SAN: INDUCING METRIZABILITY OF GAN WITH DISCRIMINATIVE NORMALIZED LINEAR LAYER

Yuhta Takida[1] **Masaaki Imaizumi**[2] **Takashi Shibuya**[1] **Chieh-Hsin Lai**[1]
Toshimitsu Uesaka[1] **Naoki Murata**[1] **Yuki Mitsufuji**[1,3]
[1]Sony AI, [2]The University of Tokyo, [3]Sony Group Corporation

ABSTRACT

Generative adversarial networks (GANs) learn a target probability distribution by optimizing a generator and a discriminator with minimax objectives. This paper addresses the question of whether such optimization actually provides the generator with gradients that make its distribution close to the target distribution. We derive *metrizable conditions*, sufficient conditions for the discriminator to serve as the distance between the distributions, by connecting the GAN formulation with the concept of sliced optimal transport. Furthermore, by leveraging these theoretical results, we propose a novel GAN training scheme called the Slicing Adversarial Network (SAN). With only simple modifications, a broad class of existing GANs can be converted to SANs. Experiments on synthetic and image datasets support our theoretical results and the effectiveness of SAN as compared to the usual GANs. We also apply SAN to StyleGAN-XL, which leads to a state-of-the-art FID score amongst GANs for class conditional generation on CIFAR10 and ImageNet 256×256. The code is available at `https://github.com/sony/san`.

1 INTRODUCTION

Utilizing a generative adversarial network (GAN) (Goodfellow et al., 2014) is a popular approach for generative modeling, and GANs have achieved remarkable performance in image (Brock et al., 2019; Karras et al., 2019; 2021), audio (Kumar et al., 2019; Donahue et al., 2019; Kong et al., 2020), and video (Tulyakov et al., 2018; Hao et al., 2021) domains. The aim of GAN is to learn a target probability measure via a neural network, called a generator. To achieve this, a discriminator is introduced, and the generator and discriminator are optimized in a minimax way.

Here, we pose the question of whether GAN optimization actually makes the generator distribution close to the target distribution. For example, likelihood-based generative models such as variational autoencoders (Kingma & Welling, 2014; Higgins et al., 2017; Zhao et al., 2019; Takida et al., 2022), normalizing flows (Tabak & Vanden-Eijnden, 2010; Tabak & Turner, 2013; Rezende & Mohamed, 2015), and denoising diffusion probabilistic models (Ho et al., 2020; Song et al., 2020; Nichol & Dhariwal, 2021) are optimized via the principle of minimizing the exact Kullback–Leibler divergence or its upper bound (Jordan et al., 1999). For GANs, in contrast, solving the minimization problem with the optimal discriminator is equivalent to minimizing a specific dissimilarity (Goodfellow et al., 2014; Nowozin et al., 2016; Lim & Ye, 2017; Miyato et al., 2018). Furthermore, GANs have been analyzed on the basis of the optimal discriminator assumption (Chu et al., 2020). However, real-world GAN optimizations hardly achieve maximization (Fiez et al., 2022), and analysis of GAN optimization without that assumption remains a challenge.

To analyze and investigate GAN optimization without the optimality assumption, there are various approaches from the perspectives of training convergence (Mescheder et al., 2017; Nagarajan & Kolter, 2017; Mescheder et al., 2018; Sanjabi et al., 2018; Xu et al., 2020), loss landscape around the saddle points of minimax problems (Farnia & Ozdaglar, 2020; Berard et al., 2020), and the smoothness of optimization (Fiez et al., 2022). Although these studies have been insightful for GAN stabilization, there has been little discussion as to whether trained discriminators indeed provide generator optimization with gradients that reduce dissimilarities.

Table 1: Common GAN losses do not simultaneously satisfy all the sufficient conditions given in Theorem 5.3. Thus, we propose SAN to address *direction optimality*. Even if a direction ω is the maximizer of the inner problems $\mathcal{V}$, it does not satisfy *direction optimality* except in Wasserstein GAN (see Sec. 6). In Appx. H, it is empirically demonstrated that a discriminator trained on Wasserstein GAN tends not to satisfy *separability*. We put $*$ in the last column since *injectivity* is not directly affected by objective functions but by other factors (see Appx. H for detailed explanations).

	Direction optimality	Separability	Injectivity
Wasserstein GAN	✓	weak	$*$
GAN (Hinge, Saturating, Non-saturating)	✗	✓	$*$
SAN (Hinge, Saturating, Non-saturating)	✓	✓	$*$

In this paper, we provide a unique perspective on GAN optimization that helps to consider whether a discriminator is *metrizable*.

Definition 1.1 (Metrizable discriminator)**.** *Let μ_θ and μ_0 be measures. Given an objective function $\mathcal{J}(\theta;\cdot)$ for θ, a discriminator f is $(\mathcal{J},\mathcal{D})$- or $\mathcal{J}$-metrizable for μ_θ and μ_0, if $\mathcal{J}(\theta;f)$ is minimized only with $\theta \in \arg\min_\theta \mathcal{D}(\mu_0,\mu_\theta)$ for a certain distance on measures, $\mathcal{D}(\cdot,\cdot)$.*

To evaluate the dissimilarity with a given GAN minimization problem $\mathcal{J}$, we are interested in other conditions besides the discriminator's optimality. Hence, we propose *metrizable conditions*, namely, *direction optimality*, *separability*, and *injectivity*, that induce a *$\mathcal{J}$-metrizable* discriminator. We first introduce a divergence, called functional mean divergence (FM*) in Sec. 3 and connect the FM* with the minimization objective function of Wasserstein GAN. Then, we obtain the *metrizable conditions* for Wasserstein GAN by investigating Question 1.2. We provide an answer to this question in Sec. 4 by relating the FM* to the concept of sliced optimal transport (Bonneel et al., 2015; Kolouri et al., 2019). Then, in Sec. 5, we formalize the proposed conditions for Wasserstein GAN and further extend the result to generic GAN.

Question 1.2. *Under what conditions is FM* a distance?*

Based on the derived *metrizable conditions*, we propose the Slicing Adversarial Network (SAN) in Sec. 6. As seen in Table 1, we find that optimal discriminators for most existing GANs do not satisfy *direction optimality*. Hence, we develop a modification scheme for GAN maximization problems to enforce *direction optimality* on our discriminator. Owing to the scheme's simplicity, GANs can easily be converted to SANs. We conduct experiments to verify our perspective and demonstrate that SANs are superior to GANs in certain generation tasks on synthetic and image datasets. In particular, we confirmed that SAN improves the state-of-the-art FID for conditional generation with StyleGAN-XL (Sauer et al., 2022) on CIFAR10 and ImageNet 256×256 despite the simple modifications.

2 Preliminaries

2.1 Notations

We consider a sample space $X \subseteq \mathbb{R}^{D_x}$ and a latent space $Z \subseteq \mathbb{R}^{D_z}$. Let $\mathcal{P}(X)$ be the set of all probability measures on X, and let $L^\infty(X,\mathbb{R}^D)$ denote the L^∞ space for functions $X \to \mathbb{R}^D$. Let μ (or ν) represent a probability measure with probability density function I_μ (or I_ν). Denote $d_h(\mu,\nu) := \mathbb{E}_{x\sim\mu}[h(x)] - \mathbb{E}_{x\sim\nu}[h(x)]$. We use the notation of the pushforward operator $\sharp$, which is defined as $g_\sharp\sigma := \sigma(g^{-1}(B))$ for $B \in X$ with a function $g : Z \to X$ and a probability measure $\sigma \in \mathcal{P}(Z)$. We denote the Euclidean inner product by $\langle\cdot,\cdot\rangle$. Lastly, $\hat{(\cdot)}$ denotes a normalized vector. Please also refer to Appx. A for some notations defined in the rest of the papers.

2.2 Problem Formulation in GANs

Assume that we have data obtained by discrete sampling from a target probability distribution $\mu_0 \in \mathcal{P}(X)$. Then, we introduce a trainable generator function $g_\theta : Z \to X$ with parameter $\theta \in \mathbb{R}^{D_\theta}$ to model a trainable probability measure as $\mu_\theta = g_{\theta\sharp}\sigma$ with $\sigma \in \mathcal{P}(Z)$. The aim of generative modeling here is to learn g_θ so that it approximates the target measure as $\mu_\theta \approx \mu_0$.

For generative modeling in GAN, we introduce the notion of a discriminator $f \in \mathcal{F}(X) \subset L^\infty(X, \mathbb{R})$. We formulate the GAN's optimization problem as a two-player game between the generator and discriminator with $\mathcal{V} : \mathcal{F}(X) \times \mathcal{P}(X) \to \mathbb{R}$ and $\mathcal{J} : \mathbb{R}^{D_\theta} \times \mathcal{F}(X) \to \mathbb{R}$, as follows:

$$\max_{f \in \mathcal{F}(X)} \mathcal{V}(f; \mu_\theta) \quad \text{and} \quad \min_{\theta \in \mathbb{R}^{D_\theta}} \mathcal{J}(\theta; f). \tag{1}$$

Regarding the choices of $\mathcal{V}$ and $\mathcal{J}$, there are GAN variants (Goodfellow et al., 2014; Nowozin et al., 2016; Arjovsky et al., 2017) that lead to different dissimilarities between μ_0 and μ_θ with the maximizer f. In this paper, we use a representation of the discriminator in an inner-product form:

$$f(x) = \langle \omega, h(x) \rangle, \tag{2}$$

where $\omega \in \mathbb{S}^{D-1}$ and $h(x) \in L(X, \mathbb{R}^D)$. The form is naturally represented by a neural network[1].

2.3 Wasserstein Distance and Its Use for GANs

We consider the Wasserstein-p distance (Villani, 2009) between probability measures μ and ν

$$W_p(\mu, \nu) := \left(\inf_{\pi \in \Pi(\mu,\nu)} \int_{X \times X} \|x - x'\|_p^p d\pi(x, x') \right)^{\frac{1}{p}}, \tag{3}$$

where $p \in [1, \infty)$ and $\Pi(\mu, \nu)$ is the set of all coupling measures whose marginal distributions are μ and ν. The idea of Wasserstein GAN is to learn a generator by minimizing the Wasserstein-1 distance between μ_0 and μ_θ. For this goal, one could adopt the Kantorovich—Rubinstein (KR) duality representation to rewrite Eq. (3) and obtain the following optimization problem:

$$\max_{f \in \mathcal{F}_{\text{Lip}}(X)} \mathcal{V}_{\text{W}}(f; \mu_\theta) := d_f(\mu_0, \mu_\theta) \tag{4}$$

where $\mathcal{F}_{\text{Lip}}$ denotes the class of 1-Lipschitz functions. In contrast, we formulate an optimization problem for the generator as a minimization of the right side of Eq. (4) w.r.t. the generator parameter:

$$\min_{\theta \in \mathbb{R}^{D_\theta}} \mathcal{J}_{\text{W}}(\theta; f) := -\mathbb{E}_{x \sim \mu_\theta}[f(x)]. \tag{5}$$

2.4 Sliced Optimal Transport

The Wasserstein distance is highly intractable when the dimension D_x is large (Arjovsky et al., 2017). However, it is well known that *sliced optimal transport* can be applied to break this intractability by projecting the data on a one-dimensional space. That is, for the $D_x = 1$ case, the Wasserstein distance has a closed-form solution:

$$W_p(\mu, \nu) = \left(\int_0^1 |F_\mu^{-1}(\rho) - F_\nu^{-1}(\rho)|^p d\rho \right), \tag{6}$$

where $F_\mu^{-1}(\cdot)$ denotes the quantile function for I_μ. The closed-form solution for a one-dimensional space prompted the emergence of the concept of sliced optimal transport.

In the original sliced Wasserstein distance (SW) (Bonneel et al., 2015), a probability density function I on the data space X is mapped to a probability density function of $\xi \in \mathbb{R}$ by the standard Radon transform (Natterer, 2001; Helgason, 2011) as $\mathcal{R}I(\xi, \omega) := \int_X I(x)\delta(\xi - \langle x, \omega \rangle)dx$, where $\delta(\cdot)$ is the Dirac delta function and $\omega \in \mathbb{S}^{D_x - 1}$ is a direction. The sliced Wasserstein distance between μ and ν is defined as $SW_p^{h,\omega}(\mu, \nu) := (\int_{\omega \in \mathbb{S}^{D_x-1}} W_p^p(\mathcal{R}I_\mu(\cdot, \omega), \mathcal{R}I_\nu(\cdot, \omega))d\omega)^{1/p}$. Intuitively, the idea behind this distance is to decompose high-dimensional distributions into an infinite number of pairs of tractable distributions by linear projections.

Various extensions of the sliced Wasserstein distance have been proposed (Kolouri et al., 2019; Deshpande et al., 2019; Nguyen et al., 2021). Here, we review an extension called augmented sliced Wasserstein distance (ASW) (Chen et al., 2022). Given a measurable injective function $h : X \to$

[1]In practice, the discriminator f is implemented as in Eq. (2), e.g., $f_\phi(x) = w_{\phi_L}^\top (l_{\phi_{L-1}} \circ l_{\phi_{L-2}} \circ \cdots \circ l_{\phi_1})(x)$ with nonlinear layers $\{l_{\phi_\ell}\}_{\ell=1}^{L-1}$, $w_{\phi_L} \in \mathbb{R}^D$, and their weights $\phi := \{\phi_\ell\}_{\ell=1}^L \in \mathbb{R}^{d_\phi}$.

$\mathbb{R}^D$, the distance is obtained via the spatial Radon transform (SRT), which is defined for any $\xi \in \mathbb{R}$ and $\omega \in \mathbb{S}^{d-1}$, as follows:

$$\mathcal{S}^h I(\xi, \omega) := \int_X I(x)\delta(\xi - \langle \omega, h(x) \rangle) dx. \tag{7}$$

The ASW-p is then obtained via the SRT in the same fashion as the standard SW:

$$ASW_p^h(\mu, \nu) := \left(\int_{\omega \in \mathbb{S}^{d-1}} W_p^p(\mathcal{S}^h I_\mu(\cdot, \omega), \mathcal{S}^h I_\nu(\cdot, \omega)) d\omega \right)^{\frac{1}{p}}. \tag{8}$$

Equation (6) can be used to evaluate the integrand in Eq. (8), which is usually evaluated via approximated quantile functions with sorted finite samples from $\mathcal{S}^h I_\nu(\cdot, \omega)$ and $\mathcal{S}^h I_\nu(\cdot, \omega)$.

3 Formulation of Question 1.2

We introduce a divergence called the functional mean divergence, FM or FM*, which is defined for a given functional space or function, respectively. Minimization of the FM* can be formulated as an optimization problem involving $\mathcal{J}_{\mathrm{W}}$, and we cast Question 1.2 in this context. In Sec. 4, we provide an answer to this question, which in turn provides the *metrizable conditions* with $\mathcal{J}_{\mathrm{W}}$ in Sec. 5.

3.1 Proposed Framework: Functional Mean Divergence

We start by defining the FM with a given functional space.

Definition 3.1 (Functional Mean Divergence (FM)). *We define a family of FMs as*

$$\mathscr{D}_D^{FM} := \left\{ (\mu, \nu) \mapsto \max_{h \in \mathcal{F}(X)} \|d_h(\mu, \nu)\|_2 \,|\mathcal{F}(X) \subseteq L^\infty(X, \mathbb{R}^D) \right\}, \tag{9}$$

Further, we denote an instance in the family as $FM_{\mathcal{F}}(\mu, \nu) \in \mathscr{D}_D^{FM}$, where $\mathcal{F}(X) \subseteq L^\infty(X, \mathbb{R}^D)$.

