

Wasserstein Divergence for GANs

Jiqing Wu¹, Zhiwu Huang¹, Janine Thoma¹, Dinesh Acharya¹, and
Luc Van Gool^{1,2}

¹ Computer Vision Lab, ETH Zurich, Switzerland
^{{jwu,zhiwu.huang,jthoma,vangool}@vision.ee.ethz.ch,}
^{acharyad@student.ethz.ch}
² VISICS, KU Leuven, Belgium

Abstract. In many domains of computer vision, generative adversarial networks (GANs) have achieved great success, among which the family of Wasserstein GANs (WGANS) is considered to be state-of-the-art due to the theoretical contributions and competitive qualitative performance. However, it is very challenging to approximate the k -Lipschitz constraint required by the Wasserstein-1 metric (W-met). In this paper, we propose a novel Wasserstein divergence (W-div), which is a relaxed version of W-met and does not require the k -Lipschitz constraint. As a concrete application, we introduce a Wasserstein divergence objective for GANs (WGAN-div), which can faithfully approximate W-div through optimization. Under various settings, including progressive growing training, we demonstrate the stability of the proposed WGAN-div owing to its theoretical and practical advantages over WGANS. Also, we study the quantitative and visual performance of WGAN-div on standard image synthesis benchmarks, showing the superior performance of WGAN-div compared to the state-of-the-art methods.

Keywords: Wasserstein metric, Wasserstein divergence, GANs, Progressive growing.

1 Introduction

Over the past few years, we have witnessed the great success of generative adversarial networks (GANs) [1] for a variety of applications. GANs are a useful family of generative models that expresses generative modeling as a zero-sum game between two networks: A generator network produces plausible samples given some noise, while a discriminator network distinguishes between the generator’s output and real data. There are numerous works inspired by the original GANs, [2,3,4,5] to name a few. While GANs can produce visually pleasing samples, they lack a reliable way of measuring the difference between fake and real data distribution, which leads to unstable training.

To address this issue, [6] introduced the Wasserstein-1 metric (W-met) to the GAN framework. Compared to the Jensen-Shannon (JS) or the Kullback-Leibler (KL) divergence, W-met is considered to be more sensible for distributions supported by low dimensional manifolds. Given that the primal form of W-met

is intractable to compute, [6] proposed to use the dual form of W-met, which requires the k -Lipschitz constraint. A series of ideas [6,7,8,9] were proposed to approximate the dual W-met and achieved impressive results compared to the non-Wasserstein based GANs. However, they generally suffer from unsatisfying regularization for the k -Lipschitz constraint, mainly because it is a very strict constraint and non-trivial to approximate [9,10].

Other studies have tackled the stability issue from different angles. For example, [10] proposed a gradient-based regularizer associated with the f -divergence [11] to address the dimensional misspecification. In order to stabilize the training towards high resolution images, [12,13] applied deep stack architectures by incorporating extra information. Recently, building upon the dual W-met objective of [7], [14] presented a sophisticated progressive growing training scheme and obtained excellent high resolution images.

In this paper, we propose to resolve the k -Lipschitz constraint by introducing a relaxed version of W-met and incorporating it in the GAN framework. Our contributions can be summarized as follows:

1. We introduce a novel Wasserstein divergence (W-div) and prove that the proposed W-div is a symmetric divergence. Moreover, we explore the connection between the proposed W-div and W-met.
2. Benefiting from the non-challenging constraint required by the W-div, we introduce Wasserstein divergence GANs (WGAN-div) as its practical application. The proposed objective can faithfully approximate the corresponding W-div through optimization.
3. We demonstrate the stability of WGAN-div under various settings including progressive growing training. Also, we conduct various experiments on standard image synthesis benchmarks and present superior results of WGAN-div compared to the state-of-the-art methods, both quantitatively and qualitatively.

2 Background

Imagine there are two players in a game. One player (Generator) intends to generate visually plausible images, aiming to fool its opponent, while the opponent (Discriminator) attempts to discriminate real images from synthetic images. Such adversarial competition is the key idea behind GAN models. To measure the distance between real and fake data distributions, [1] proposed the objective

$$L_{\text{JS}}(\mathbb{P}_r, \mathbb{P}_g) = \mathbb{E}_{\mathbf{x} \sim \mathbb{P}_r} [\ln(f(\mathbf{x}))] + \mathbb{E}_{\tilde{\mathbf{x}} \sim \mathbb{P}_g} [\ln(1 - f(\tilde{\mathbf{x}}))], \quad (1)$$

which can be interpreted as the JS divergence up to a constant [15] and where f is a discriminative function. The model can thus be defined as a min-max optimization problem:

$$\min_{G} \max_{D} \mathbb{E}_{\mathbf{x} \sim \mathbb{P}_r} [\ln(D(\mathbf{x}))] + \mathbb{E}_{G(\mathbf{z}) \sim \mathbb{P}_g} [\ln(1 - D(G(\mathbf{z})))], \quad (2)$$

where G is the generator parametrized by a neural network and D is the discriminative neural network parametrizing f . Usually, we let \mathbf{z} be low dimensional random noise, and $\mathbf{x}, G(\mathbf{z})$ are the real and fake data satisfying the probability measures $\mathbb{P}_r, \mathbb{P}_g$.

Wasserstein GANs (WGANs). The rise of the Wasserstein-1 metric (W-met) in GAN models is primarily motivated by unstable training caused by the gradient vanishing problem [6]. Given two probability measures $\mathbb{P}_r, \mathbb{P}_g$, the W-met [16] is defined as

$$\mathcal{W}_1(\mathbb{P}_r, \mathbb{P}_g) = \sup_{f \in \text{Lip}_1} \mathbb{E}_{\mathbf{x} \sim \mathbb{P}_r} [f(\mathbf{x})] - \mathbb{E}_{\tilde{\mathbf{x}} \sim \mathbb{P}_g} [f(\tilde{\mathbf{x}})], \quad (3)$$

where Lip_1 is the function space of all f satisfying the 1-Lipschitz constraint $\|f\|_L \leq 1$. It is worth mentioning that \mathcal{W}_1 is invariant up to a positive scalar k if the Lipschitz constraint is modified to be k . \mathcal{W}_1 is believed to be more sensible to distributions supported by low dimensional manifolds such as image, video, etc. Generally, the existing Wasserstein GANs (WGANs) fall into two categories:
Weight Constraints. To approximately satisfy the Lipschitz constraint, [6] proposed a weight clipping method that imposes a hard threshold $c > 0$ on the weights \mathbf{w} of the discriminator D , which parametrizes f in Eq. 3:

$$\mathbf{w}' = \begin{cases} \mathbf{w} & \text{if } |\mathbf{w}| < c \\ c & \text{if } \mathbf{w} \geq c \\ -c & \text{if } \mathbf{w} \leq -c \end{cases} \quad (4)$$

This approach was proven to be unsatisfactory by [7], since through weight clipping, the neural network tends to learn oversimplified functions. Later, [8] proposed spectral normalization GANs (SNGANs). To impose the 1-Lipschitz constraint, SNGANs normalize the weights \mathbf{w}_i of each layer i by the L_2 matrix norm,

$$\mathbf{w}'_i = \frac{\mathbf{w}_i}{\|\mathbf{w}_i\|_2}. \quad (5)$$

Because the set of functions satisfying the local 1-Lipschitz constraint is merely a subset of the function space Lip_1 , such a constraint inevitably narrows the effective search space and entails a sub-optimal solution.

