ramsauer, and S. Hochreiter. Coulomb GANs: Provably optimal nash equilibria via potential fields. In *ICLR*, 2018. 7

[33] C. Villani. *Optimal Transport: Old and New*. Springer Berlin Heidelberg, 2008. 5

[34] X. Wang and A. Gupta. Generative image modeling using style and structure adversarial networks. In *ECCV*, 2016. 1

[35] X. Wang, A. Shrivastava, and A. Gupta. A-fast-rcnn: Hard positive generation via adversary for object detection. In *CVPR*, 2017. 1

[36] D. Warde-Farley and Y. Bengio. Improving generative adversarial networks with denoising feature matching. In *ICLR*, 2017. 8

[37] X. Wei, Z. Liu, L. Wang, and B. Gong. Improving the improved training of wasserstein GANs. In *ICLR*, 2018. 1, 2, 3, 5, 7

[38] F. Yu, A. Seff, Y. Zhang, S. Song, T. Funkhouser, and J. Xiao. Lsun: Construction of a large-scale image dataset using deep learning with humans in the loop. *arXiv preprint arXiv:1506.03365*, 2015. 6

[39] J. Yu, Z. Lin, J. Yang, X. Shen, X. Lu, and T. S. Huang. Generative image inpainting with contextual attention. In *CVPR*, 2018. 1

[40] Z. Zhang, L. Yang, and Y. Zheng. Translating and segmenting multimodal medical volumes with cycle- and shape-consistency generative adversarial network. In *CVPR*, 2018. 1

[41] J. Zhao, M. Mathieu, and Y. LeCun. Energy-based generative adversarial network. In *ICLR*, 2017. 2