By definition, the FM family includes the integral probability metric (IPM) (Müller, 1997), which includes the Wasserstein distance in KR form as a special case.

Proposition 3.2. *For $\mathcal{F}(X) \in L^\infty(X, \mathbb{R})$, $IPM_{\mathcal{F}}(\cdot, \cdot) := \max_{f \in \mathcal{F}} d_f(\cdot, \cdot) \in \mathscr{D}_1^{FM}$.*

The FM is an extension of the IPM to deal with vector-valued functional spaces. Although the FM with a properly selected functional space yields a distance between target distributions, the maximization in Eq. (9) is generally difficult to achieve. Instead, we use the following metric, which is defined for a given function.

Definition 3.3 (Functional Mean Divergence* (FM*)). *Given a functional space $\mathcal{F}(X) \subseteq L^\infty(X, \mathbb{R}^D)$, we define a family of FMs* as*

$$\mathscr{D}_{\mathcal{F}}^{FM^*} := \{(\mu, \nu) \mapsto \|d_h(\mu, \nu)\|_2 | h \in \mathcal{F}(X)\}. \tag{10}$$

Further, we denote an instance in the family as $FM_h^*(\mu, \nu) \in \mathscr{D}_{\mathcal{F}}^{FM^*}$, where $h \in \mathcal{F}(X)$.

We are interested in Question 3.4, which is a mathematical formulation of Question 1.2. We give an answer to this question in Sec. 4. Since optimization of the FM* is related to $\mathcal{J}_{\mathrm{W}}$ in Sec. 3.2, the conditions in Question 3.4 enable us to derive $(\mathcal{J}_{\mathrm{W}}, FM_h^*)$*-metrizable conditions* in Sec. 5.

Question 3.4. *Under what conditions for $\mathcal{F}(X) \in L^\infty(X, \mathbb{R}^D)$ are $FM_h^*(\cdot, \cdot) \in \mathscr{D}_{\mathcal{F}}^{FM^*}$ distances?*

3.2 Direction Optimality to Connect FM* and $\mathcal{J}_{\mathrm{W}}$

Optimization of the FM* with a given $h \in L^\infty(X, \mathbb{R}^D)$ returns us to an optimization problem involving $\mathcal{J}_{\mathrm{W}}$.

Proposition 3.5 (*Direction optimality* connects FM* and $\mathcal{J}_{\mathrm{W}}$). *Let ω be on $\mathbb{S}^{D-1}$. For any $h \in L^\infty(X, \mathbb{R}^D)$, minimization of $FM_h^*(\mu_\theta, \mu_0)$ is equivalent to optimization of $\min_{\theta \in \mathbb{R}^{D_\theta}} \max_{\omega \in \mathbb{S}^{D-1}} \mathcal{J}_W(\theta; \langle \omega, h \rangle)$. Thus, $\nabla_\theta FM_h^*(\mu_\theta, \mu_0) = \nabla_\theta \mathcal{J}_W(\theta; \langle \omega^*, h \rangle)$, where ω^* is the optimal solution (direction) given as follows:*

$$\omega^* = \underset{\omega \in \mathbb{S}^{D-1}}{\arg\max}\ d_{\langle \omega, h \rangle}(\mu_0, \mu_\theta). \tag{11}$$

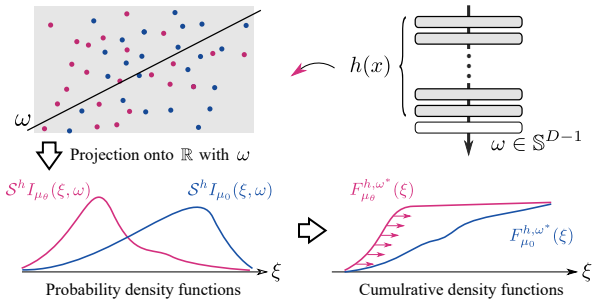


Figure 1: Discriminator decomposition into inner-product form $\langle \omega, h(x) \rangle$. Direction ω projects $h(x)$ onto $\mathbb{R}$. In this figure, h is *separable* because $F^{h,\omega^*}_{\mu_0}(\xi) \leq F^{h,\omega^*}_{\mu_\theta}(\xi)$ for all $\xi \in \mathbb{R}$ (see Definition 4.2).

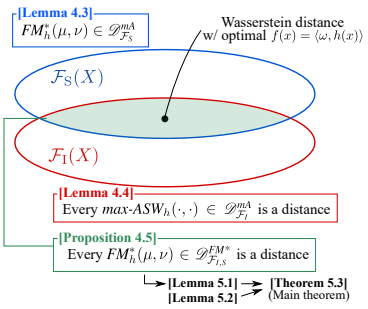


Figure 2: Outline of Sec. 4. Proposition 4.5 is a major step toward our main theorem.

The proof can be found in Appx. D, where the key idea involves using the Cauchy-Schwartz inequality, $\|d_h(\mu_0, \mu_\theta)\|_2 \geq \langle \omega, d_h(\mu_0, \mu_\theta) \rangle$ with $\omega \in \mathbb{S}^{D-1}$, and the linearity of $d_h(\mu_0, \mu_\theta)$, leading to $\|d_h(\mu_0, \mu_\theta)\|_2 \geq d_{\langle \omega, h \rangle}(\mu_0, \mu_\theta)$.

Recall that we formulated the discriminator in the inner-product form (2), which is aligned with Proposition 3.5. Here, we introduced a notion of *optimal direction*, which most distinguishes μ_0 and μ_θ in the feature space given by h, intuitively (refer to Figure 1 for illustration). We refer to the condition for the direction in Eq. (11) as *direction optimality*. It is obvious here that, given function h, the maximizer ω^* becomes $\hat{d}_h(\mu_0, \mu_\theta)$.

From the discussion in this section, Proposition 3.5 supports the notion that investigating Question 3.4 will reveal the $(\mathcal{J}_{\mathrm{W}}, FM^*_h)$*-metrizable conditions*.

4 Conditions for Metrizability: Analysis by Max-ASW

4.1 Strategy for Answering Question 3.4

We consider the conditions of $\mathcal{F}(X)$ for Question 3.4 in the context of sliced optimal transport. To this end, we define a variant of sliced optimal transport, called maximum augmented sliced Wasserstein divergence (max-ASW) in Definition 4.1. In Sec. 4.2, we first introduce a condition called the *separable* condition, where divergences included in the FM family are also included in the max-ASW family. In Sec. 4.3, we further introduce a condition called the *injective* condition, where the max-ASW is a distance. Finally, imposing these conditions on $\mathcal{F}(X)$ brings us the desired conditions (see Fig. 2 for the discussion flow).

Definition 4.1 (Maximum Augmented Sliced Wasserstein Divergence (max-ASW))**.** *Given a functional space $\mathcal{F}(X) \subseteq L^\infty(X, \mathbb{R}^D)$, we define a family of max-ASWs as*

$$\mathscr{D}^{mA}_{\mathcal{F}} := \left\{ (\mu, \nu) \mapsto \max_{\omega \in \mathbb{S}^{D-1}} W_1 \left(\mathcal{S}^h I_\mu(\cdot, \omega), \mathcal{S}^h I_\nu(\cdot, \omega) \right) | h \in \mathcal{F}(X) \right\}. \tag{12}$$

Further, we denote an instance of the family as max-ASW$_h(\mu, \nu) \in \mathscr{D}^{mA}_{\mathcal{F}}$, where $h \in \mathcal{F}(X)$.

Note that although the formulation in the definition includes the SRT, similarly to the ASW in Eq. (8), they differ in terms of the method of direction sampling (ω).

4.2 *Separability* for Equivalence of FM* and Max-ASW

We introduce a property for the function h, called *separability*, which connects the FM* and the max-ASW as in Lemma 4.3.

Definition 4.2 (Separable)**.** *Given $\mu, \nu \in \mathcal{P}(X)$, let ω be on $\mathbb{S}^{D-1}$, and let $F^{h,\omega}_\mu(\cdot)$ be the cumulative distribution function of $\mathcal{S}^h I_\mu(\cdot, \omega)$. If $\omega^* = \hat{d}_h(\mu, \nu)$ satisfies $F^{h,\omega^*}_\mu(\xi) \leq F^{h,\omega^*}_\nu(\xi)$ for any $\xi \in \mathbb{R}$, $h \in L^\infty(X, \mathbb{R}^D)$ is separable for those probability measures. We denote the class of all these separable functions for them as $\mathcal{F}_{S(\mu,\nu)}(X)$ or $\mathcal{F}_S$ for notation simplicity.*

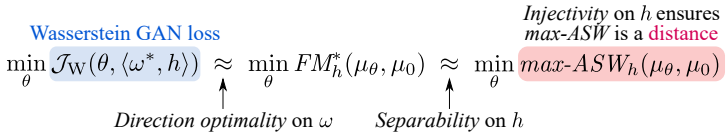


Figure 3: An example with Wasserstein GAN loss: the metrizable conditions (*direction optimality*, *separability*, and *injectivity*) ensure that Wasserstein GAN loss evaluates the distance between data and generator distributions.

Lemma 4.3. *Given $\mu, \nu \in \mathcal{P}(X)$, every $h \in \mathcal{F}_{S(\mu,\nu)}(X)$ satisfies $FM_h^*(\mu,\nu) \in \mathscr{D}_{\mathcal{F}_S}^{mA}$.*

Intuitively, *separability* ensures that optimal transport maps for all the samples from $\mathcal{S}^h I_{\mu_\theta}(\cdot,\omega)$ to $\mathcal{S}^h I_{\mu_0}(\cdot,\omega)$ share the same direction as in Figure 1. For a general function $h \in L^\infty(X,\mathbb{R}^D)$, $FM_h^*(\cdot,\cdot)$ is not necessarily included in the max-ASW family, i.e., $\mathcal{D}(\cdot,\cdot) \in \mathscr{D}_{\mathcal{F}}^{\mathrm{mA}}(\cdot,\cdot) \not\Leftrightarrow \mathcal{D}(\cdot,\cdot) \in \mathscr{D}_{\mathcal{F}}^{\mathrm{FM}*}$ for general $\mathcal{F}(X) \subseteq L^\infty(X,\mathbb{R}^D)$. Given h, calculation of the max-ASW via Eq. (6) generally involves evaluating the sign of the difference between the quantile functions. Intuitively, the equivalence between the FM and max-ASW distances holds if the sign is always positive regardless of $\rho \in [0,1]$; otherwise, the sign's dependence on ρ breaks the equivalence.

4.3 Injectivity for Max-ASW To Be A Distance

Imposing *injectivity* on h guarantees that the induced max-ASW is indeed a distance. Intuitively, *injectivity* prevents the loss of information from the original samples $x \sim \mu_\theta$ and $x \sim \mu_0$.

Lemma 4.4. *Every max-ASW$_h(\cdot,\cdot) \in \mathscr{D}_{\mathcal{F}_I}^{mA}$ is a distance, where $\mathcal{F}_I$ indicates a class of all the injective functions in $L^\infty(X,\mathbb{R}^D)$.*

Lemmas 4.3 and 4.4immediately tell us that $FM_h^*(\mu,\nu)$ with a *separable* and *injective* h is indeed a distance because it is included in the family of max-ASW *distances*, as stated in Proposition 4.5. With the result, we now have one of the answers to Question 3.4, which hints at our main theorem.

Proposition 4.5. *Given $\mu, \nu \in \mathcal{P}(X)$, every $FM_h^*(\mu,\nu) \in \mathscr{D}_{\mathcal{F}_{I,S}}^{FM*}$ is indeed a distance, where $\mathcal{F}_{I,S}$ indicates $\mathcal{F}_I \cap \mathcal{F}_S$.*

5 Does GAN Training Actually Minimize Distance?

We present Theorem 5.3, which is our main theoretical result and gives sufficient conditions for the discriminator to be $\mathcal{J}$-metrizable. This theorem inspires us to propose modification schemes for GAN training in Sec. 6.

We directly apply the discussion in the previous section to $\mathcal{J}_{\mathrm{W}}$, and by extending the result for Wasserstein GAN to a general GAN, we derive the main result. First, a simple combination of Propositions 3.5 and 4.5 yields the following lemma. Lemma 5.1 provides the conditions for the discriminator to be $\mathcal{J}_{\mathrm{W}}$-*metrizable* (see Fig. 3 for how each condition works).

Lemma 5.1 ($\mathcal{J}_{\mathrm{W}}$-metrizable)**.** *Given $h \in \mathcal{F}_I \cap \mathcal{F}_S$ with $\mathcal{F}_I, \mathcal{F}_S \subseteq L^\infty(X,\mathbb{R}^D)$, let $\omega^* \in \mathbb{S}^{D-1}$ be $\hat{d}_h(\mu_0,\mu_\theta)$. Then $f(x) = \langle \omega, h(x)\rangle$ is $(\mathcal{J}_W, FM_h^*)$-metrizable.*

Next, we generalize this result to more generic minimization problems. Our scope here is the minimization problems of general GANs that are formalized in the form $\mathcal{J}(\theta; f) = \mathbb{E}_{x\sim\mu_\theta}[R_{\mathcal{J}} \circ f(x)]$ with $R_{\mathcal{J}} : \mathbb{R} \to \mathbb{R}$. We utilize the gradient of such minimization problems w.r.t. θ:

$$\nabla_\theta \mathcal{J}(\theta; f) = -\mathbb{E}_{z\sim\sigma}\left[r_{\mathcal{J}} \circ f(g_\theta(z)) \nabla_\theta f(g_\theta(z))\right], \tag{13}$$

where $r_{\mathcal{J}}(\cdot)$ is the derivative of $R_{\mathcal{J}}(\cdot)$, as listed in Table 2. By ignoring a scaling factor, Eq. (13) can be regarded as a gradient of $d_f(\tilde{\mu}_0^{r_{\mathcal{J}}\circ f}, \tilde{\mu}_\theta^{r_{\mathcal{J}}\circ f})$, where $\tilde{\mu}^{r\circ f}$ is defined via $I_{\tilde{\mu}^{r\circ f}}(x) \propto r \circ f(x) I_\mu(x)$. To examine whether updating the generator with the gradient in Eq. (13) can minimize a certain distance between μ_0 and μ_θ, we introduce the following lemma.

Lemma 5.2. *For any $\rho : X \to \mathbb{R}_+$ and a distance for probability measures $\mathcal{D}(\cdot,\cdot)$, $\mathcal{D}(\tilde{\mu}^\rho, \tilde{\nu}^\rho)$ indicates a distance between μ and ν.*

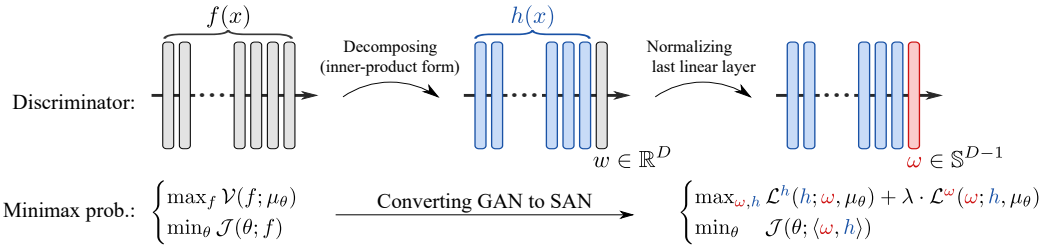


Figure 4: Converting GAN to SAN requires only simple modifications to discriminators.

We finally derive our main result, the *$\mathcal{J}$-metrizable conditions* for μ_0 and μ_θ, by leveraging Lemma 5.2 and applying Propositions 3.5 and 4.5 to $\tilde{\mu}_0^{r_{\mathcal{J}}\circ f}$ and $\tilde{\mu}_\theta^{r_{\mathcal{J}}\circ f}$, The theorem suggests that the discriminator can serve as a distance between the generator and target distributions even if it is not the optimal solution to the original maximization problem $\mathcal{V}$.