Gradient Constraints. To overcome the disadvantages of weight clipping, [7] introduced a gradient penalty term to Wasserstein GANs (WGAN-GP). The objective is defined as

$$L_{\text{GP}} = \underbrace{\mathbb{E}_{\mathbf{x} \sim \mathbb{P}_r} [f(\mathbf{x})] - \mathbb{E}_{\tilde{\mathbf{x}} \sim \mathbb{P}_g} [f(\tilde{\mathbf{x}})]}_{\text{Wasserstein term}} + k \underbrace{\mathbb{E}_{\hat{\mathbf{x}} \sim \mathbb{P}_y} [(\|\nabla f(\hat{\mathbf{x}})\|_2 - 1)^2]}_{\text{gradient penalty}}, \quad (6)$$

where ∇ is the gradient operator and \mathbb{P}_y is the distribution obtained by sampling uniformly along straight lines between points from the real and fake data distributions \mathbb{P}_r and \mathbb{P}_g . As pointed out by [9,10], with a finite number of training iterations on limited input samples, it is very difficult to guarantee the k -Lipschitz

constraint for the whole input domain. Thus, [9] further proposed Wasserstein GANs with a consistency term (CTGANs). Inspired by the original 1-Lipschitz constraint, CTGANs add the following term to Eq. 6,

$$\text{CT}|_{\mathbf{x}_1, \mathbf{x}_2} = \mathbb{E}_{\mathbf{x}_1, \mathbf{x}_2} [\max(0, \frac{d(f(\mathbf{x}_1), f(\mathbf{x}_2))}{d(\mathbf{x}_1, \mathbf{x}_2)} - c)], \quad (7)$$

where $\mathbf{x}_1, \mathbf{x}_2$ are two data points, d is a metric and c is a threshold. Recently, to improve stability and image quality, [14] proposed a training scheme in which GANs are grown progressively. In addition to progressive growing, [14] also proposed an objective $L_{\text{PG}} = L_{\text{GP}} + \text{PG}$, where

$$\text{PG} = \begin{cases} \mathbb{E}_{\hat{\mathbf{x}} \sim \mathbb{P}_y} [(\|\nabla f(\hat{\mathbf{x}})\|_2 - 750)^2 / 750^2] & \text{for CIFAR-10} \\ 0.001 \mathbb{E}_{\hat{\mathbf{x}} \sim \mathbb{P}_y} [\|\nabla f(\hat{\mathbf{x}})\|_2^2] & \text{for other datasets} \end{cases} \quad (8)$$

f-GANs. Outside the family of Wasserstein metrics, there is another important family of divergences—the f-divergences. [11] argued that f-divergence can be used for training generative samplers and proposed f-GANs. Since the f-GANs are vulnerable to the dimension mismatch between fake and real data, [10] proposed a gradient-based regularizer to stabilize the training and gave an example based on JS-divergence:

$$\begin{aligned} L_{\text{RJS}}(\mathbb{P}_r, \mathbb{P}_g) &= \mathbb{E}_{\mathbf{x} \sim \mathbb{P}_r} [\ln(f(\mathbf{x}))] + \mathbb{E}_{\tilde{\mathbf{x}} \sim \mathbb{P}_g} [\ln(1 - f(\tilde{\mathbf{x}}))] - k\Omega(\mathbb{P}_r, \mathbb{P}_g) \\ \Omega(\mathbb{P}_r, \mathbb{P}_g) &:= \mathbb{E}_{\mathbf{x} \sim \mathbb{P}_r} [(1 - f(\mathbf{x}))^2 \|\nabla f(\mathbf{x})\|^2] + \mathbb{E}_{\tilde{\mathbf{x}} \sim \mathbb{P}_g} [f(\tilde{\mathbf{x}})^2 \|\nabla f(\tilde{\mathbf{x}})\|^2]. \end{aligned} \quad (9)$$

Information Geometry. In information geometry, [17] studied the connections between the Wasserstein distance and the Kullback-Leibler (KL) divergence employed by early GANs. They exploit the fact that by regularizing the Wasserstein distance with entropy, the entropy relaxed Wasserstein distance introduces a divergence and naturally defines certain geometrical structures from the information geometry viewpoint.

3 Proposed Method

As discussed above, it is very challenging to approximate the W-met. This is due to the gap between limited input samples on the one hand and the strict 1-Lipschitz constraint on the whole input sample domain [18,9] on the other hand. At the same time, it is natural to ask whether there exists an optimal f^* for W-met (Eq. 3). According to [19], by solving a family of minimization problems given $p > 0$

$$f_p = \underset{f \in W_c^{1,p}}{\operatorname{argmin}} \mathbb{E}_{\mathbf{x} \sim \mathbb{P}_r} [f(\mathbf{x})] - \mathbb{E}_{\tilde{\mathbf{x}} \sim \mathbb{P}_g} [f(\tilde{\mathbf{x}})] + \frac{1}{p} \mathbb{E}_{\hat{\mathbf{x}} \sim \mathbb{P}_u} [\|\nabla f(\hat{\mathbf{x}})\|^p], \quad (10)$$

where \mathbb{P}_u is a Radon probability measure and $W_c^{1,p}$ is the Sobolev space containing all the functions f in L^p space with first order weak derivatives and compact support, we can find a sequence $p_k \rightarrow \infty$ such that $f_{p_k} \rightarrow -f^*$.

3.1 Wasserstein Divergence

The connection between Eq. 10 and W-met inspires us to propose a novel Wasserstein divergence (W-div) and we prove that it is indeed a valid symmetric divergence.