Theorem 5.3 ($\mathcal{J}$-Metrizability). *Given a functional $\mathcal{J}(\theta;f) := \mathbb{E}_{x\sim\mu_\theta}[R\circ f(x)]$ with $R:\mathbb{R}\to\mathbb{R}$ whose derivative, denoted as R', is positive, let $h \in L^\infty(X,\mathbb{R}^D)$ and $\omega \in \mathbb{S}^{D-1}$ satisfy the following conditions. Then, $f(x) = \langle\omega, h(x)\rangle$ is $\mathcal{J}$-metrizable for μ_θ and μ_0.*

- ***(Direction optimality)*** $\omega = \arg\max_{\tilde{\omega}\in\mathbb{S}^{D-1}} d_{\langle\tilde{\omega},h\rangle}(\tilde{\mu}_0^{R'\circ f}, \tilde{\mu}_\theta^{R'\circ f})$,
- ***(Separability)*** *h is separable for $\tilde{\mu}_0^{R'\circ f}$ and $\tilde{\mu}_\theta^{R'\circ f}$,*
- ***(Injectivity)*** *h is an injective function.*

In particular, *direction optimality* is the most interesting aspect for investigating GAN among the three conditions. It should be noted that most existing GANs besides Wasserstein GAN do not satisfy *direction optimality* with the maximizer ω of $\mathcal{V}$, which will be explained in Sec. 6. Therefore, we will focus on *direction optimality* in the next section. Please refer to Appx. H for a more detailed explanation and empirical investigation about the other conditions. Our observations regarding inductive effects of GAN losses to the metrizable conditions are summarized in Table 1. Note that *injectivity* cannot be directly controlled by loss designs as mentioned in the table.

6 Slicing Adversarial Network

This section describes our proposed model, the Slicing Adversarial Network (SAN) to achieve the *direction optimality* of ω in Theorem 5.3 while keeping *separability* of h. We develop SAN by modifying the maximization problem $\mathcal{V}$ to guarantee that the optimal solution ω achieves *direction optimality*. The proposed modification scheme is applicable to most existing GAN objectives and does not necessitate additional exhaustive hyperparameter tuning or computational complexity.

As mentioned in Sec. 5, given a function h, the maximization problems in most GANs (excluding Wasserstein GAN) cannot achieve *direction optimality* with the maximum solution of $\mathcal{V}$ (see Table 1). We use hinge GAN as an example to illustrate this claim. The objective function to be maximized in hinge GAN is formulated as

$$\mathcal{V}_{\text{Hinge}}(\langle\omega,h\rangle;\mu_\theta) := \mathbb{E}_{x\sim\mu_0}[\min(0,-1+\langle\omega,h(x)\rangle)] + \mathbb{E}_{x\sim\mu_\theta}[\min(0,-1-\langle\omega,h(x)\rangle)]. \quad (14)$$

Given h, the maximizer ω becomes $\hat{d}_h(\mu_0^{\text{tr}},\mu_\theta^{\text{tr}})$, where μ_0^{tr} and μ_θ^{tr} denote truncated distributions whose supports are restricted by conditioning x on $\langle\omega,h(x)\rangle < 1$ and $\langle\omega,h(x)\rangle > -1$, respectively. Since $(\mu_0^{\text{tr}},\mu_\theta^{\text{tr}})$ is generally different from $(\tilde{\mu}_0^{r_{\mathcal{J}}},\tilde{\mu}_\theta^{r_{\mathcal{J}}}) = (\mu_0,\mu_\theta)$, the maximum solution does not satisfy *direction optimality* for $\tilde{\mu}_0^{r_{\mathcal{J}}}$ and $\tilde{\mu}_\theta^{r_{\mathcal{J}}}$.

In line with the above discussion, we propose the following novel maximization problem:

$$\max_{\omega\in\mathbb{S}^{d-1},h\in\mathcal{F}(X)} \mathcal{V}^{\text{SAN}}(\omega,h;\mu_\theta) := \underbrace{\mathcal{V}(\langle\omega^-,h\rangle;\mu_\theta)}_{\mathcal{L}^h(h;\omega,\mu_\theta)} + \lambda\cdot\underbrace{d_{\langle\omega,h^-\rangle}(\tilde{\mu}_0^{r_{\mathcal{J}}\circ f},\tilde{\mu}_\theta^{r_{\mathcal{J}}\circ f})}_{\mathcal{L}^\omega(\omega;h,\mu_\theta)}, \quad (15)$$

where $(\cdot)^-$ indicates a stop-gradient operator. The first and second terms induce *separability* on h and *direction optimality* on ω, respectively. We simply set $\lambda \in \mathbb{R}_+$ to 1 in our experiments. The proposed modification scheme in Eq. (15) enables us to select any maximization objective for h with

Table 2: Minimization problem and weighting function for direction optimization.

	Minimization problem $\mathcal{J}$	Weighting $r_{\mathcal{J}} \circ f(x)$
Wasserstein GAN / Hinge GAN	$-\mathbb{E}_{x\sim\mu_\theta}\left[f(x)\right]$	1
Saturating GAN	$-\mathbb{E}_{x\sim\mu_\theta}\left[\log\varsigma(f(x))\right]$	$1-\varsigma(f(x))$
Non-saturating GAN	$\mathbb{E}_{x\sim\mu_\theta}\left[\log\varsigma(1-f(x))\right]$	$\varsigma(f(x))$

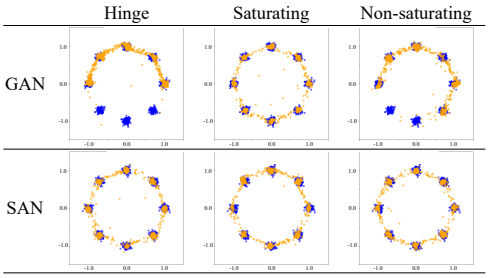


Figure 5: Comparison of the learned distributions (at 10,000 iterations) between GAN and SAN with various objectives. In all cases, SANs cover all modes whereas mode collapse occurs in some GAN cases.

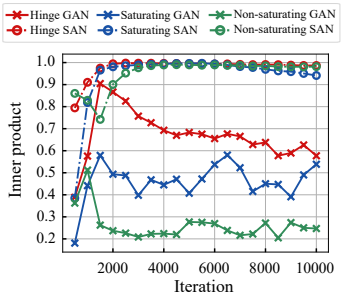


Figure 6: Inner product of trained ω and numerically estimated *optimal direction* for $\tilde{\mu}_0^{r_{\mathcal{J}}}$ and $\tilde{\mu}_\theta^{r_{\mathcal{J}}}$ during training. The trained ω were closer to the *optimal direction* with SAN than with GAN.

$\mathcal{L}^h$, while $\mathcal{L}^\omega$ enforces *direction optimality* on ω. Hence, we can convert most GAN maximization problems into SAN problems by using Eq. (15) (also see Algorithm 1). One can also easily extend SAN to class conditional generation by introducing directions for respective classes (see Appx. C).

A key strength of SAN is that the model can be trained with just two small modifications to existing GAN implementations (see Fig. 4 and Appx. B). The first is to implement the discriminator's final linear layer to be on a hypersphere, i.e., $f_\varphi(x) = w_{\varphi_L}^\top h_{\varphi_{<L}}(x)$ with $w_{\varphi_L} \in \mathbb{S}^{D-1}$, where φ_l is the parameter for the lth layer. The second is to use the maximization problem defined in Eq. (15).

Comparison with other generative models. First, we compare SAN with GAN. In GAN, other than Wasserstein GAN, the optimal discriminator is known to involve a dilemma: it induces a certain dissimilarity but leads to exploding and vanishing gradients (Arjovsky et al., 2017; Lin et al., 2021). In contrast, in SAN, the *metrizable conditions* ensure that the optimal discriminator is not necessary to obtain a *metrizable* discriminator. Next, similarly to variants of the maximum SW (Deshpande et al., 2019; Kolouri et al., 2019), SAN's discriminator is interpreted as extracting features $h(x)$ and slicing them in the most distinctive direction ω as in Fig. 1. In the conventional methods, one-dimensional Wasserstein distances are approximated by sample sorting (see Remark 4.2). It has been reported that the generation performances get better when the batch size increases (Nguyen et al., 2021; Chen et al., 2022), but larger batch sizes lead to slower training speeds, and the size is capped by memory constraints. In contrast, SAN removes the necessity of the approximation.

7 Experiments

We perform experiments with synthetic and image datasets to (1) verify our perspective on GANs as presented in Sec. 5 in terms of *direction optimality*, *separability*, and *injectivity*, and (2) show the effectiveness of SAN against GAN. For fair comparisons, we essentially use the same architectures in SAN and GAN. However, we modify the last linear layer of SAN's discriminators (see Sec. 6).

7.1 Mixture of Gaussians

To empirically investigate *optimal direction*, we conduct experiments on a mixture of Gaussian (MoG). Please refer to Appx. H for empirical verification of the implications of Theorem 5.3 regarding *separability* and *injectivity*. We use a two-dimensional sample space $X = \mathbb{R}^2$. The target MoG on X comprises eight isotropic Gaussians with variances 0.05^2 and means distributed evenly on a circle of radius 1.0. We utilize a 10-dimensional latent space Z to model a generator measure.

Table 3: FID scores (↓) on DCGAN.

Dataset	Hinge loss		Saturating loss		Non-saturating loss	
	GAN	SAN	GAN	SAN	GAN	SAN
CIFAR10	24.07±0.56	**20.23**±0.86	25.63±0.98	**20.62**±0.94	24.90±0.21	**20.51**±0.36
CelebA	32.51±2.53	**27.79**±1.60	37.33±1.02	**28.16**±1.60	28.22±2.16	**27.78**±4.59

Table 4: FID and IS results with the experimental setup of BigGAN (Brock et al., 2019). Scores marked with $*$ are results from our implementation, which is based on the BigGAN authors' PyTorch implementation. For reference, scores reported in their paper are denoted with †.

Metric	CIFAR10			CIFAR100	
	Hinge GAN†	Hinge GAN*	Hinge SAN*	Hinge GAN*	Hinge SAN*
FID (↓)	14.73	8.25±0.82	**6.20**±0.27	10.73±0.16	**8.05**±0.04
IS (↑)	9.22	9.05±0.05	**9.16**±0.08	10.56±0.01	**10.72**±0.05

Table 5: Numerical results for StyleGAN-XL (Sauer et al., 2022) and StyleSAN-XL. Scores marked with † and ‡ are reported in their paper and repository, respectively. Note that our StyleSAN-XL model trained on CIFAR10 is larger in model size than StyleGAN-XL.

Method	CIFAR10		ImageNet (256×256)	
	StyleGAN-XL‡	StyleSAN-XL*	StyleGAN-XL†	StyleSAN-XL*
FID (↓)	(1.85)	**1.36**	2.30	**2.14**
IS (↑)	–	–	265.12	**274.20**

We compare SAN and GAN with various objectives. As shown in Fig. 5, the generator measures trained with SAN cover all modes, whereas mode collapse (Srivastava et al., 2017) occurs with hinge GAN and non-saturating GAN. In addition, Fig. 6 shows a plot of the inner product of the learned direction ω (or the normalized weight in GAN's last linear layer) and the estimated *optimal direction* for $\tilde{\mu}_0^{r_\mathcal{J}}$ and $\tilde{\mu}_\theta^{r_\mathcal{J}}$. Recall that there is no guarantee that a non-optimal direction ω induces a distance.

7.2 Image Generation

We apply the SAN scheme to image generation tasks to show it scales beyond toy experiments.

DCGAN. We train SANs and GANs with various objective functions on CIFAR10 (Krizhevsky et al., 2009) and CelebA (128×128) (Liu et al., 2015). We adopt the DCGAN architectures (Radford et al., 2016), and for the discriminator, we apply spectral normalization (Miyato et al., 2018). As shown in Table 3, SANs outperform GANs in terms of the FID score in all cases.

BigGAN. Next, we apply SAN to BigGAN (Brock et al., 2019) on CIFAR10 and CIFAR100. In this experiment, we calculate the Inception Score (IS) (Salimans et al., 2016), as well as the FID, to evaluate the sample quality by following the experiment in the original paper. As in Table 4, the adoption of SAN consistently improves the generation performance in terms of both metrics.

StyleGAN-XL. Lastly, we apply our SAN training framework to StyleGAN-XL (Sauer et al., 2022), which is a state-of-the-art GAN-based generative model. We name the StyleGAN-XL model combined with our SAN training framework StyleSAN-XL. We train StyleSAN-XL on CIFAR10 and ImageNet (256×256) (Russakovsky et al., 2015), and report the numerical results, as with BigGAN. As shown in Table 5, our models improve the state-of-the-art scores on both cases.

8 Conclusion

We have proposed a unique perspective on GANs to derive sufficient conditions for the discriminator to serve as a distance between the data and generator probability measures. To this end, we introduced the FM* and max-ASW families. By using a class of metrics that are included in both families, we derived the *metrizable conditions* for Wasserstein GAN. We then extended the result to a general GAN. The derived conditions consist of *direction optimality*, *separability*, and *injectivity*.

We then leveraged the theoretical results to develop the Slicing Adversarial Network (SAN), in which a generator and discriminator are trained with a modified GAN training scheme. Despite the ease of modifications and the generality, this model can impose *direction optimality* on the discriminator. Our experiments on synthetic and image datasets showed that SANs outperformed GANs in terms of both the sample quality and mode coverage.

REPRODUCIBILITY STATEMENT

The experimental setups are provided in Appx. F with a detailed description of the network architectures and hyperparameters. We further provide our source code to reproduce our results at `https://github.com/sony/san`. Our implementation for all the image generation tasks is based on open-source repositories. The proofs of all the theoretical claims can be found in Appx. D.

ACKNOWLEDGEMENTS

We sincerely acknowledge the support of Basavaraj Murali, Riya Kuriakose, Srinidhi Srinivasa, and Takuya Narihira, who assisted with our experiments during the rebuttal period. Computational resource of AI Bridging Cloud Infrastructure (ABCI) provided by National Institute of Advanced Industrial Science and Technology (AIST) was used.

REFERENCES

Martin Arjovsky, Soumith Chintala, and Léon Bottou. Wasserstein GAN. *arXiv preprint arXiv:1701.07875*, 2017.

Sanjeev Arora, Rong Ge, Yingyu Liang, Tengyu Ma, and Yi Zhang. Generalization and equilibrium in generative adversarial nets (GANs). In *Proc. International Conference on Machine Learning (ICML)*, volume 70, pp. 224–232, 2017.

Jens Behrmann, Will Grathwohl, Ricky TQ Chen, David Duvenaud, and Jörn-Henrik Jacobsen. Invertible residual networks. In *Proc. International Conference on Machine Learning (ICML)*, pp. 573–582, 2019.

Hugo Berard, Gauthier Gidel, Amjad Almahairi, Pascal Vincent, and Simon Lacoste-Julien. A closer look at the optimization landscapes of generative adversarial networks. In *Proc. International Conference on Learning Representation (ICLR)*, 2020.

Clément Bonet, Paul Berg, Nicolas Courty, François Septier, Lucas Drumetz, and Minh-Tan Pham. Spherical sliced-wasserstein. In *Proc. International Conference on Learning Representation (ICLR)*, 2023.

Nicolas Bonneel, Julien Rabin, Gabriel Peyré, and Hanspeter Pfister. Sliced and radon Wasserstein barycenters of measures. *Journal of Mathematical Imaging and Vision*, 51(1):22–45, 2015.

Andrew Brock, Jeff Donahue, and Karen Simonyan. Large scale gan training for high fidelity natural image synthesis. In *Proc. International Conference on Learning Representation (ICLR)*, 2019.

Robin Chan, Sarina Penquitt, and Hanno Gottschalk. Lu-net: Invertible neural networks based on matrix factorization. *arXiv preprint arXiv:2302.10524*, 2023.

Tatjana Chavdarova, Gauthier Gidel, François Fleuret, and Simon Lacoste-Julien. Reducing noise in gan training with variance reduced extragradient. In *Proc. Advances in Neural Information Processing Systems (NeurIPS)*, volume 32, 2019.

Tatjana Chavdarova, Matteo Pagliardini, Sebastian U Stich, François Fleuret, and Martin Jaggi. Taming gans with lookahead-minmax. In *Proc. International Conference on Learning Representation (ICLR)*, 2022.

Xiongjie Chen, Yongxin Yang, and Yunpeng Li. Augmented sliced wasserstein distances. In *Proc. International Conference on Learning Representation (ICLR)*, 2022.

Casey Chu, Kentaro Minami, and Kenji Fukumizu. Smoothness and stability in GANs. In *Proc. International Conference on Learning Representation (ICLR)*, 2020.

Ishan Deshpande, Yuan-Ting Hu, Ruoyu Sun, Ayis Pyrros, Nasir Siddiqui, Sanmi Koyejo, Zhizhen Zhao, David Forsyth, and Alexander G Schwing. Max-sliced Wasserstein distance and its use for GANs. In *Proc. IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 10648–10656, 2019.

Chris Donahue, Julian McAuley, and Miller Puckette. Adversarial audio synthesis. In *Proc. International Conference on Learning Representation (ICLR)*, 2019.

Jiaojiao Fan, Qinsheng Zhang, Amirhossein Taghvaei, and Yongxin Chen. Variational wasserstein gradient flow. In *Proc. International Conference on Machine Learning (ICML)*, volume 162, pp. 6185–6215, 2022.

Farzan Farnia and Asuman Ozdaglar. Do GANs always have nash equilibria? In *Proc. International Conference on Machine Learning (ICML)*, pp. 3029–3039, 2020.

Tanner Fiez, Chi Jin, Praneeth Netrapalli, and Lillian J Ratliff. Minimax optimization with smooth algorithmic adversaries. In *Proc. International Conference on Learning Representation (ICLR)*, 2022.

Yuan Gao, Yuling Jiao, Yang Wang, Yao Wang, Can Yang, and Shunkang Zhang. Deep generative learning via variational gradient flow. In *Proc. International Conference on Machine Learning (ICML)*, pp. 2093–2101, 2019.

Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Proc. Advances in Neural Information Processing Systems (NeurIPS)*, pp. 2672–2680, 2014.

Ishaan Gulrajani, Faruk Ahmed, Martin Arjovsky, Vincent Dumoulin, and Aaron C Courville. Improved training of Wasserstein GANs. In *Proc. Advances in Neural Information Processing Systems (NeurIPS)*, pp. 5767–5777, 2017.

Zekun Hao, Arun Mallya, Serge Belongie, and Ming-Yu Liu. GANcraft: Unsupervised 3d neural rendering of minecraft worlds. In *Proc. IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 14072–14082, 2021.

Sigurdur Helgason. The Radon transform on Rn. In *Integral Geometry and Radon Transforms*, pp. 1–62. Springer, 2011.

Irina Higgins, Loic Matthey, Arka Pal, Christopher Burgess, Xavier Glorot, Matthew Botvinick, Shakir Mohamed, and Alexander Lerchner. beta-VAE: Learning basic visual concepts with a constrained variational framework. In *Proc. International Conference on Learning Representation (ICLR)*, 2017.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In *Proc. Advances in Neural Information Processing Systems (NeurIPS)*, pp. 6840–6851, 2020.

Ya-Ping Hsieh, Chen Liu, and Volkan Cevher. Finding mixed nash equilibria of generative adversarial networks. In *Proc. International Conference on Machine Learning (ICML)*, pp. 2810–2819, 2019.

Chi Jin, Praneeth Netrapalli, and Michael Jordan. What is local optimality in nonconvex-nonconcave minimax optimization? In *Proc. International Conference on Machine Learning (ICML)*, volume 119, pp. 4880–4889, 2020.

Rie Johnson and Tong Zhang. A framework of composite functional gradient methods for generative adversarial models. *IEEE transactions on pattern analysis and machine intelligence*, 43(1):17–32, 2019.

Michael I Jordan, Zoubin Ghahramani, Tommi S Jaakkola, and Lawrence K Saul. An introduction to variational methods for graphical models. *Machine Learning*, 37(2):183–233, 1999.

Mahdi Karami, Dale Schuurmans, Jascha Sohl-Dickstein, Laurent Dinh, and Daniel Duckworth. Invertible convolutional flow. In *Proc. Advances in Neural Information Processing Systems (NeurIPS)*, volume 32, 2019.

Tero Karras, Timo Aila, Samuli Laine, and Jaakko Lehtinen. Progressive growing of gans for improved quality, stability, and variation. In *Proc. International Conference on Learning Representation (ICLR)*, 2018.

Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. In *Proc. IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4401–4410, 2019.

Tero Karras, Miika Aittala, Samuli Laine, Erik Härkönen, Janne Hellsten, Jaakko Lehtinen, and Timo Aila. Alias-free generative adversarial networks. In *Proc. Advances in Neural Information Processing Systems (NeurIPS)*, 2021.

Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *Proc. International Conference on Learning Representation (ICLR)*, 2015.

Diederik P Kingma and Max Welling. Auto-encoding variational Bayes. In *Proc. International Conference on Learning Representation (ICLR)*, 2014.

Soheil Kolouri, Kimia Nadjahi, Umut Simsekli, Roland Badeau, and Gustavo Rohde. Generalized sliced Wasserstein distances. In *Proc. Advances in Neural Information Processing Systems (NeurIPS)*, volume 32, 2019.

Jungil Kong, Jaehyeon Kim, and Jaekyoung Bae. HiFi-gan: Generative adversarial networks for efficient and high fidelity speech synthesis. In *Proc. Advances in Neural Information Processing Systems (NeurIPS)*, volume 33, pp. 17022–17033, 2020.

Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009.

Kundan Kumar, Rithesh Kumar, Thibault de Boissiere, Lucas Gestin, Wei Zhen Teoh, Jose Sotelo, Alexandre de Brébisson, Yoshua Bengio, and Aaron C Courville. MelGAN: Generative adversarial networks for conditional waveform synthesis. In *Proc. Advances in Neural Information Processing Systems (NeurIPS)*, volume 32, 2019.

Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proc. IEEE*, 86(11):2278–2324, 1998.

Jose Lezama, Wei Chen, and Qiang Qiu. Run-sort-rerun: Escaping batch size limitations in sliced wasserstein generative models. In *Proc. International Conference on Machine Learning (ICML)*, volume 139, pp. 6275–6285, 2021.

Jerry Li, Aleksander Madry, John Peebles, and Ludwig Schmidt. On the limitations of first-order approximation in gan dynamics. In *Proc. International Conference on Machine Learning (ICML)*, pp. 3005–3013, 2018.

Jae Hyun Lim and Jong Chul Ye. Geometric gan. *arXiv preprint arXiv:1705.02894*, 2017.

Zinan Lin, Vyas Sekar, and Giulia Fanti. Why spectral normalization stabilizes GANs: Analysis and improvements. In *Proc. Advances in Neural Information Processing Systems (NeurIPS)*, volume 34, pp. 9625–9638, 2021.

Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Deep learning face attributes in the wild. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3730–3738, 2015.

Lars Mescheder, Sebastian Nowozin, and Andreas Geiger. The numerics of gans. In *Proc. Advances in Neural Information Processing Systems (NeurIPS)*, volume 30, 2017.

Lars Mescheder, Andreas Geiger, and Sebastian Nowozin. Which training methods for GANs do actually converge? In *Proc. International Conference on Machine Learning (ICML)*, pp. 3481–3490, 2018.

Paul Milgrom and Ilya Segal. Envelope theorems for arbitrary choice sets. *Econometrica*, 70(2): 583–601, 2002.

Takeru Miyato and Masanori Koyama. cgans with projection discriminator. In *Proc. International Conference on Learning Representation (ICLR)*, 2018.

Takeru Miyato, Toshiki Kataoka, Masanori Koyama, and Yuichi Yoshida. Spectral normalization for generative adversarial networks. In *Proc. International Conference on Learning Representation (ICLR)*, 2018.

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

Vaishnavh Nagarajan and J Zico Kolter. Gradient descent GAN optimization is locally stable. In *Proc. Advances in Neural Information Processing Systems (NeurIPS)*, volume 30, 2017.

Frank Natterer. *The mathematics of computerized tomography*. SIAM, 2001.

Khai Nguyen and Nhat Ho. Revisiting sliced wasserstein on images: From vectorization to convolution. In *Proc. Advances in Neural Information Processing Systems (NeurIPS)*, volume 35, pp. 17788–17801, 2022.

Khai Nguyen, Nhat Ho, Tung Pham, and Hung Bui. Distributional sliced-Wasserstein and applications to generative modeling. In *Proc. International Conference on Learning Representation (ICLR)*, 2021.

Alexander Quinn Nichol and Prafulla Dhariwal. Improved denoising diffusion probabilistic models. In *Proc. International Conference on Machine Learning (ICML)*, pp. 8162–8171. PMLR, 2021.

Atsushi Nitanda and Taiji Suzuki. Gradient layer: Enhancing the convergence of adversarial training for generative models. In *Proc. International Conference on Artificial Intelligence and Statistics (AISTATS)*, pp. 1008–1016, 2018.

Sebastian Nowozin, Botond Cseke, and Ryota Tomioka. f-GAN: Training generative neural samplers using variational divergence minimization. In *Proc. Advances in Neural Information Processing Systems (NeurIPS)*, volume 29, 2016.

Michael Puthawala, Konik Kothari, Matti Lassas, Ivan Dokmanić, and Maarten De Hoop. Globally injective relu networks. *The Journal of Machine Learning Research*, 23(1):4544–4598, 2022.

Chongli Qin, Yan Wu, Jost Tobias Springenberg, Andrew Brock, Jeff Donahue, Timothy P Lillicrap, and Pushmeet Kohli. Training generative adversarial networks by solving ordinary differential equations. In *Proc. Advances in Neural Information Processing Systems (NeurIPS)*, volume 33, pp. 5599–5609, 2020.

Alec Radford, Luke Metz, and Soumith Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks. In *Proc. International Conference on Learning Representation (ICLR)*, 2016.

Lillian J Ratliff, Samuel A Burden, and S Shankar Sastry. Characterization and computation of local nash equilibria in continuous games. In *Annual Allerton Conference on Communication, Control, and Computing (Allerton)*, pp. 917–924. IEEE, 2013.

Danilo Jimenez Rezende and Shakir Mohamed. Variational inference with normalizing flows. In *Proc. International Conference on Machine Learning (ICML)*, pp. 1530–1538, 2015.

Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *International journal of computer vision*, 115:211–252, 2015.

Tim Salimans, Ian Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford, and Xi Chen. Improved techniques for training GANs. In *Proc. Advances in Neural Information Processing Systems (NeurIPS)*, volume 29, 2016.

Maziar Sanjabi, Jimmy Ba, Meisam Razaviyayn, and Jason D Lee. On the convergence and robustness of training gans with regularized optimal transport. In *Proc. International Conference on Machine Learning (ICML)*, volume 31, 2018.

Axel Sauer, Katja Schwarz, and Andreas Geiger. Stylegan-xl: Scaling stylegan to large diverse datasets. In *ACM SIGGRAPH 2022 conference proceedings*, pp. 1–10, 2022.

Axel Sauer, Tero Karras, Samuli Laine, Andreas Geiger, and Timo Aila. Stylegan-t: Unlocking the power of gans for fast large-scale text-to-image synthesis. In *Proc. International Conference on Machine Learning (ICML)*, 2023.

Samarth Sinha, Anirudh Goyal, Colin Raffel, and Augustus Odena. Top-k training of GANs: Improving generators by making critics less critical. In *Proc. Advances in Neural Information Processing Systems (NeurIPS)*, 2020.

Jiaming Song and Stefano Ermon. Bridging the gap between f-gans and wasserstein gans. In *Proc. International Conference on Machine Learning (ICML)*, pp. 9078–9087, 2020.

Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. In *Proc. International Conference on Learning Representation (ICLR)*, 2020.

Yang Song, Chenlin Meng, and Stefano Ermon. Mintnet: Building invertible neural networks with masked convolutions. In *Proc. Advances in Neural Information Processing Systems (NeurIPS)*, volume 32, 2019.

Akash Srivastava, Lazar Valkov, Chris Russell, Michael U Gutmann, and Charles Sutton. VEEGAN: Reducing mode collapse in gans using implicit variational learning. In *Proc. Advances in Neural Information Processing Systems (NeurIPS)*, volume 30, 2017.

Esteban G Tabak and Cristina V Turner. A family of nonparametric density estimation algorithms. *Communications on Pure and Applied Mathematics*, 66(2):145–164, 2013.

Esteban G Tabak and Eric Vanden-Eijnden. Density estimation by dual ascent of the log-likelihood. *Communications in Mathematical Sciences*, 8(1):217–233, 2010.

Yuhta Takida, Takashi Shibuya, WeiHsiang Liao, Chieh-Hsin Lai, Junki Ohmura, Toshimitsu Uesaka, Naoki Murata, Takahashi Shusuke, Toshiyuki Kumakura, and Yuki Mitsufuji. SQ-VAE: Variational Bayes on discrete representation with self-annealed stochastic quantization. In *Proc. International Conference on Machine Learning (ICML)*, 2022.

Dávid Terjék. Adversarial lipschitz regularization. In *Proc. International Conference on Learning Representation (ICLR)*, 2020.

Sergey Tulyakov, Ming-Yu Liu, Xiaodong Yang, and Jan Kautz. MoCoGAN: Decomposing motion and content for video generation. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1526–1535, 2018.

Cédric Villani. *Optimal transport: old and new*, volume 338. Springer, 2009.

Kun Xu, Chongxuan Li, Jun Zhu, and Bo Zhang. Understanding and stabilizing GANs' training dynamics with control theory. In *Proc. International Conference on Machine Learning (ICML)*, pp. 10566–10575, 2020.

Yasin Yaz, Chuan-Sheng Foo, Stefan Winkler, Kim-Hui Yap, Georgios Piliouras, Vijay Chandrasekhar, et al. The unusual effectiveness of averaging in gan training. In *Proc. International Conference on Learning Representation (ICLR)*, 2019.

Han Zhang, Ian Goodfellow, Dimitris Metaxas, and Augustus Odena. Self-attention generative adversarial networks. In *Proc. International Conference on Machine Learning (ICML)*, pp. 7354–7363, 2019.

Han Zhang, Zizhao Zhang, Augustus Odena, and Honglak Lee. Consistency regularization for generative adversarial networks. In *Proc. International Conference on Learning Representation (ICLR)*, 2020.

Shengjia Zhao, Jiaming Song, and Stefano Ermon. InfoVAE: Balancing learning and inference in variational autoencoders. In *Proc. AAAI Conference on Artificial Intelligence (AAAI)*, pp. 5885–5892, 2019.