Theorem 1. (*Wasserstein divergence*) Let $\Omega \subset \mathbb{R}^n$ be an open, bounded, connected set and S be the set of all the Radon probability measures on Ω . If for some $p \neq 1, k > 0$ we define

$$\begin{aligned} \mathcal{W}'_{p,k} : S \times S &\rightarrow \mathbb{R}^- \cup \{0\} \\ (\mathbb{P}_r, \mathbb{P}_g) &\mapsto \inf_{f \in C_c^1(\Omega)} \mathbb{E}_{\mathbf{x} \sim \mathbb{P}_r} [f(\mathbf{x})] - \mathbb{E}_{\tilde{\mathbf{x}} \sim \mathbb{P}_g} [f(\tilde{\mathbf{x}})] + k \mathbb{E}_{\hat{\mathbf{x}} \sim \mathbb{P}_u} [\|\nabla f(\hat{\mathbf{x}})\|^p], \end{aligned} \quad (11)$$

where $C_c^1(\Omega)$ is the function space of all the first order differentiable functions on Ω with compact support, then $\mathcal{W}'_{p,k}$ is a symmetric divergence (up to the negative sign).

Proof. See supplementary material.

By imposing the $C_c^1(\Omega)$ function space, we rule out pathological functions with weak derivatives. Compared to the k -Lipschitz constraint, $f \in C_c^1(\Omega)$ is less restrictive, since $\|\nabla f\|$ does not need to be bounded by a hard threshold k . Given the universal approximation theorem and the modern architecture of neural networks—stacking differentiable layers to form a nonlinear differentiable function— $f \in C_c^1(\Omega)$ can easily be parameterized by a neural network.

In the following we further explore the connection between the proposed W-div and the original W-met in Eq. 3.

Remark 1. (*Upper bound*) Given Radon probability measures $\mathbb{P}_r, \mathbb{P}_g, \mathbb{P}_u$ on Ω , let

$$\mathcal{W}'_{\mathbb{P}_u}(\mathbb{P}_r, \mathbb{P}_g) := \inf_{f \in C_c^\infty(\Omega)} \mathbb{E}_{\mathbf{x} \sim \mathbb{P}_r} [f(\mathbf{x})] - \mathbb{E}_{\tilde{\mathbf{x}} \sim \mathbb{P}_g} [f(\tilde{\mathbf{x}})] + \frac{1}{2} \mathbb{E}_{\hat{\mathbf{x}} \sim \mathbb{P}_u} [\|\nabla f(\hat{\mathbf{x}})\|^2], \quad (12)$$

where C_c^∞ is the function space of all the smooth functions f with compact support. There exists an optimal f^* for \mathcal{W}_1 (Eq. 3) such that

$$\mathcal{W}_1(\mathbb{P}_r, \mathbb{P}_g) = \mathbb{E}_{\mathbf{x} \sim \mathbb{P}_r} [f^*(\mathbf{x})] - \mathbb{E}_{\tilde{\mathbf{x}} \sim \mathbb{P}_g} [f^*(\tilde{\mathbf{x}})], \quad (13)$$

and a $\mathcal{W}'_{\mathbb{P}_{u^*}}$ determined by f^* such that

$$\mathcal{W}'_{\mathbb{P}_{u^*}}(\mathbb{P}_r, \mathbb{P}_g) = \sup_{\mathbb{P}_u \in S} \mathcal{W}'_{\mathbb{P}_u}(\mathbb{P}_r, \mathbb{P}_g). \quad (14)$$

Please see the detailed discussion in [19].

Remark 1 indicates that $\mathcal{W}'_{\mathbb{P}_{u^*}}$, which is determined by the optimal f^* , is the upper bound of our W-div $\mathcal{W}'_{\mathbb{P}_u}$ ³.

³ $\mathcal{W}'_{\mathbb{P}_u}$ is a family of special cases of Eq. 11 with a more restrictive function space C_c^∞ .

Given the similarities between our proposed W-div and L_{GP} (Eq. 6), it may be interesting to know if there exists a divergence corresponding to L_{GP} . In general, the answer is no.

Remark 2. *If for $n > 0$ we let*

$$\mathcal{W}_{p,k,n}''(\mathbb{P}_r, \mathbb{P}_g) := \inf_{f \in C_c^1(\Omega)} \mathbb{E}_{\mathbf{x} \sim \mathbb{P}_r} [f(\mathbf{x})] - \mathbb{E}_{\tilde{\mathbf{x}} \sim \mathbb{P}_g} [f(\tilde{\mathbf{x}})] + k \mathbb{E}_{\hat{\mathbf{x}} \sim \mathbb{P}_u} [(\|\nabla f(\hat{\mathbf{x}})\| - n)^p], \quad (15)$$

*then $\mathcal{W}_{p,k,n}''$ is **not** a divergence in general.*

Counterexample. Assuming $\Omega = (-1, 1)$ and $p = 2$, it suffices to show that $\mathcal{W}_{2,k,n}''(\mathbb{P}_r, \mathbb{P}_g) \neq 0$ for $\mathbb{P}_r = \mathbb{P}_g$ almost everywhere. Since $\mathbb{E}_{\mathbf{x} \sim \mathbb{P}_r} [f(\mathbf{x})]$ and $\mathbb{E}_{\tilde{\mathbf{x}} \sim \mathbb{P}_g} [f(\tilde{\mathbf{x}})]$ cancel out, in order to guarantee $\mathcal{W}_{2,k,n}''(\mathbb{P}_r, \mathbb{P}_g) = 0$, $\|\nabla f(\hat{\mathbf{x}})\|$ must be equal to n on $(-1, 1)$, which implies that f is affine and contradicts the compact support constraint. For m -dimensional sets such as $(-1, 1)^m$ and an even integer p we need to employ the uniqueness argument of the Picard-Lindelöf Theorem to show that f can only be affine.

Remark 2 implies that the plausible statistic distance $\mathcal{W}_{2,k,1}''$ corresponding to Eq. 6 is neither a divergence, nor a valid metric.