Contents

A Notations **15**

B Converting GANs to SANs **15**

C Class Conditional SAN **16**

D Proofs **17**

E Related Work **21**
- E.1 Literature on GAN 21
- E.2 Literature on Sliced Wasserstein Distance 22
- E.3 Limitations 22

F Experimental Setup **23**
- F.1 Mixture of Gaussians 23
- F.2 Image Generation 23

G Supplementary Experiments on BigGAN **26**

H Investigation of Separability and Injectivity **26**
- H.1 Inductions for Injectivity 26
- H.2 Mixture of Gaussians 27
- H.3 Vision Datasets 27

I Investigation of $\mathcal{V}_{\text{GAN}}$ in SW **27**

J Generated Samples **28**

A Notations

We summarize several notations that are used throughout the paper but not introduced in Sec. 2.1 in Table 6.

B Converting GANs to SANs

We can convert GANs to SANs simply by adding two modifications to existing GAN implementations, as shown in Fig. 4. First, we propose to modify the last linear layer of the discriminator as in the following PyTorch-like code snippet. In the code snippet, `feature_network` represents any feature extractor $h(\cdot)$ built with nonlinear layers. On top of it, we propose using the novel maximization objective functions $\mathcal{L}^h$ and $\mathcal{L}^\omega$. The discriminator's outputs, `out_fun` and `out_dir`, are utilized for $\mathcal{L}^h$ and $\mathcal{L}^\omega$, respectively.

```
import torch.nn as nn

class GANDiscriminator(nn.Module):
    def __init__(self,
        dim # dimension of output of "h(x)"
    ):
        super(GANDiscriminator, self).__init__()

        self.main = feature_network(dim) # nonlinear function "h(x)"
        self.fc = nn.Linear(dim, 1) # last linear layer "w"

    def forward(self,
        x # batch of real/fake samples
    ):
        feature = self.main(x)
        out = self.fc(feature)
```

```

        return out
```

Listing 1: GAN discriminator

```
import torch
import torch.nn as nn
import torch.nn.functional as F

class SANDiscriminator(nn.Module):
    def __init__(self,
        dim # dimension of output of "h(x)"
    ):
        super(SANDiscriminator, self).__init__()

        self.main = feature_network(dim) # nonlinear function "h(x)"
        vec_init = torch.randn(dim, ) # initialization for direction
        self.vec = nn.Parameter(vec_init) # unnormalized direction

    def forward(self,
        x # batch of real/fake samples
    ):
        feature = self.main(x)
        direction = F.normalize(self.vec, dim=0) # direction "omega"
        out_fun = feature @ direction.detach()
        out_dir = feature.detach() @ direction

        return out_fun, out_dir
```

Listing 2: SAN discriminator

As an example, converting the maximization objective of Hinge GAN to SAN leads to

$$\mathcal{V}_{\text{Hinge}}^{\text{SAN}}(\omega, h; \mu_\theta) = \underbrace{\mathcal{V}_{\text{Hinge}}(\langle \omega^-, h\rangle, \mu_\theta)}_{\mathcal{L}^h} + \lambda \cdot \underbrace{\mathcal{V}_{\text{W}}(\langle \omega, h^-\rangle, \mu_\theta)}_{\mathcal{L}^\omega}. \tag{16}$$

For general GANs, the approximations of $\tilde{\mu}_0^{r_\mathcal{J} \circ f}$ and $\tilde{\mu}_\theta^{r_\mathcal{J} \circ f}$ are needed. The objective term $\mathcal{L}^\omega$ can be formulated as

$$\mathcal{L}^\omega(\omega; h, \mu_\theta) = \frac{1}{Z_0}\mathbb{E}_{x\sim\mu_0}[r_\mathcal{J} \circ f^-(x)\langle \omega, h(x)\rangle] - \frac{1}{Z_\theta}\mathbb{E}_{x\sim\mu_\theta}[r_\mathcal{J} \circ f^-(x)\langle \omega, h(x)\rangle], \tag{17}$$

where $Z_0 = \mathbb{E}_{x\sim\mu_0}[r_\mathcal{J} \circ f^-(x)]$ and $Z_\theta = \mathbb{E}_{x\sim\mu_\theta}[r_\mathcal{J} \circ f^-(x)]$. The normalizing terms Z_0 and Z_θ can be approximated per batch. However, we found it beneficial to use the following auxiliary objective function instead:

$$\mathcal{L}^\omega_{\text{simple}}(\omega; h, \mu_\theta) = \mathbb{E}_{x\sim\mu_0}[R_\mathcal{J}(\langle \omega, h\rangle)] - \mathbb{E}_{x\sim\mu_\theta}[R_\mathcal{J}(\langle \omega, h\rangle)], \tag{18}$$

which ignores the normalization terms Z_0 and Z_θ in Eq. (17). The psuedo code can found in Algorithm1

C Class Conditional SAN

SAN can easily be extended to conditional generation. Let C be the number of classes. For a conditional case, we adopt the usual generator parameterization, $g_\theta : Z \times \{c\}_{c=1}^C \to X$, which gives a conditional generator distribution $\mu_\theta^c := g_\theta(\cdot, c)_\sharp\sigma$. For the discriminator, we prepare trainable directions for the respective classes as $\Omega := \{\omega_c\}_{c=1}^C$. Under this setup, we define the minimax problem of conditional SAN as follows:

$$\max_{\Omega, h} \mathbb{E}_{c\sim P(c)}\left[\mathcal{V}_{\mu_0^c}^{\text{SAN}}(\omega_c, h; \mu_\theta^c)\right] \quad \text{and} \quad \min_\theta \mathbb{E}_{c\sim P(c)}\left[\mathcal{J}(\theta; \langle \omega_c, h\rangle)\right], \tag{19}$$

where $P(c)$ is a pre-fix probability mass function for c, and μ_0^c is the target probability measure conditioned on c.

Table 6: Several notations used throughout the main paper.

Sliced optimal transport	
$\mathcal{R}I_\mu(\cdot,\omega)$	The RT of a probability density function I with a direction $\omega \in \mathbb{S}^{D-1}$ given.
$\mathcal{S}^h I_\mu(\cdot,\omega)$	The SRT of a probability density function I with a direction $\omega \in \mathbb{S}^{D-1}$ and function $h : X \to \mathbb{R}^D$ given.
$F_\mu^{h,\omega}(\cdot)$	The cumulative distribution function of $\mathcal{S}^h I_\mu(\cdot,\omega)$.
Functional spaces	
$\mathcal{F}(X)$	Any functional space included in $L^\infty(X,\mathbb{R}^D)$.
$\mathcal{F}_{\mathrm{S}(\mu,\nu)}(X)$	The class of all the separable functions for $\mu,\nu \in \mathcal{P}(X)$, which is often simply denoted as $\mathcal{F}_{\mathrm{S}}$.
$\mathcal{F}_{\mathrm{I}}(X)$	The class of all the injectivity functions in $L^\infty(X,\mathbb{R}^D)$, which is often simply denoted as $\mathcal{F}_{\mathrm{I}}$.
$\mathcal{F}_{\mathrm{I,S}}(X)$	The intersection set of $\mathcal{F}_{\mathrm{I}}$ and $\mathcal{F}_{\mathrm{S}}$, i.e., $\mathcal{F}_{\mathrm{I,S}} = \mathcal{F}_{\mathrm{I}} \cap \mathcal{F}_{\mathrm{S}}$.
Divergences and distances	
$\mathscr{D}_D^{\mathrm{FM}}$	The family of FM with $D \in \mathbb{N}$ given. Its instance is denoted as $FM_{\mathcal{F}}(\mu,\nu) \in \mathscr{D}_D^{\mathrm{FM}}$.
$\mathscr{D}_{\mathcal{F}}^{\mathrm{FM}^*}$	The family of FM* with $\mathcal{F}$ given. Its instance is denoted as $FM_h^*(\mu,\nu) \in \mathscr{D}_{\mathcal{F}}^{\mathrm{FM}^*}$ with $h \in \mathcal{F}$.
$\mathscr{D}_{\mathcal{F}}^{\mathrm{mA}}$	The family of max-ASW with $\mathcal{F}$ given. Its instance is denoted as $max\text{-}ASW_h(\mu,\nu) \in \mathscr{D}_{\mathcal{F}}^{\mathrm{mA}}$ with $h \in \mathcal{F}$.

Algorithm 1 Training SAN (the blue lines indicate modified steps against GAN training)

Require: Consider a sample space X and a latent space Z. A target distribution $\mu_0 \in \mathcal{P}(X)$ and a base distribution $\sigma \in \mathcal{P}(Z)$ are given. Initialize a parameter θ modeling a generator function as $g_\theta : Z \to X$, and discriminator parameters, φ and $\omega \in \mathbb{S}^{D-1}$, modeling a discriminator function as $\langle \omega, h_\varphi \rangle$, where $h_\varphi : X \to \mathbb{R}^D$. Step sizes for the trainable parameters are set as η_θ, η_φ, and η_ω. Original minimization and maximization problems for GANs, denoted as $\mathcal{J}_{\mathrm{GAN}}$ and $\mathcal{V}_{\mathrm{GAN}}$, are given.

while not converged **do**
  /* Discriminator update */
  for certain number of steps **do**
    Sample a minibatch of data $\{x_i\}_{i=1}^M \sim \mu_0$ (i.i.d.).
    Get generated samples $\{y_i\}_{i=1}^N$, where $y_i = g_\theta(z_i)$ with $\{z_i\}_{i=1}^N \sim \sigma$ (i.i.d.).
    Compute $\hat{\mathcal{L}}^h$ as the Monte-Carlo estimate of $\mathcal{V}_{\mathrm{GAN}}(\langle \omega, h \rangle; \theta)$ with $\{x_i\}_{i=1}^M$ and $\{y_i\}_{i=1}^N$.
    Compute $\hat{\mathcal{L}}^\omega$ as the Monte-Carlo estimate of Eq. (18) with $\{x_i\}_{i=1}^M$ and $\{y_i\}_{i=1}^N$.
    Update the parameters as $\omega \leftarrow \mathrm{Proj}_{\mathbb{S}^{D-1}}(\omega + \nabla_\omega \hat{\mathcal{L}}^\omega)$ and $\varphi \leftarrow \varphi + \eta_\varphi \nabla_\varphi \hat{\mathcal{L}}^\varphi$.
  end for
  /* Generator update */
  for certain number of steps **do**
    Get generated samples $\{y_i\}_{i=1}^N$, where $y_i = g_\theta(z_i)$ with $\{z_i\}_{i=1}^N \sim \sigma$ (i.i.d.).
    Compute $\tilde{\mathcal{J}}$ as the Monte-Carlo estimate of $\mathcal{J}_{\mathrm{GAN}}(\theta, \langle \omega, h \rangle)$ with $\{y_i\}_{i=1}^N$.
    Update the parameters as $\theta \leftarrow \theta - \eta_\theta \nabla_\theta \tilde{\mathcal{J}}$.
  end for
end while

GAN can easily be converted into SAN even for conditional generation, as many previous works have implemented a conditional discriminator with the class-conditional last layer as $w_{\phi_L,c}^\top h_{\phi_{<L}}(x)$ (Miyato & Koyama, 2018; Miyato et al., 2018; Brock et al., 2019; Zhang et al., 2019).

D PROOFS

Proposition 3.2. *For $\mathcal{F}(X) \in L^\infty(X,\mathbb{R})$, $IPM_{\mathcal{F}}(\cdot,\cdot) := \max_{f\in\mathcal{F}} d_f(\cdot,\cdot) \in \mathscr{D}_1^{FM}$.*

Proof. The claim is a direct consequence of the definitions of the IPM and the FM. □

We introduce a lemma, which will come in handy for the proofs of Proposition 3.5 and Lemma 4.3.

Lemma D.1. *Given a function $h \in L^\infty(X, \mathbb{R}^D)$ and probability measures $\mu, \nu \in \mathcal{P}(X)$, we have*

$$\|d_h(\mu,\nu)\|_2 = \max_{\omega \in \mathbb{S}^{D-1}} d_{\langle \omega, h\rangle}(\mu,\nu). \tag{20}$$

Proof. From the Cauchy–Schwarz inequality, we have the following lower bound using $\omega \in \mathbb{S}^{D-1}$:

$$\begin{aligned}\|d_h(\mu,\nu)\|_2 &= \|\mathbb{E}_{x\sim\mu}[h(x)] - \mathbb{E}_{x\sim\nu}[h(x)]\|_2 \\ &\geq \langle \omega, \mathbb{E}_{x\sim\mu}[h(x)] - \mathbb{E}_{x\sim\nu}[h(x)]\rangle,\end{aligned} \tag{21}$$

where the equality holds if and only if $\omega = \hat{d}(\mu,\nu)$. Recall that $\hat{(\cdot)}$ denotes a normalization operator. Then, calculating the norm $\|d_h(\mu,\nu)\|_2$ is formulated as the following maximization problem:

$$\|d_h(\mu,\nu)\|_2 = \max_{\omega \in \mathbb{S}^{D-1}} \mathbb{E}_{x\sim\mu}[\langle \omega, h(x)\rangle] - \mathbb{E}_{x\sim\nu}[\langle \omega, h(x)\rangle] \tag{22}$$

$$= \max_{\omega \in \mathbb{S}^{D-1}} d_{\langle \omega, h\rangle}(\mu,\nu). \tag{23}$$

Note that we can interchange the inner product and the expectation operator by the independency of ω w.r.t. x. □

Proposition 3.5. *Let ω be on $\mathbb{S}^{D-1}$. For any $h \in L^\infty(X, \mathbb{R}^D)$, minimization of $FM_h^*(\mu_\theta, \mu_0)$ is equivalent to optimization of $\min_{\theta\in\mathbb{R}^{D_\theta}} \max_{\omega\in\mathbb{S}^{D-1}} \mathcal{J}_W(\theta; \langle\omega, h\rangle)$. Thus,*

$$\nabla_\theta FM_h^*(\mu_\theta, \mu_0) = \nabla_\theta \mathcal{J}_W(\theta; \langle \omega^*, h\rangle), \tag{24}$$

where ω^* is the optimal solution (direction) given as follows:

$$\omega^* = \arg\max_{\omega \in \mathbb{S}^{D-1}} d_{\langle \omega, h\rangle}(\mu_0, \mu_\theta). \tag{25}$$

Proof. From Lemma D.1, we have

$$\|d_h(\mu_0,\mu_\theta)\|_2 = \max_{\omega \in \mathbb{S}^{D-1}} d_{\langle \omega, h\rangle}(\mu_0,\mu_\theta) \tag{26}$$

$$= \max_{\omega \in \mathbb{S}^{D-1}} \mathbb{E}_{x\sim\mu_0}[\langle \omega, h(x)\rangle] - \mathbb{E}_{x\sim\mu_\theta}[\langle \omega, h(x)\rangle]. \tag{27}$$

By a simple envelope theorem (Milgrom & Segal, 2002), we obtain

$$\nabla_\theta \|d_h(\mu_0,\mu_\theta)\|_2 = \nabla_\theta \left(\mathbb{E}_{x\sim\mu_0}[\langle \omega^*, h(x)\rangle] - \mathbb{E}_{x\sim\mu_\theta}[\langle \omega^*, h(x)\rangle]\right), \tag{28}$$

where $\omega^* = \max_{\omega\in\mathbb{S}^{D-1}} d_{\langle\omega,h\rangle}(\mu_0,\mu_\theta)$. Since the first term in the right-hand side of Eq. (28) does not depend on θ, taking the gradient of $\|d_h(\mu_0,\mu_\theta)\|_2$ w.r.t. θ becomes

$$\nabla_\theta \|d_h(\mu_0,\mu_\theta)\|_2 = -\nabla_\theta \mathbb{E}_{x\sim\mu_\theta}[\langle \omega^*, h(x)\rangle]. \tag{29}$$

Substituting the definition of $\mathcal{J}_{\mathrm{W}}$ and $FM_h^*(\cdot,\cdot)$ into Eq. (29) leads to the equality in the statement of Proposition 3.5. □

Lemma 4.3. *Given $\mu, \nu \in \mathcal{P}(X)$, every $h \in \mathcal{F}_{S(\mu,\nu)}(X)$ satisfies $FM_h^*(\mu,\nu) \in \mathscr{D}^{mA}_{\mathcal{F}_S}$.*

Proof. From Lemma D.1, we have

$$\|d_h(\mu,\nu)\|_2 = \max_{\omega \in \mathbb{R}^{D-1}} \mathbb{E}_{x\sim\mu}[\langle \omega, h(x)\rangle] - \mathbb{E}_{x\sim\nu}[\langle \omega, h(x)\rangle] \tag{30}$$

$$= \mathbb{E}_{x\sim\mu}[\langle \omega^*, h(x)\rangle] - \mathbb{E}_{x\sim\nu}[\langle \omega^*, h(x)\rangle], \tag{31}$$

where the maximizer is $\omega^* = \hat{d}_h(\mu,\nu)$.