3.2 Wasserstein Divergence GANs

Although W-met enjoys the tempting property of providing useful gradients, in practice, the original formulation $\mathbb{E}_{\mathbf{x} \sim \mathbb{P}_r} [f(\mathbf{x})] - \mathbb{E}_{\tilde{\mathbf{x}} \sim \mathbb{P}_g} [f(\tilde{\mathbf{x}})]$ of W-met cannot be directly applied as an objective without imposing the strict 1-Lipschitz constraint. In contrast, it is very straightforward to use our proposed W-div as an objective. Therefore, we introduce Wasserstein divergence GANs (WGAN-div). Our objective can be smoothly derived as

$$L_{\text{DIV}} = \mathbb{E}_{\mathbf{x} \sim \mathbb{P}_r} [f(\mathbf{x})] - \mathbb{E}_{\tilde{\mathbf{x}} \sim \mathbb{P}_g} [f(\tilde{\mathbf{x}})] + k \mathbb{E}_{\hat{\mathbf{x}} \sim \mathbb{P}_u} [\|\nabla f(\hat{\mathbf{x}})\|^p], \quad (16)$$

which is identical to the formulation of W-div without the infimum. Minimizing L_{DIV} faithfully approximates $\mathcal{W}_{p,k}'$, in a sense that the decrease of L_{DIV} indicates a better approximation of $\mathcal{W}_{p,k}'$. In comparison, lowering L_{GP} does not necessarily imply that L_{GP} approximates \mathcal{W}_1 better, since L_{GP} can be decreased at the cost of violating the gradient penalty term (Eq. 6).

By incorporating our objective L_{DIV} in the GAN framework, together with parameterizing $f \in C_c^1$ by a discriminator D and the fake data distribution \mathbb{P}_g by a generator G , our min-max optimization problem can be written as

$$\min_G \max_D \mathbb{E}_{G(\mathbf{z}) \sim \mathbb{P}_g} [D(G(\mathbf{z}))] - \mathbb{E}_{\mathbf{x} \sim \mathbb{P}_r} [D(\mathbf{x})] - k \mathbb{E}_{\hat{\mathbf{x}} \sim \mathbb{P}_u} [\|\nabla_{\hat{\mathbf{x}}} D(\hat{\mathbf{x}})\|^p], \quad (17)$$

where \mathbf{z} is random noise, \mathbf{x} is the real data, and $\hat{\mathbf{x}}$ is sampled as a linear combination of real and fake data points. For more studies of sampling strategies we refer readers to our supplementary material. The final algorithm is obtained as shown in Alg. 1. Following the good practice of [7], our building blocks for D and

Algorithm 1 The proposed WGAN-div algorithm

Require: Batch size m , generator G and discriminator D , power p , coefficient k , training iterations n , and other hyperparameters

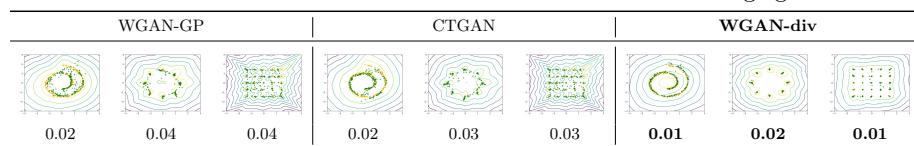
- 1: **for** $i \leftarrow 1$ to n **do**
- 2: Sample real data $\mathbf{x}_1, \dots, \mathbf{x}_m$ from \mathbb{P}_r
- 3: Sample Gaussian noise $\mathbf{z}_1, \dots, \mathbf{z}_m$ from $\mathcal{N}(0, 1)$
- 4: Sample vector $\boldsymbol{\mu} = (\mu_1, \dots, \mu_m)$ from uniform distribution $U[0, 1]$ such that
- 5: $\hat{\mathbf{x}}_j = (1 - \mu_j)\mathbf{x}_j + \mu_j G(\mathbf{z}_j)$
- 6: Update the weights \mathbf{w}_G of G by descending:

$$\mathbf{w}_G \leftarrow \text{Adam}(\nabla_{\mathbf{w}_G} (\frac{1}{m} \sum_{j=1}^m D(G(\mathbf{z}_j))), \mathbf{w}_G, \alpha, \beta_1, \beta_2)$$
- 7: Update the weights \mathbf{w}_D of D by descending:

$$\mathbf{w}_D \leftarrow \text{Adam}(\nabla_{\mathbf{w}_D} (\frac{1}{m} \sum_{j=1}^m D(\hat{\mathbf{x}}_j) - D(G(\mathbf{z}_j)) + k \|\nabla_{\hat{\mathbf{x}}_j} D(\hat{\mathbf{x}}_j)\|^p), \mathbf{w}_D, \alpha, \beta_1, \beta_2)$$
- 8: **end for**

Generator	Kernel size	Resampling	Output shape
Noise	–	–	128
Linear	–	–	$512 \times 4 \times 4$
Residual block	$[3 \times 3] \times 2$	Up	$512 \times 8 \times 8$
Residual block	$[3 \times 3] \times 2$	Up	$256 \times 16 \times 16$
Residual block	$[3 \times 3] \times 2$	Up	$128 \times 32 \times 32$
Residual block	$[3 \times 3] \times 2$	Up	$64 \times 32 \times 32$
Conv, tanh	3×3	–	$3 \times 64 \times 64$

Discriminator			
Conv	3×3	–	$64 \times 64 \times 64$
Residual block	$[3 \times 3] \times 2$	Down	$128 \times 32 \times 32$
Residual block	$[3 \times 3] \times 2$	Down	$256 \times 16 \times 16$
Residual block	$[3 \times 3] \times 2$	Down	$512 \times 8 \times 8$
Residual block	$[3 \times 3] \times 2$	Down	$512 \times 4 \times 4$
Linear	–	–	1

Table 1. The default architecture of WGAN-div for 64×64 image generation**Table 2.** Visual and FID comparison for generated samples (green dots) and real samples (yellow dots) on Swiss Roll, 8 Gaussians and 25 Gaussians. The value surfaces of the discriminators are also plotted.

G are residual blocks [20]. The default architecture of WGAN-div is presented in Tab. 1. We apply Adam optimization [21] to update G and D . We study the crucial hyperparameters such as the coefficient k and the power p in the next section.

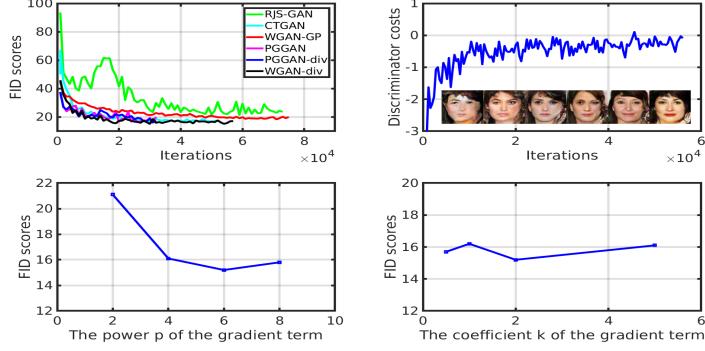


Fig. 1. Curves of FID vs. iteration (top left), Discriminator cost vs. iteration (top right), FID vs. power p (bottom left), and FID vs. coefficient k (bottom right) for WGAN-div on CelebA.