The expected values are reformulated by the SRT as

$$\mathbb{E}_{x\sim\mu}[\langle \omega, h(x)\rangle] = \int_X \left(\int_{\mathbb{R}} \xi\delta(\xi - \langle \omega, h(x)\rangle) d\xi\right) I_\mu(x) dx \tag{32}$$

$$= \int_{\mathbb{R}} \xi \left(\int_X I_\mu(x)\delta(\xi - \langle \omega, h(x)\rangle) dx\right) d\xi \tag{33}$$

$$= \mathbb{E}_{\xi\sim\mathcal{S}^h I_\mu(\xi,\omega)}[\xi]. \tag{34}$$

The interchange of the integral follows from Fubini's theorem. Here, the applicability of the theorem is justified by the absolute integrability of the integrant function. That is, $\int_X \|h(x)\|_2 d\mu(x) \leq \|h\|_{L^\infty(X,\mathbb{R}^D)}\mu(X) = \|h\|_{L^\infty(X,\mathbb{R}^D)} < \infty$.

Further, the right-hand side of Eq. (34) follows the representation of an expectation by an integral of quantiles:

$$\mathbb{E}_{\xi\sim\mathcal{S}^h I_\mu(\xi,\omega)}[\xi] = \int_0^1 Q_\mu^{h,\omega}(\rho)d\rho \tag{35}$$

By substituting Eqs. (34) and (35) into Eq. (31), we obtain

$$\|d_h(\mu,\nu)\|_2 = \int_0^1 \left(Q_\mu^{h,\omega^*}(\rho) - Q_\nu^{h,\omega^*}(\rho)\right) d\rho. \tag{36}$$

The right-hand side of Eq. (36) is bounded above:

$$\|d_h(\mu,\nu)\|_2 \leq \int_0^1 \left|Q_\mu^{h,\omega^*}(\rho) - Q_\nu^{h,\omega^*}(\rho)\right| d\rho. \tag{37}$$

The equality holds if and only if $Q_\mu^{h,\omega^*}(\rho) - Q_\nu^{h,\omega^*}(\rho) \geq 0$ for all ρ or $Q_\mu^{h,\omega^*}(\rho) - Q_\nu^{h,\omega^*}(\rho) \leq 0$ for all ρ. By considering the fact that selecting $-\omega$ changes the sign and Eq. (36) is non-negative, the necessary and sufficient condition for the equality becomes $Q_\mu^{h,\omega^*}(\rho) - Q_\nu^{h,\omega^*}(\rho) \geq 0$ for $\rho \in [0,1]$. From the monotonicity of quantile functions, the condition for the equality is written as $F_\mu^{h,\omega}(\rho) \leq F_\nu^{h,\omega}(\rho)$. □

We here borrow a lemma from Chen et al. (2022), which is used in the proof of Lemma 4.4.

Lemma D.2 (Chen et al. (2022), Lemma 1)**.** *Given an injective function $h : X \to \mathbb{R}^D$ and two probability measures $\mu, \nu \in \mathcal{P}(X)$, for all $\xi \in \mathbb{R}$ and $\omega \in \mathbb{S}^{D-1}$, $\mathcal{S}^h I_\mu(\xi,\omega) = \mathcal{S}^h I_\nu(\xi,\omega)$ if and only if $\mu = \nu$.*

Lemma 4.4. *Every max-ASW$_h(\cdot,\cdot) \in \mathscr{D}_{\mathcal{F}_I}^{mA}$ is a distance, where $\mathcal{F}_I$ indicates a class of injective functions in $L^\infty(X,\mathbb{R}^D)$.*

Proof. One can prove this claim by following similar procedures to the proof of Proposition 1 in Kolouri et al. (2019).

First, the non-negativity and symmetry properties are immediately proven by the fact that the Wasserstein distance is indeed a distance satisfying non-negatibity and symmetry.

Next, let ω^* denote

$$\omega^* = \arg\max_{\omega\in\mathbb{S}^{D-1}} W_1(\mathcal{S}^h I_{\mu_1}(\cdot,\omega), \mathcal{S}^h I_{\mu_2}(\cdot,\omega)). \tag{38}$$

Pick $\nu \in \mathcal{P}(X)$ arbitrarily. The triangle inequality with *max-ASW*$_h(\cdot,\cdot)$ is satisfied since

$$\begin{aligned}
\textit{max-ASW}_h(\mu_1,\mu_2) &= \max_{\omega\in\mathbb{S}^{D-1}} W_1(\mathcal{S}^h I_{\mu_1}(\cdot,\omega), \mathcal{S}^h I_{\mu_2}(\cdot,\omega)) \\
&= W_1(\mathcal{S}^h I_{\mu_1}(\cdot,\omega^*), \mathcal{S}^h I_{\mu_2}(\cdot,\omega^*)) \\
&\leq W_1(\mathcal{S}^h I_{\mu_1}(\cdot,\omega^*), \mathcal{S}^h I_{\nu}(\cdot,\omega^*)) + W_1(\mathcal{S}^h I_{\nu}(\cdot,\omega^*), \mathcal{S}^h I_{\mu_2}(\cdot,\omega^*)) \qquad (39) \\
&\leq \max_{\omega\in\mathbb{R}^{D-1}} W_1(\mathcal{S}^h I_{\mu_1}(\cdot,\omega), \mathcal{S}^h I_{\nu}(\cdot,\omega^*)) \\
&\quad + \max_{\omega\in\mathbb{R}^{D-1}} W_1(\mathcal{S}^h I_{\nu}(\cdot,\omega^*), \mathcal{S}^h I_{\mu_2}(\cdot,\omega)) \qquad (40) \\
&= \textit{max-ASW}_h(\mu_1,\nu) + \textit{max-ASW}_h(\nu,\mu_2). \qquad (41)
\end{aligned}$$

We next show the identity of indiscernibles. For any μ, from $W_1(\mu,\mu) = 0$, we have $W_1(\mathcal{S}^h I_\mu(\cdot,\omega), \mathcal{S}^h I_\mu(\cdot,\omega)) = 0$; therefore, *max-ASW*$_h(\mu,\mu) = 0$. Conversely, *max-ASW*$_h(\mu,\nu) = 0$ is equivalent to $\mathcal{S}^h I_\mu(\cdot,\omega) = \mathcal{S}^h I_\nu(\cdot,\omega)$ for any $\omega \in \mathbb{S}^{D-1}$. This holds if and only if $\mu = \nu$ from the injectivity assumption on h and Lemma D.2. Therefore, *max-ASW*$_h(\mu,\nu) = 0$ holds if and only if $\mu = \nu$. □

Proposition 4.5. *Given $\mu, \nu \in \mathcal{P}(X)$, every $FM_h^*(\mu,\nu) \in \mathscr{D}^{FM^*}_{\mathcal{F}_{I,S}}$ is indeed a distance, where $\mathcal{F}_{I,S}$ indicates $\mathcal{F}_I \cap \mathcal{F}_S$.*

Proof. The claim is a direct consequence of the results of Lemmas 4.3 and 4.4.

From $\mathcal{F}_{\mathrm{I,S}} := \mathcal{F}_{\mathrm{I}} \cap \mathcal{F}_{\mathrm{S}} \subset \mathcal{F}_{\mathrm{S}}$ and Lemma 4.3, $\mathcal{D}(\mu,\nu) \in \mathscr{D}^{\mathrm{mA}}_{\mathcal{F}_{\mathrm{I,S}}}$ holds for any $\mathcal{D}(\mu,\nu) \in \mathscr{D}^{\mathrm{FM}^*}_{\mathcal{F}_{\mathrm{I,S}}}$. At the same time, from $\mathcal{F}_{\mathrm{I,S}} \subset \mathcal{F}_{\mathrm{I}}$ and Lemma 4.4, $\mathcal{D}(\mu,\nu) \in \mathscr{D}^{\mathrm{mA}}_{\mathcal{F}_{\mathrm{I,S}}}$ is a distance. Thus, the claim is proven. □

Lemma 5.1. *Given $h \in \mathcal{F}_I \cap \mathcal{F}_S$ with $\mathcal{F}_I, \mathcal{F}_S \subseteq L^\infty(X, \mathbb{R}^D)$, let $\omega^* \in \mathbb{S}^{D-1}$ be $\hat{d}_h(\mu_0,\mu_\theta)$. Then $f(x) = \langle \omega, h(x)\rangle$ is $(\mathcal{J}_W, FM_h^*)$-metrizable.*

Proof. From Proposition 4.5, given that h is injective and separable for μ_0 and μ_θ, $FM_h^*(\mu_0,\mu_\theta)$ is a distance between μ_0 and μ_θ.

From Proposition 3.5, minimizing $FM_h^*(\mu_0,\mu_\theta)$ w.r.t. θ is equivalent to optimizing the following:

$$\min_\theta \mathcal{J}_{\mathrm{W}}(\theta, \langle \omega^*, h\rangle), \tag{42}$$

where ω^* is the maximizer of $\hat{d}_{\langle\omega,h\rangle}(\mu_0,\mu_\theta)$, i.e., $\omega^* = d_h(\mu_0,\mu_\theta)$. Now, $FM_h^*(\mu_0,\mu_\theta)$ is distance, so $x \mapsto \langle \omega^*, h(x)\rangle$ is $(\mathcal{J}_{\mathrm{W}}, FM_h^*)$-*metrizable* for μ_0 and μ_θ. □

Lemma 5.2. *For any $\rho : X \to \mathbb{R}_+$ and a distance for probability measures $\mathcal{D}(\cdot,\cdot)$, $\mathcal{D}(\tilde{\mu}^\rho, \tilde{\nu}^\rho)$ indicates a distance between μ and ν.*

Proof. For notational simplicity, we denote $\mathcal{D}(\tilde{\mu}^\rho, \tilde{\nu}^\rho)$ as $\tilde{\mathcal{D}}_\rho(\mu,\nu)$. First, the non-negativity and symmetry properties are immediately proven by the assumption that $\mathcal{D}(\cdot,\cdot)$ is a distance.

The triangle inequality regarding $\tilde{\mathcal{D}}_\rho(\cdot,\cdot)$ is satisfied since the following inequality holds for any $\mu_1, \mu_2, \mu_3 \in \mathcal{P}(X)$:

$$\begin{aligned}\tilde{\mathcal{D}}_\rho(\mu_1,\mu_2) &= \mathcal{D}(\tilde{\mu}_1^\rho, \tilde{\mu}_2^\rho)\\ &\le \mathcal{D}(\tilde{\mu}_1^\rho, \tilde{\mu}_3^\rho) + \mathcal{D}(\tilde{\mu}_3^\rho, \tilde{\mu}_2^\rho)\end{aligned} \tag{43}$$

where $I_{\tilde{\mu}_i^\rho}(x) \propto \rho(x) I_{\mu_i}(x)$ for $i = 1,2,3$. Now, from the positivity assumption on ρ, $I_\mu \mapsto I_{\tilde{\mu}^\rho}$ is invertible. In particular, one can recover μ_i from $\tilde{\mu}_i^\rho$ uniquely by

$$I_{\mu_i}(x) = \frac{I_{\tilde{\mu}_i^\rho}(x)}{\rho(x)\int_X \rho(x)^{-1} d\mu_i(x)}. \tag{44}$$

Thus, with the μ_i given by Eq. (44), we have

$$\tilde{\mathcal{D}}_\rho(\mu_i,\mu_3) = \mathcal{D}(\tilde{\mu}_i^\rho, \tilde{\mu}_3^\rho) \quad (i = 1,2). \tag{45}$$

From Eqs. (43) and (45), we have

$$\tilde{\mathcal{D}}_\rho(\mu_1,\mu_2) \le \tilde{\mathcal{D}}_\rho(\mu_1,\mu_3) + \tilde{\mathcal{D}}_\rho(\mu_3,\mu_2). \tag{46}$$

Obviously, $\tilde{\mathcal{D}}_\rho(\mu,\mu) = 0$. Now, $\tilde{\mathcal{D}}_\rho(\mu,\nu) = 0$ is equivalent to $\tilde{\mu}^\rho = \tilde{\nu}^\rho$. Here, from the invertibility of $I_\mu \mapsto I_{\tilde{\mu}^\rho}$ again, $\tilde{\mu}^\rho = \tilde{\nu}^\rho$ if and only if $\mu = \nu$. Thus, $\tilde{\mathcal{D}}_\rho(\mu,\nu) = 0 \iff \mu = \nu$. □

Theorem 5.3. *Given a functional $\mathcal{J}(\theta; f) := \mathbb{E}_{x\sim\mu_\theta}[R \circ f(x)]$ with $R : \mathbb{R} \to \mathbb{R}$ whose derivative, denoted as R', is positive, let $h \in L^\infty(X, \mathbb{R}^D)$ and $\omega \in \mathbb{S}^{D-1}$ satisfy the following conditions. Then, $f(x) = \langle \omega, h(x)\rangle$ is $\mathcal{J}$-metrizable for μ_θ and μ_0.*

- ***(Direction optimality)*** $\omega = \arg\max_{\tilde{\omega}\in\mathbb{S}^{D-1}} d_{\langle\tilde{\omega},h\rangle}(\tilde{\mu}_0^{R'\circ f}, \tilde{\mu}_\theta^{R'\circ f})$,
- ***(Separability)*** *h is separable for $\tilde{\mu}_0^{R'\circ f}$ and $\tilde{\mu}_\theta^{R'\circ f}$,*
- ***(Injectivity)*** *h is an injective function.*

Then, $x \mapsto \langle \omega, h(x) \rangle$ *is metrizable with* $\mathcal{J}$ *for* μ_θ *and* μ_0.

Proof. For notational simplicity, we omit the superscripts of $\tilde{\mu}_0^{R' \circ f}$ and $\tilde{\mu}_\theta^{R' \circ f}$ in this proof. From Lemma 5.2, if $FM_h^*(\tilde{\mu}_0, \tilde{\mu}_\theta)$ is a distance between $\tilde{\mu}_0$ and $\tilde{\mu}_\theta$, then it is also a distance between μ_0 and μ_θ.

Following the same procedures as in Lemma 5.1 leads to the claim.

From Proposition 4.5, given that h is injective and separable for $\tilde{\mu}_0$ and $\tilde{\mu}_\theta$, $FM_h^*(\tilde{\mu}_0, \tilde{\mu}_\theta)$ is a distance between $\tilde{\mu}_0$ and $\tilde{\mu}_\theta$.

From Proposition 3.5, minimizing $FM_h^*(\tilde{\mu}_0, \tilde{\mu}_\theta)$ w.r.t. θ is equivalent to optimizing the following:

$$\min_\theta \tilde{\mathcal{J}}_{\mathrm{W}}(\theta, \langle \omega^*, h \rangle), \tag{47}$$

where $\tilde{\mathcal{J}}_{\mathrm{W}}(\theta, \langle \omega^*, h \rangle) = -\mathbb{E}_{x \sim \tilde{\mu}_\theta}[\langle \omega^*, h(x) \rangle]$ and $\omega^* = \arg\max_{\tilde{\omega} \in \mathbb{S}^{D-1}} d_{\langle \tilde{\omega}, h \rangle}(\tilde{\mu}_0^{R' \circ f}, \tilde{\mu}_\theta^{R' \circ f})$. Now that $\omega = \omega^*$ from the direction optimality assumption, the gradient of the minimization objective in Eq. (47) w.r.t. θ becomes

$$\nabla_\theta \tilde{\mathcal{J}}_{\mathrm{W}}(\theta, \langle \omega^*, h \rangle) = -\nabla_\theta \mathbb{E}_{x \sim \tilde{\mu}_\theta}[\langle \omega, h(x) \rangle] \tag{48}$$

$$= -\frac{1}{Z} \mathbb{E}_{z \sim \sigma}[R' \circ f(g_\theta(z)) \nabla_\theta \langle \omega, h(g_\theta(z)) \rangle] \tag{49}$$

$$= \frac{1}{Z} \nabla_\theta \mathcal{J}(\theta, \langle \omega, h(g_\theta(z)) \rangle), \tag{50}$$

where we used an envelope theorem (Milgrom & Segal, 2002) to get Eq. (48) and $Z := \mathbb{E}_{x \sim \mu_\theta}[r \circ f(x)]$ is a normalizing constant.

Now, $FM_h^*(\tilde{\mu}_0, \tilde{\mu}_\theta)$ is distance and the gradient of $FM_h^*(\tilde{\mu}_0, \tilde{\mu}_\theta)$ is equivalent to $\nabla_\theta \mathcal{J}(\theta, \langle \omega, h(g_\theta(z)) \rangle)$ except for the scaling factor Z, so $x \mapsto \langle \omega, h(x) \rangle$ is $\mathcal{J}$*-metrizable*. □

E Related Work

In this section, we present the related work, some of which are described in Secs. 1, 2, and 6, while mentioning the limitations of the current work.

E.1 Literature on GAN

Goodfellow et al. (2014) was responsible for pioneering work that developed GAN with various theoretical analyses under the optimal discriminator (maximizer) assumption. Since then, GAN has been applied to a variety of tasks, yet the instability of its learning has always been a challenge. Arjovsky et al. (2017) elaborated on one of the causes by showing that the maximizers of f-GANs lead to exploding and vanishing gradients with disjoint supports of the generator and data distributions. Although they proposed Wasserstein GAN, which avoids the gradient issue, it suffers from another instability issue (Chu et al., 2020; Xu et al., 2020; Qin et al., 2020). We empirically observe that Wasserstein GAN tends not to satisfy *separability*, which gives us another explanation for the instability. Several techniques to improve the GAN training for general GANs have been proposed, such as regularization terms (Nagarajan & Kolter, 2017; Terjék, 2020; Zhang et al., 2020), guidance losses (Sauer et al., 2022; 2023), and optimization schemes (Yaz et al., 2019; Chavdarova et al., 2019; Qin et al., 2020; Chavdarova et al., 2022). It should be noted that the above techniques are orthogonal to our proposed method imposing *direction optimality*, which means SAN has potential to enhance many existing GANs. Our experiment on StyleGAN-XL, which adopts many of the techniques to achieve the state-of-the-art performance in large-scale generation tasks (Sauer et al., 2022), supports this expectation (see Sec. 7.2).

Several papers have focused on a theoretical understanding of GAN optimization. The discriminator optimality is a convenient assumption for theoretical analysis (Goodfellow et al., 2014; Johnson & Zhang, 2019; Gao et al., 2019; Fan et al., 2022), although achieving such optimality in practical cases is rarely feasible (Li et al., 2018; Chu et al., 2020). Some studies shed light on the analysis of GAN optimization from the perspective of the local stability and convergence behavior of gradient

descent-ascent (Nagarajan & Kolter, 2017; Mescheder et al., 2018). In another direction, researches have introduced and characterized variants of equilibrium notions such as the local equilibria (Ratliff et al., 2013; Jin et al., 2020), proximal equilibria (Farnia & Ozdaglar, 2020), and mixed Nash equilibria (Arora et al., 2017; Hsieh et al., 2019). Optimization smoothness, a useful perspective for looking into stability of gradient-based learning dynamics, is beneficial even for GAN cases under certain assumptions (Nitanda & Suzuki, 2018; Chu et al., 2020; Fiez et al., 2022). In our work, we analyze GAN optimization in terms of the concept of *metrizability* (see Definition 1.1), which does not fall into the above taxonomy.

E.2 Literature on Sliced Wasserstein Distance

SW was initially proposed to mitigate the course of dimension (Bonneel et al., 2015). The calculation of SW involves Radon transform and direction sampling (ω). Kolouri et al. (2019) first extended SW to the generalized SW in the context of deep learning. They replaced Radon transform with a generalized Radon transform (GRT) by using defining functions, which allows the use of a broad class of injective functions. Further, Chen et al. (2022) proposed the SRT to make it easier to incorporate neural functions into a variant of Radon transform. There are also some variants of Radon Transform, which take into account the properties of the target data geometry (Bonet et al., 2023) and neural layers (Nguyen & Ho, 2022).

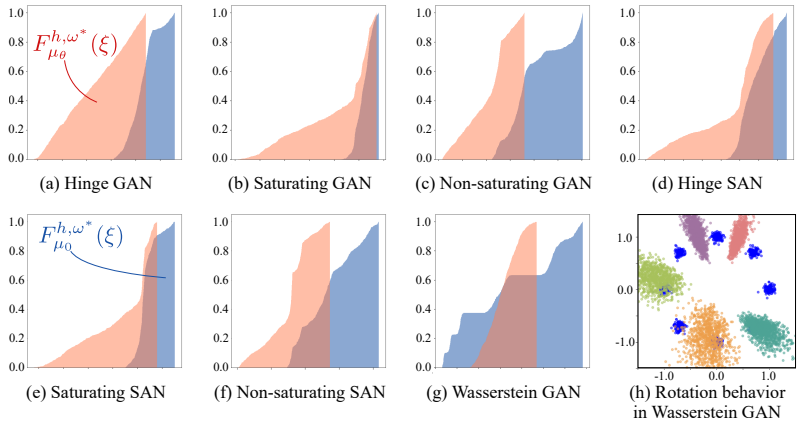
Regarding the slicing part, the most basic approach is to sample directions according to uniform distribution on the hypersphere to discretely approximate the integral of Eq. (8). Since the random nature of slices could result in an underestimation of the distance (Kolouri et al., 2019), more efficient sampling schemes have been investigated. Deshpande et al. (2019) proposed the concept of max-SW, which uses only the direction that maximizes the one-dimensional Wasserstein distance (e.g., the integrand in Eq. (8)). Nguyen et al. (2021) generalized the vanilla sliced distances and max-sliced distances by introducing probability distributions on the hypersphere.

Lastly, it is well known that a larger number of samples leads to a better approximation of the one-dimensional Wasserstein distance, thus leading to a stronger generation performance (Nguyen et al., 2021; Lezama et al., 2021; Chen et al., 2022). Therefore, simply increasing the batch size can enhance the performance. However, the number of samples utilized for the model training is typically constrained by the memory required to store the model (Lezama et al., 2021). To overcome this limitation, Lezama et al. (2021) proposed an algorithm that evaluates the one-dimensional Wasserstein distance with a virtual large batch size, although they did not apply the algorithm to the training of generative models from scratch but to the fine-tuning of GAN models due to its high computational cost. This kind of algorithm may be useful even for the fine-tuning of SAN models especially when the obtained h does not satisfy *separability* rigorously.

E.3 Limitations

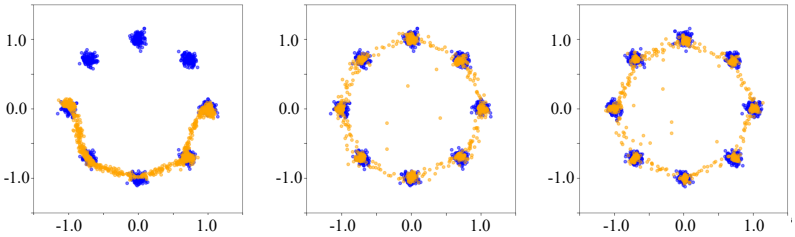
Metrizable conditions. Although our framework provides insights into GAN training without any unrealistic assumptions, a theoretical investigation into the *separability* is left as future work. In the current work, we conducted empirical studies on *separability* rather than a mathematical analysis. While our empirical observations on this property are consistent across different datasets, we believe additional theoretical investigation on it will be helpful for further understanding and improving GAN training.

SAN. One limitation of SAN is that it is not applicable to specific GANs (e.g., least-squares GAN) in a straightforward way. Naive least-squares SAN does not satisfy the metrizable condition, as its minimization problem does not satisfy the monotonic property regarding R in Theorem 5.3. However, we should emphasize that SAN still covers a broad class of GAN objective including f-GANs, and it is also possible to construct a new SAN objective by adopting different GAN variants for $\mathcal{J}_{\text{GAN}}$ and $\mathcal{V}_{\text{GAN}}$.



(a) Hinge GAN (b) Saturating GAN (c) Non-saturating GAN (d) Hinge SAN
(e) Saturating SAN (f) Non-saturating SAN (g) Wasserstein GAN (h) Rotation behavior in Wasserstein GAN

Figure 7: (a)–(g) Estimated cumulative density function at 5,000 iterations. *Separability* is satisfied in most GANs and SANs, but in the Wasserstein GAN, the discriminator does not satisfy *separability*. (h) Data samples and generated samples during training with Wasserstein GAN. Different colors represent different iterations (2,000, 4,000,. . ., 10,000). Rotational behavior is observed.



(a) Baseline (ReLU) (b) + LReLU (c) + GP

Figure 8: Effects of techniques to induce injectivity. The use of LReLU instead of ReLU and the addition of a GP to the maximization problem both improve the performance.

F Experimental Setup

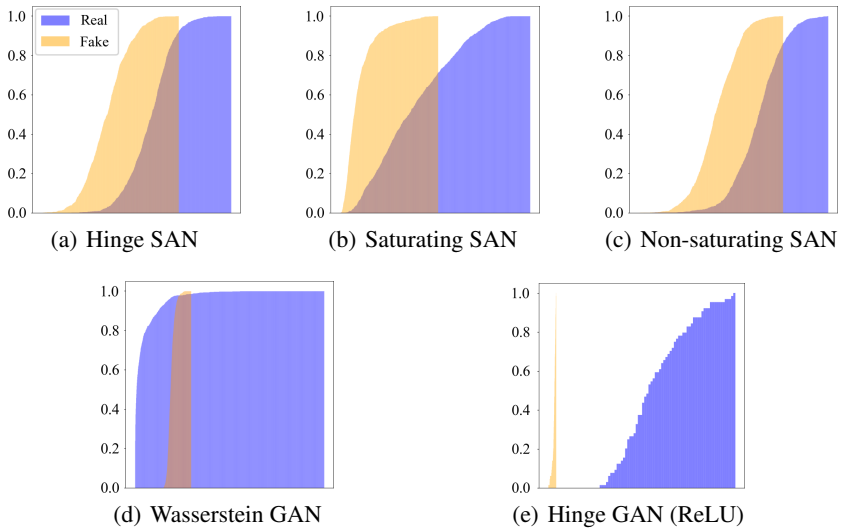
F.1 Mixture of Gaussians

In Sec. 7.1 and Appx. H, we use the same experimental setup as below. For both the generator and discriminator, we adopt simple architectures comprising fully connected layers by following the previous works (Mescheder et al., 2017; Nagarajan & Kolter, 2017; Sinha et al., 2020). We basically use leaky ReLU (LReLU) with a negative slope of 0.1 for the discriminator.

F.2 Image Generation

DCGAN. We borrow most parts of the implementation from a public repository[2]. Please refer to Tables 7 and 8 for details of the architectures. Spectral normalization is applied to the discriminator. We use the Adam optimizer (Kingma & Ba, 2015) with betas of $(0.0, 0.9)$ and an initial learning rate of 0.0002 for both the generator and discriminator. To convert GANs into SANs, we add two small modifications to the implementations. First, we apply a weight normalization instead of spectral normalization on the last linear layer of the discriminator, which maintains its exact weight on a hypersphere throughout the training. Next, we utilize the proposed objective functions for the maximization problems. We train the models for 2,000 iterations with the minibatch size of 64. We run the training with three different seeds for all the models and report the mean and deviation values in Table 3.

[2] https://github.com/christiancosgrove/pytorch-spectral-normalization-gan



(a) Hinge SAN (b) Saturating SAN (c) Non-saturating SAN

(d) Wasserstein GAN (e) Hinge GAN (ReLU)

Figure 9: Estimated cumulative density functions for the MNIST case. We follow the same procedure as in Fig. 5 to plot them. (a)–(c) *Separability* is satisfied by the SAN models. (d) In contrast, Wasserstein GAN does not satisfy the property, leading to bad generation results (same as in Fig. 7(g). (e) The use of ReLU in Hinge SAN does not destroy *separability* but leads to bad generation results due to the non-*injectivity* caused by the non-*injective* activation (see Fig. 10(a)).

Table 7: DCGAN architecture on CIFAR10.

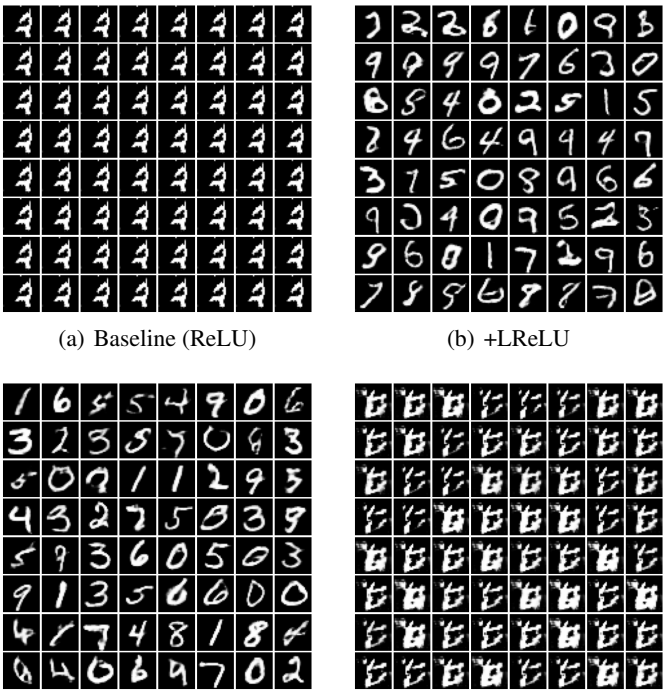
Generator
Input: $z \in \mathbb{R}^{128} \sim \mathcal{N}(0, I)$
Deconv (4×4, stride=1, channel=512), BN, ReLU
Deconv (4×4, stride=2, channel=256), BN, ReLU
Deconv (4×4, stride=2, channel=128), BN, ReLU
Deconv (4×4, stride=2, channel=64), BN, ReLU
Deconv (4×4, stride=1, channel=3), Tanh

Discriminator
Input: $x \in \mathbb{R}^{3\times32\times32}$
3×3, stride= 1, channel=64, LReLU
SNConv (4×4, stride= 2, channel=64), LReLU
SNConv (3×3, stride=1, channel=128), LReLU
SNConv (4×4, stride=2, channel=128), LReLU
SNConv (3×3, stride=1, channel=256), LReLU
SNConv (4×4, stride=2, channel=256), LReLU
SNConv (3×3, stride=1, channel=512), LReLU
SNLinear

BigGAN. We use the authors' PyTorch implementation[3] to compare GANs and SANs in the setting of BigGAN on CIFAR10. We exactly follow the instructions in the repository to train the GAN in the class conditional generation task. The discriminator is constructed with ReLUs, which may impede *injectivity* as discussed in Appx. H.1. Since the aim of the experiment is to validate our training scheme presented in Sec 6 in existing settings, we do not change the activation function. We defer an ablation study on injectivity induction by the activation functions to Appx. H.3. Regarding the unconditional generation task, we simply replace the class embedding for the last linear layer with a single linear layer. Similarly to the DCGAN setting, we implement the SANs only by modifying the last linear layer of the discriminator and the maximization objective function. We run the training with three different seeds for respective cases and report the mean and deviation values in Table 4. Note that it has been reported that there are numerical gaps between the evaluation scores reported in the original paper and those obtained by reproduction with the PyTorch implementation (Song & Ermon, 2020).