4 Experiments

In this section, we evaluate WGAN-div on toy datasets and three widely used image datasets—CIFAR-10, CelebA [22] and LSUN [23]. As a preliminary evaluation, we use low-dimensional datasets such as Swiss roll, 8 Gaussians and 25 Gaussians to justify that our proposed W-div can be more effectively learned than W-met used by WGAN-GP and CTGAN, in terms of more meaningful value surfaces of discriminator D i.e. f , and better generated data distribution (Tab. 2). Meanwhile, the three large scale datasets highlight a variety of challenges that WGAN-div should address and evaluation on them is adequate to support the advantages of WGAN-div.

Recently, [24] pointed out that the inception score (IS) [25] is not reliable because it does not incorporate the statistics of real image samples. As an alternative, they introduced the Fréchet inception distance (FID) to measure the difference between real and fake data distributions. Experiments verified that the FID score is consistent with visual judgment by humans. Later, [26] conducted a comprehensive study of the state-of-the-art GANs based on FID, which confirmed that FID provides fairer assessment. Hence, we consider the FID score as the major criterion for evaluating our method. Also, visual results are provided as a complementary form of verification.

We compare our WGAN-div to the state-of-the-art DCGAN [2], WGAN-GP [7], RJS-GAN [18], CTGAN [9], SNGAN [8], and PGGAN [14]. For each method, we apply the default architectures and hyperparameters recommended by their papers. The default architectures for G and D of WGAN-div follow the ResNet design [20] as presented in Tab. 1. We use Adam optimization [21] for updating G and D with a learning rate of 0.0002 for all three datasets. The number of training steps are 100000 for CelebA and CIFAR-10, and 200000 for LSUN. By cross validation we determine the number of iterations for D per training step to be 4 for CelebA and LSUN, and 5 for CIFAR-10.

4.1 Hyperparameter Study

We demonstrate the impact of two important hyperparameters—the power p and the coefficient k —on our WGAN-div method. Both of them control the gradient term of L_{DIV} . We report the obtained FID scores on the 64×64 CelebA dataset in the bottom row of Fig. 1. For a fixed optimal $p = 6$ and varying k , Fig. 1 shows that L_{DIV} is not sensitive to changes of k , with the FID score fluctuating mildly around 16. On the other hand, for a fixed $k = 2$ and changing p , we obtain the optimal FID at $p = 6$, which differs from the common choice $p = 2$ applied in WGAN methods. The fact that f_p (Eq. 10) converges to the optimal discriminator when p becomes larger may explain why L_{DIV} favors a larger power p . To summarize, our default p, k are determined to be $p = 6$ and $k = 2$.

4.2 Stability Study

In this section we evaluate the stability of our method to changes in architecture and compare it to other approaches. In this light, we apply various architecture settings for WGAN-div, WGAN-GP, and RJS-GAN, which represent three types of statistical distances: W-div, W-met, and f-divergence. We train these methods with two standard architectures—ConvNet as used by DCGAN [2] and ResNet [20], which is used by WGAN-GP [7]. Since batch normalization [27] (BN) is considered to be a key ingredient in stabilizing the training process [2], we also evaluate the FID without BN. In total, we use four settings: ResNet, ResNet without BN, ConvNet, and ConvNet without BN. As shown in Tab. 3, each column reports the visual and FID results obtained under the same architecture. Our WGAN-div achieves the best FID scores for all four settings. Tab. 3 also features corresponding visual results. Compared to WGAN-GP and RJS-GAN, WGAN-div produces more visually pleasing images and the visual quality remains more stable under changing settings. This experimental study confirms the advantages gained by our W-div and its identical objective L_{DIV} .

4.3 Evaluation on the Standard Training Scheme

In this experiment, we intend to fairly compare the performance of various GANs by ruling out the impact caused by fine-tuned training strategies. For this purpose, we follow the standard, i.e. non-growing, training scheme, which fixes the size and architecture of the discriminator and generator through the whole training process. We compute the FID scores for DCGAN, WGAN-GP, RJS-GAN, CTGAN, and WGAN-div. The configurations of the compared methods are set according to the recommendations from the authors. The results are reported in Tab. 4. WGAN-div reaches the best FID scores among the compared approaches, which quantitatively confirms the advantages of our method.

While the FID score of WGAN-div mildly outperforms the state-of-the-art methods on the dataset CIFAR-10, it demonstrates clearer improvements on the larger scale datasets CelebA and LSUN. Similarly, the facial results shown in Fig. 2 tell us that WGAN-div is better than the compared methods with regard

	ResNet	ResNet without BN	ConvNet	ConvNet without BN
WGAN-GP	18.4	20.3	21.2	24.6
				
RJS-GAN	21.4	23.2	21.7	22.4
				
WGAN-div	15.2	18.6	17.5	21.5
				

Table 3. FID scores and qualitative comparison of various architectures on CelebA.

to diversity and semantics. For example, Fig. 2 shows diverse faces generated by WGAN-div in terms of gender, age, facial expression and makeup. We can make the same conclusions on LSUN. The proposed WGAN-div outperforms the compared methods with a considerable margin both quantitatively and qualitatively. For example, WGAN-div achieves an FID score of 15.9 on LSUN, which is 4.4 lower than CTGAN, which is already an improved version of WGAN-GP, that introduced an extra regularizer to enhance WGAN-GP.

The examples of visually plausible bedrooms shown in Fig. 2 further highlight the advantages gained by introducing W-div in the GAN model. For the interpolation results in the latent space please check our supplementary material.

The top row of Fig. 1 reports the learning curve of the compared methods showing that the training process of our WGAN-div is comparatively stable and converges fast. It achieves top FID scores with less than 60K iterations. The top right plot of Fig. 1 illustrates the meaningful correlation between image quality and discriminator cost. It is worth mentioning that [24] proposed a two time-scale update method to generally improve the training of a variety of GANs. We believe that WGAN-div can also benefit from such a sophisticated update rule. However, due to the space limit, this is left for further studies.



Fig. 2. Visual results of WGAN-div and compared methods on CelebA (top row), LSUN (middle row), and CIFAR-10 (bottom row).

	CIFAR-10	CelebA	LSUN
DCGAN [2]	30.9	52.0	61.1
WGAN-GP [7]	18.8	18.4	26.8
RJS-GAN [10]	19.6	21.4	16.7
CTGAN [9]	18.6	16.4	20.3
SNGAN [8]	21.7*	-	-
WGAN-div	18.1	15.2	15.9

Table 4. FID comparison between WGAN-div and the state-of-the-art methods. The result with a * was taken from the original paper [8].