StyleGAN-XL. We use the official implementation[4] to compare a GAN and a SAN in the setting of StyleGAN-XL. We follow the instructions in the repository to train StyleSAN-XL in the class

[3] https://github.com/ajbrock/BigGAN-PyTorch
[4] https://github.com/autonomousvision/stylegan-xl



(a) Baseline (ReLU) (b) +LReLU

(c) +GP (d) Wasserstein GAN

Figure 10: (a)–(c) Effects of techniques to induce injectivity. (a) The use of ReLU leads to a collapse issue. (b) The use of LReLU instead of ReLU and (c) the addition of a GP to the maximization problem both improve the performance. (d) Generated results for Wasserstein GAN as a reference.

Table 8: DCGAN architecture on CelebA.

Generator
Input: $z \in \mathbb{R}^{128} \sim \mathcal{N}(0, I)$
Deconv (4×4, stride=1, channel=512), BN, ReLU
Deconv (4×4, stride=2, channel=256), BN, ReLU
Deconv (4×4, stride=2, channel=128), BN, ReLU
Deconv (4×4, stride=2, channel=64), BN, ReLU
Deconv (4×4, stride=2, channel=32), BN, ReLU
Deconv (4×4, stride=2, channel=3), Tanh

Discriminator
Input: $x \in \mathbb{R}^{3\times128\times28}$
3×3, stride= 1, channel=128, LReLU
SNConv (4 × 4, stride= 2, channel=128), LReLU
SNConv (3×3, stride=1, channel=256), LReLU
SNConv (4×4, stride=2, channel=256), LReLU
SNConv (3×3, stride=1, channel=512), LReLU
SNConv (4×4, stride=2, channel=512), LReLU
SNConv (3×3, stride=1, channel=1024), LReLU
SNLinear

conditional generation task. As in the DCGAN and BigGAN settings, we implement the SAN by modifying the last linear layer of the discriminator and the maximization objective function. Besides, since just normalizing the last linear layer may deteriorate the representation ability of the overall network, we introduce the trainable scaling factor $s \in \mathbb{R}$ to keep the ability. Note that the scaling scalar is included in h rather than ω to keep ω on the hypersphere, which is consistent with Listing 2. For ImageNet, we train a 256×256 generative model on top of the pretrained 128×128 generative model[5] in a progressive growing manner (Karras et al., 2018; Sauer et al., 2022). We set the `--kimg` option to 11,000 according to an author's comment in the repository[6]. We then report the FID and IS calculated with the provided code. For CIFAR10, we train a model from 16×16

[5] https://s3.eu-central-1.amazonaws.com/avg-projects/stylegan_xl/models/imagenet128.pkl

[6] https://github.com/autonomousvision/stylegan-xl/issues/43#issuecomment-1111028101

Table 9: Numerical results for BigGAN (Brock et al., 2019) and BigSAN on CIFAR10 in unconditional setting.

Method	FID (↓)	IS (↑)
BigGAN	17.16±1.34	8.42±0.11
BigSAN	**14.45**±0.58	**8.81**±0.04

Table 10: Numerical results for BigGAN (Brock et al., 2019) and the one fine-tuned by SAN training scheme.

Method	FID (↓)	IS (↑)
BigGAN (baseline)	8.25±0.82	9.05±0.05
BigGAN (fine-tuned with SAN)	**7.59**±0.23	**9.04**±7.59

resolution to the target resolution in a progressive way. We train a model with our arguments that should be different from the original, resulting in a different model size.

G Supplementary Experiments on BigGAN

In this section, we present two supplementary experimental results to further support the effectiveness of SAN training scheme.

Unconditional BigSAN. We also compare SAN and GAN with BigGAN architecture in an unconditional generation case by following the experimental setting in Song & Ermon (2020). As reported in Table 9, SAN consistently improves the generation performance in terms of FID and IS.

Fine-tuning GAN with SAN. Another interesting question is whether SAN training scheme is valid even for fine-tuning GAN. We fine-tuned the BigGAN models trained on CIFAR-10 with the SAN training objectives for 10k iterations. We found our proposed method is valid for fine-tuning existing GANs as reported in Table 10. Still, training the SAN from scratch benefits from our modification scheme more, leading to better performance than the fine-tuned model.

H Investigation of Separability and Injectivity

In this section, *separability* and *injectivity* are empirically investigated.

H.1 Inductions for Injectivity

We present some potential inductions for injectivity, which are empirically verified in the next subsections.

There are various ways to impose *injectivity* on the discriminator, one of which is to implement the discriminator with an invertible neural network. Although this topic has been actively studied (Behrmann et al., 2019; Karami et al., 2019; Song et al., 2019), such networks have higher computational costs for training (Chen et al., 2022). Another way is to add regularization terms to the maximization problem. For example, a gradient penalty (GP) (Gulrajani et al., 2017) can promote injectivity by explicitly regularizing the discriminator's gradient. The penalty enforces the norm of the discriminator Jacobian to be unity. This mitigates the occurrence of zero-gradient, one of the prominent factors contributing to non-injectivity.

In contrast, simple removal of operators that can destroy *injectivity*, such as ReLU activation, can implicitly overcome this issue. It is obvious that invertible activations (such as leaky ReLU) contribute to layerwise injectivity. Several studies suggest that constructing an injective neural network with ReLU is more challenging compared to invertible activation functions (Puthawala et al., 2022; Chan et al., 2023).

Table 11: Effect of injectivity induction by activation functions on FID and IS in the BigSAN setting. The use of LReLU leads to better or competitive scores than that of

Metric	CIFAR10		CIFAR100	
	BigSAN w/ ReLU	BigSAN w/ LReLU	BigSAN w/ ReLU	BigSAN w/ LReLU
FID (↓)	6.20±0.27	5.97±0.12	8.05±0.04	7.97±0.06
IS (↑)	9.16±0.08	9.23±0.03	10.72±0.05	10.84±0.03

H.2 Mixture of Gaussians

We visualize the generated samples mainly to confirm that the generator measures cover all the modes of the eight Gaussians. In addition, we plot the cumulative density functions of $\mathcal{S}^h I_{\mu_0}(\cdot, \omega)$ and $\mathcal{S}^h I_{\mu_\theta}(\cdot, \omega)$ to verify *separability*. As shown in Fig. 7, the cumulative density function for the generator measures trained with Wasserstein GAN does not satisfy *separability*, whereas the generator measures trained with other GAN losses do satisfy it. This may be what causes the rotation behavior of the Wasserstein GAN, as seen in Fig. 7(h).

To investigate the effect of *injectivity* on the training results, we train Hinge GAN using a discriminator with ReLU as a baseline. We apply two techniques to induce *injectivity*: (1) the use of LReLU and (2) the addition of a GP to the discriminator's maximization problem. As shown in Fig. 8, with either of these techniques, the training is improved and mode collapse does not occur.

H.3 Vision Datasets

We extend the experiment in Appx. H.2 to MNIST (LeCun et al., 1998) to investigate *injectivity* and *separability* on vision datasets. First, we train SANs with various objective functions and Wasserstein GAN on MNIST. We adopt DCGAN architectures and apply spectral normalization to the discriminator. As in Appx. H.2, we plot the cumulative density function for the generator measures trained with various objective functions in Fig. 9. The trained generator measures empirically satisfy *separability* except for the case of Wasserstein GAN, which is a similar tendency to the case of MoG. We also conducted an ablation study regarding the technique inducing *injectivity* by following Appx. H.2. We train Hinge SAN using a discriminator with ReLU as a baseline, and compare this baseline with two cases: (1) the use of LReLU and (2) the addition of a GP to the discriminator's objective function. The generated samples for these cases are shown in Fig. 10(a)–(c). As we can see, the generator trained with the ReLU-based discriminator suffers from the mode collapse issue, while in contrast, the techniques inducing *injectivity* mitigate the issue and lead to reasonable generation results. As a reference, we depict the generation result from a generator that is trained with a LReLU-based discriminator and Wasserstein GAN in Fig. 10(d), where we can see that the generated samples are completely corrupted. Note that the discriminator with ReLU satisfies *separability*, the same as in Fig. 9(e). This suggests that the deterioration in Hinge SAN with the ReLU-based discriminator (Fig. 10(a)) and that in the Wasserstein GAN model (Fig. 10(d)) are triggered by different causes, i.e., from non-injectivity and non-separability, respectively.

To further verify our insights in more realistic setting, we conduct another experiment using BigGAN architecture, where the discriminator is constructed with ReLUs. For ablation of the injectivity induction by the activation functions, we replace ReLUs with LReLUs in the discriminator, and subsequently train BigSAN on both CIFAR-10 and CIFAR-100. The use of the injective activation leads to slightly better scores than the original BigSAN as reported in Table 11. This result is consistent with our theoretical and empirical investigations.

I Investigation of $\mathcal{V}_{\text{GAN}}$ in SW

The aim of this empirical study is to compare $\mathcal{V}_{\text{GAN}}$ in the SW context. To this end, we train max-SW-based generative models on CIFAR10 using the same experimental setting as in Sec. 7.2 (DCGAN). We apply the ASW to max-SWGAN (Deshpande et al., 2019), which was shown to achieve competitive or better generation performances compared to the uniform sampling of ω despite reducing the computational complexity and the memory footprint of the model. We train the model by the

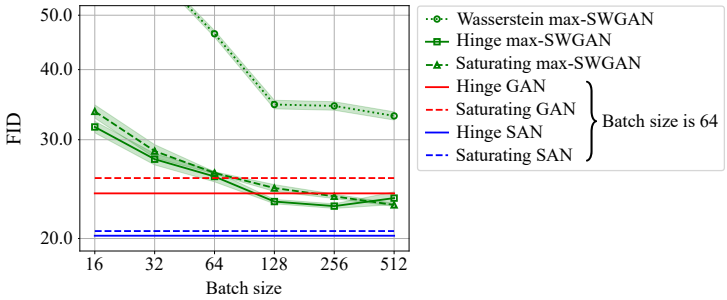


Figure 11: Empirical studies on the impact of batch size on SWGAN trained on CIFAR10. The FID score decreases as the batch size increases especially with $\mathcal{V}_{\mathrm{W}}$.

following optimization problem:

$$\max_{\omega,h} \mathcal{V}_{\mathrm{GAN}}(\langle \omega, h(x) \rangle), \quad \text{and} \tag{51a}$$

$$\min_{\theta} W_2(\mathcal{S}^h I_{\mu_0}(\cdot,\omega), \mathcal{S}^h I_{\mu_\theta}(\cdot,\omega)), \tag{51b}$$

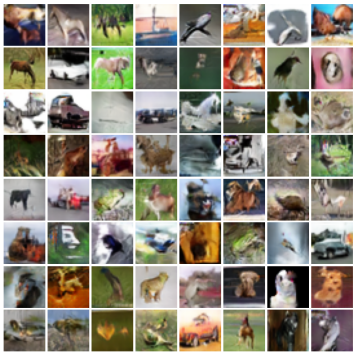
Note that the maximizer ω in Eq. (51a) does not generally maximize the Wasserstein distance in Eq. (51b)[7]. Since the minimization problem (51b) requires sample sorting according to discriminator values to approximate the Wasserstein distance, the larger batch size is expected to enable a better distance approximation, leading to a better generation performance, at the expense of computational cost (refer to Fig. 6 in Nguyen et al. (2021) and Fig. 14 in Chen et al. (2022)). Regarding $\mathcal{V}_{\mathrm{GAN}}$, we utilize the maximization problems for Wasserstein GAN, hinge GAN, and saturating GAN (non-saturating GAN) for comparison.

We plot the FID score w.r.t. batch size in Fig. 11. Interestingly, with the maximization problem of Wasserstein GAN, the batch size greatly influences the generation performance, unlike the other GAN variants. We suspect the smaller sample size causes noisy approximation of the Wasserstein distance, since $\mathcal{V}_{\mathrm{W}}$ tends to bring non-separable h (as shown in our experiments in Sec. H). In contrast, the other GAN variants lead to reasonable generative models even with a smaller batch size. The FID scores of GAN and SAN reported in Table 3 are also plotted here for reference. When the batch size is larger, the FID score of the max-SWGANs converges to a certain value, which is better than that of GAN but worse than that of SAN. We suspect this is because, among the comparison methods, SAN training is the only way to ensure *optimal direction*.

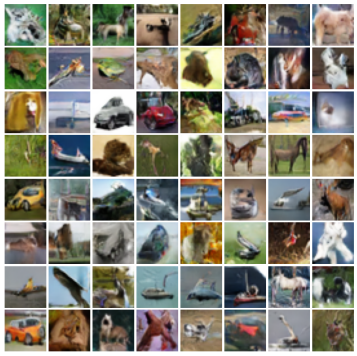
J Generated Samples

Images generated by GANs and SANs are shown in Figs. 12, 13, 14, and 16.

[7]Although the idea of max-SW is to select the *optimal direction* (Deshpande et al., 2019), the proposed model does not satisfy the property due to its surrogate maximization objective $\mathcal{V}_{\mathrm{GAN}}$.



(a) Saturating GAN



(b) Saturating SAN

Figure 12: Generated samples from DCGAN trained on CIFAR10.

(a) Non-saturating GAN

(b) Non-saturating SAN

Figure 13: Generated samples from DCGAN trained on CelebA (128×128).

(a) Hinge GAN (unconditional)

(b) Hinge SAN (unconditional)

(c) Hinge GAN (conditional)

(d) Hinge SAN (conditional)

Figure 14: Generated samples from BigGAN trained on CIFAR10.

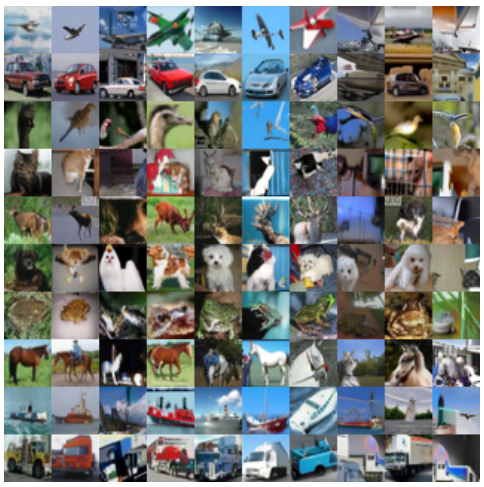


Figure 15: Generated samples from StyleSAN-XL trained on CIFAR10.

[8] https://s3.eu-central-1.amazonaws.com/avg-projects/stylegan_xl/models/imagenet256.pkl

(a) StyleGAN-XL

(b) StyleSAN-XL

Figure 16: Generated samples from StyleGAN-XL[8]and StyleSAN-XL trained on ImageNet.