4.4 Evaluation on the Progressive Growing Training Scheme

Inspired by the success of PGGAN [14], which trained a W-met based GAN model in a progressive growing fashion, we evaluate how our objective L_{DIV} performs with this sophisticated training scheme. More specifically, we replace L_{PG} with our L_{DIV} while following the default configurations suggested in [14] and propose PGGAN-div. However, computing the FID scores for this experimental setting is challenging, as it is non-trivial to adapt existing FID models for eval-

		Resolution		CelebA	LSUN
PGGAN	64 × 64	16.3	17.8		
PGGAN-div	64 × 64	16.0	16.5		
PGGAN	128 × 128	14.1	15.4		
PGGAN-div	128 × 128	13.5	15.5		
PGGAN	256 × 256	-	15.1		
PGGAN-div	256 × 256	-	14.9		

Table 5. FID comparison between PGGAN-div and PGGAN at different resolutions.

ating higher resolution generated images. Since [14] does not specify the details of how their FID scores were computed for higher resolution images, we propose to downscale higher resolution images to 64×64 resolution and then compute the FID score. The resulting scores are reported in Tab. 5.

Interestingly, Tab. 5 shows that, for low resolution images, the FID score of PGGAN is slightly worse than the one of some top methods reported in Tab. 4, including WGAN-div. We believe that this phenomenon is not surprising. Since it is comparatively easy to learn a data distribution in low dimensional space, applying the standard training scheme suffices to achieve good FID scores. There is no need to introduce the sophisticated progressive growing strategy during the low dimensional phase. For higher resolution images (128×128 and 256×256) on the other hand, the FID scores for both PGGAN and PGGAN-div decrease with non-negligible margin. It is worth mentioning that our PGGAN-div slightly improves the FID scores over the original PGGAN, demonstrating the stability of our objective L_{DIV} under a sophisticated training scheme.

We also present the 256×256 visual results for CelebA-HQ (Fig. 3) and LSUN (Fig. 4). Since CelebA-HQ was generated by post-processing CelebA [14], we do not report its FID scores due to the distribution shift introduced by the artificial post-processing algorithms. The visual results in Fig. 3 and Fig. 4 demonstrate that our PGGAN-div is very competitive compared to the original PGGAN for both datasets. To summarize, we demonstrate the stability of our W-div objective under this training scheme.

5 Conclusion

In this paper, we introduced a novel Wasserstein divergence which does not require the 1-Lipschitz constraint. As a concrete example, we equip the GAN model with our Wasserstein divergence objective, resulting in WGAN-div. Both FID score and qualitative performance evaluation demonstrate the stability and superiority of the proposed WGAN-div over the state-of-the-art methods.

Acknowledgment. We would like to thank Nvidia for donating the GPUs used in this work.



Fig. 3. Visual results of PGGAN (top), PGGAN-div (bottom) on CelebA-HQ.

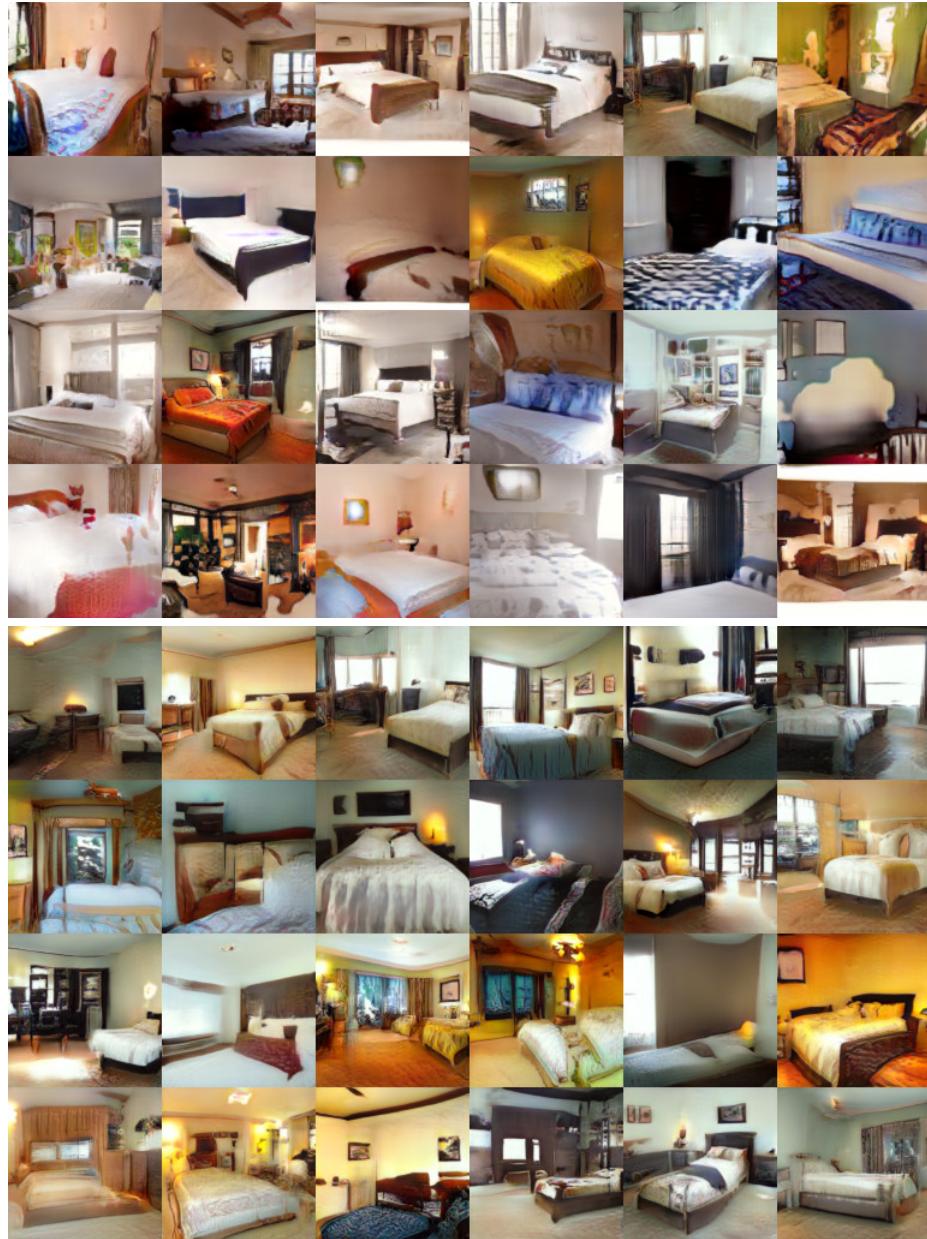


Fig. 4. Visual results of PGGAN (top), PGGAN-div (bottom) on 256×256 LSUN.

A Proof of Theorem 1

Proof. First of all, it is trivial to see that $\mathcal{W}'_{p,k}$ is nonpositive since we can always let $f = 0$. Next, we show identity:

“ \Rightarrow ” : Assume two probability measures $\mathbb{P}_r = \mathbb{P}_g$ almost everywhere, then the first two terms of Eq.(11) vanish, that is,

$$\mathbb{E}_{\mathbf{x} \sim \mathbb{P}_r} [f(\mathbf{x})] - \mathbb{E}_{\tilde{\mathbf{x}} \sim \mathbb{P}_g} [f(\tilde{\mathbf{x}})] = 0. \quad (18)$$

Since

$$k \mathbb{E}_{\hat{\mathbf{x}} \sim \mathbb{P}_u} [\|\nabla f(\hat{\mathbf{x}})\|^p] \geq 0, \quad (19)$$

and the equality holds if $f = 0$, hence $\mathcal{W}'_{p,k}(\mathbb{P}_r, \mathbb{P}_g) = 0$.

“ \Leftarrow ” : Assume $\mathbb{P}_r \neq \mathbb{P}_g$, there exists a $f \in C_c^1(\Omega)$ such that

$$\mathbb{E}_{\mathbf{x} \sim \mathbb{P}_r} [f(\mathbf{x})] - \mathbb{E}_{\tilde{\mathbf{x}} \sim \mathbb{P}_g} [f(\tilde{\mathbf{x}})] = -\lambda_1 < 0, \quad (20)$$

and w.l.o.g

$$k \mathbb{E}_{\hat{\mathbf{x}} \sim \mathbb{P}_u} [\|\nabla f(\hat{\mathbf{x}})\|^p] = \lambda_2 > 0. \quad (21)$$

We can always find a positive λ with $\lambda^{p-1} < \frac{\lambda_1}{\lambda_2}$ such that for $f' = \lambda f$ it satisfies

$$\begin{aligned} & \mathbb{E}_{\mathbf{x} \sim \mathbb{P}_r} [f'(\mathbf{x})] - \mathbb{E}_{\tilde{\mathbf{x}} \sim \mathbb{P}_g} [f'(\tilde{\mathbf{x}})] + k \mathbb{E}_{\hat{\mathbf{x}} \sim \mathbb{P}_u} [\|\nabla f'(\hat{\mathbf{x}})\|^p] = \\ & \lambda \left(\mathbb{E}_{\mathbf{x} \sim \mathbb{P}_r} [f(\mathbf{x})] - \mathbb{E}_{\tilde{\mathbf{x}} \sim \mathbb{P}_g} [f(\tilde{\mathbf{x}})] \right) + \lambda^p k \mathbb{E}_{\hat{\mathbf{x}} \sim \mathbb{P}_u} [\|\nabla f(\hat{\mathbf{x}})\|^p] = \\ & \lambda(-\lambda_1 + \lambda^{p-1} \lambda_2) < 0, \end{aligned} \quad (22)$$

taking the infimum over all f we have $\mathcal{W}'_{p,k}(\mathbb{P}_r, \mathbb{P}_g) < 0$. As for the symmetry, we observe that

$$\begin{aligned} \mathcal{W}'_{p,k}(\mathbb{P}_r, \mathbb{P}_g) &= \inf_{f \in C_c^1(\Omega)} \mathbb{E}_{\mathbf{x} \sim \mathbb{P}_r} [-f(\mathbf{x})] - \mathbb{E}_{\tilde{\mathbf{x}} \sim \mathbb{P}_g} [-f(\tilde{\mathbf{x}})] + k \mathbb{E}_{\hat{\mathbf{x}} \sim \mathbb{P}_u} [\|\nabla f(\hat{\mathbf{x}})\|^p] \\ &= \inf_{f \in C_c^1(\Omega)} \mathbb{E}_{\tilde{\mathbf{x}} \sim \mathbb{P}_g} [f(\tilde{\mathbf{x}})] - \mathbb{E}_{\mathbf{x} \sim \mathbb{P}_r} [f(\mathbf{x})] + k \mathbb{E}_{\hat{\mathbf{x}} \sim \mathbb{P}_u} [\|\nabla f(\hat{\mathbf{x}})\|^p] \\ &= \mathcal{W}'_{p,k}(\mathbb{P}_g, \mathbb{P}_r). \quad \square \end{aligned} \quad (23)$$

B Study of Sampling Strategies

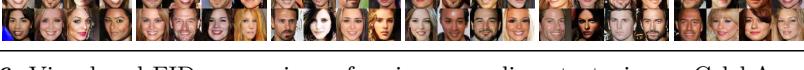
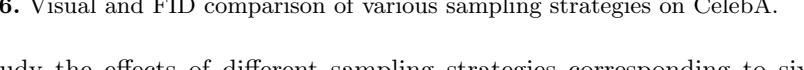
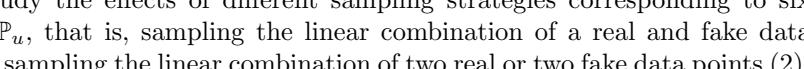
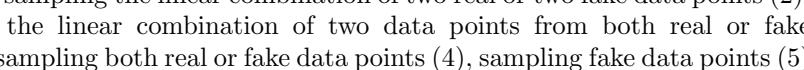
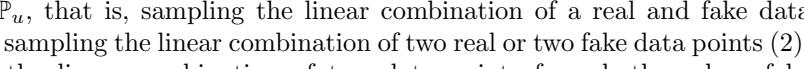
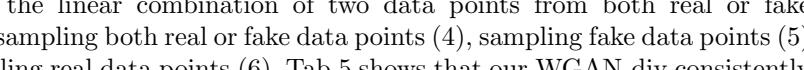
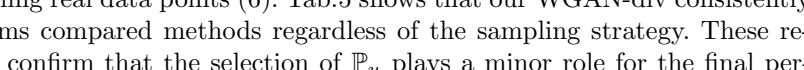
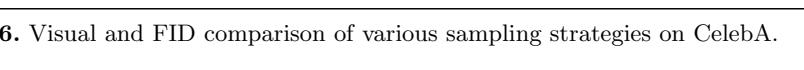
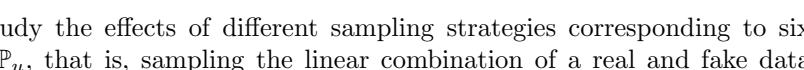
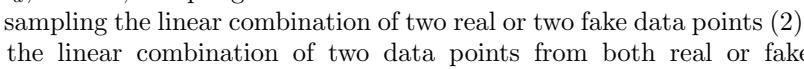
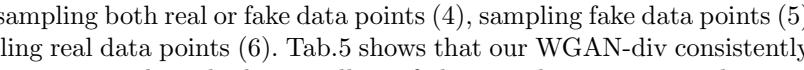
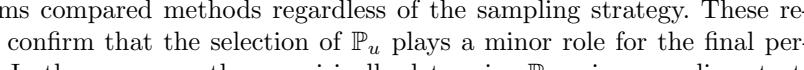
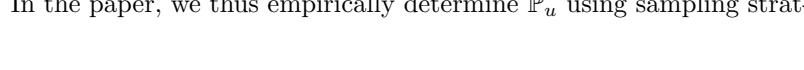
	(1)	(2)	(3)	(4)	(5)	(6)
WGAN-GP	18.4	20.1	19.3	19.0	18.3	17.0
						
CTGAN	16.4	16.5	17.7	17.2	17.9	17.5
						
WGAN-div	15.2	15.9	15.1	14.5	15.5	14.9
						

Table 6. Visual and FID comparison of various sampling strategies on CelebA.

We study the effects of different sampling strategies corresponding to six types of \mathbb{P}_u , that is, sampling the linear combination of a real and fake data point (1), sampling the linear combination of two real or two fake data points (2), sampling the linear combination of two data points from both real or fake data (3), sampling both real or fake data points (4), sampling fake data points (5) and sampling real data points (6). Tab.5 shows that our WGAN-div consistently outperforms compared methods regardless of the sampling strategy. These results also confirm that the selection of \mathbb{P}_u plays a minor role for the final performance. In the paper, we thus empirically determine \mathbb{P}_u using sampling strategy (1).

C Interpolation Results



Fig. 5. Interpolation results of 64×64 CelebA and LSUN.

References

1. Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., Bengio, Y.: Generative adversarial nets. In: NIPS. (2014)
2. Radford, A., Metz, L., Chintala, S.: Unsupervised representation learning with deep convolutional generative adversarial networks. arXiv preprint arXiv:1511.06434 (2015)
3. Berthelot, D., Schumm, T., Metz, L.: BEGAN: Boundary equilibrium generative adversarial networks. arXiv preprint arXiv:1703.10717 (2017)
4. Mao, X., Li, Q., Xie, H., Lau, R.Y., Wang, Z., Smolley, S.P.: Least squares generative adversarial networks. arXiv preprint arXiv:1611.04076 (2016)
5. Zhao, J., Mathieu, M., LeCun, Y.: Energy-based generative adversarial network. arXiv preprint arXiv:1609.03126 (2016)
6. Arjovsky, M., Chintala, S., Bottou, L.: Wasserstein generative adversarial networks. In: ICML. (2017)
7. Gulrajani, I., Ahmed, F., Arjovsky, M., Dumoulin, V., Courville, A.: Improved training of wasserstein gans. In: NIPS. (2017) 5767–5777
8. Miyato, T., Kataoka, T., Koyama, M., Yoshida, Y.: Spectral normalization for generative adversarial networks. arXiv preprint arXiv:1802.05957 (2018)
9. Wei, X., Gong, B., Liu, Z., Lu, W., Wang, L.: Improving the improved training of wasserstein gans: A consistency term and its dual effect. In: ICLR. (2018)
10. Roth, K., Lucchi, A., Nowozin, S., Hofmann, T.: Stabilizing training of generative adversarial networks through regularization. In: NIPS. (2017) 2015–2025
11. Nowozin, S., Cseke, B., Tomioka, R.: f-gan: Training generative neural samplers using variational divergence minimization. In: NIPS. (2016) 271–279
12. Zhang, H., Xu, T., Li, H., Zhang, S., Huang, X., Wang, X., Metaxas, D.: Stackgan: Text to photo-realistic image synthesis with stacked generative adversarial networks. In: ICCV. (2017)
13. Huang, X., Li, Y., Poursaeed, O., Hopcroft, J., Belongie, S.: Stacked generative adversarial networks. arXiv preprint arXiv:1612.04357 (2016)

14. Karras, T., Aila, T., Laine, S., Lehtinen, J.: Progressive growing of gans for improved quality, stability, and variation. arXiv preprint arXiv:1710.10196 (2017)
15. Arjovsky, M., Bottou, L.: Towards principled methods for training generative adversarial networks. In: NIPS Workshop. Volume 2016. (2017)
16. Villani, C.: Optimal transport: old and new. Volume 338. Springer Science & Business Media (2008)
17. Karakida, R., Amari, S.i.: Information geometry of wasserstein divergence. In: International Conference on Geometric Science of Information, Springer (2017) 119–126
18. Rothe, R., Timofte, R., Gool, L.V.: Deep expectation of real and apparent age from a single image without facial landmarks. International Journal of Computer Vision (IJCV) (July 2016)
19. Evans, L.C.: Partial differential equations and monge-kantorovich mass transfer. Current developments in mathematics **1997**(1) (1997) 65–126
20. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: CVPR. (2016) 770–778
21. Kingma, D., Ba, J.: Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980 (2014)
22. Liu, Z., Luo, P., Wang, X., Tang, X.: Deep learning face attributes in the wild. In: ICCV. (2015)
23. Yu, F., Seff, A., Zhang, Y., Song, S., Funkhouser, T., Xiao, J.: Lsun: Construction of a large-scale image dataset using deep learning with humans in the loop. arXiv preprint arXiv:1506.03365 (2015)
24. Heusel, M., Ramsauer, H., Unterthiner, T., Nessler, B., Hochreiter, S.: GANs trained by a two time-scale update rule converge to a local nash equilibrium. In: NIPS. (2017) 6629–6640
25. Salimans, T., Goodfellow, I., Zaremba, W., Cheung, V., Radford, A., Chen, X.: Improved techniques for training gans. In: NIPS. (2016) 2234–2242
26. Lucic, M., Kurach, K., Michalski, M., Gelly, S., Bousquet, O.: Are GANs created equal? a large-scale study. arXiv preprint arXiv:1711.10337 (2017)
27. Ioffe, S., Szegedy, C.: Batch normalization: Accelerating deep network training by reducing internal covariate shift. arXiv preprint arXiv:1502.03167 (2015)