

---

# Non-Parametric Priors For Generative Adversarial Networks

---

Rajhans Singh<sup>1</sup> Pavan Turaga<sup>1,2</sup> Suren Jayasuriya<sup>1,2</sup> Ravi Garg<sup>3</sup> Martin W. Braun<sup>3</sup>

## Abstract

The advent of generative adversarial networks (GAN) has enabled new capabilities in synthesis, interpolation, and data augmentation heretofore considered very challenging. However, one of the common assumptions in most GAN architectures is the assumption of simple parametric latent-space distributions. While easy to implement, a simple latent-space distribution can be problematic for uses such as interpolation. This is due to distributional mismatches when samples are interpolated in the latent space. We present a straightforward formalization of this problem; using basic results from probability theory and off-the-shelf-optimization tools, we develop ways to arrive at appropriate non-parametric priors. The obtained prior exhibits unusual qualitative properties in terms of its shape, and quantitative benefits in terms of lower divergence with its mid-point distribution. We demonstrate that our designed prior helps improve image generation along any Euclidean straight line during interpolation, both qualitatively and quantitatively, without any additional training or architectural modifications. The proposed formulation is quite flexible, paving the way to impose newer constraints on the latent-space statistics.

## 1. Introduction

Advances in deep learning have resulted in state-of-the-art generative models for a wide variety of data generation tasks. Generative methods map sampled points in a low-dimensional latent space with known distribution to

points in high-dimensional space with distributions matching real-data. In particular, generative adversarial networks (GANs) (Goodfellow et al., 2014) have shown successful applications in super-resolution (Ledig et al., 2017), image-to-image translation (Isola et al., 2017; Zhu et al., 2017), text-to-image translation (Reed et al., 2016), image inpainting (Pathak et al., 2016), image manipulation (Zhu et al., 2016), synthetic data generation (Shrivastava et al., 2017) and domain adaptation (Tzeng et al., 2017).

A GAN architecture consists of a generator  $G$  and a discriminator  $D$ . The generator  $G$  maps low-dimensional latent points  $z \sim \mathcal{P}_z$  to high-dimensional data distribution  $\mathcal{P}_{data}$ . The latent-space distribution  $\mathcal{P}_z$  is typically chosen to be a normal or uniform distribution. The goal of the generator  $G$  is to produce data such that they are perceptually indistinguishable from real data. However, the discriminator  $D$  is trained to distinguish between ‘fake’ and ‘real’ data. Both the generator and discriminator are trained in an adversarial fashion, and at the end of training the generator learns to generate data with a distribution similar to the real one.

One natural question for generative models is how to model the latent space effectively to generate diverse and varied output. Interpolating between samples in the latent space can lead to semantic interpolation in image space (Radford et al., 2016). Interpolation can help transfer certain semantic features of one image to another. Successful interpolation also shows that GANs do not simply over-fit or reproduce the training set, but generate novel output. Interpolation has been shown to disentangle factors of variation in the latent space with many applications (Liu et al., 2018a; Kumar & Chellappa, 2018; Liu et al., 2018b; Yin et al., 2017).

Imposing a parametric structure on the latent space can cause distributional mismatches where the prior distribution does not match the interpolated point’s distribution. This mismatch causes the interpolated points to lose fidelity in quality (White, 2016). Previous research has resulted in various parametric models to fix this problem (White, 2016; Kilcher et al., 2018; Agustsson et al., 2019). One of the findings in prior work (Leśniak et al., 2019) is that the use of a Cauchy distributed prior solves the distributional mismatch problem. But, Cauchy is a very peculiar distribution, with undefined moments and a heavy-tail. This means that during inference there will always be a number of undesirable outputs (as acknowledged also in (Leśniak et al., 2019)) due

---

<sup>1</sup>School of Electrical, Computer, and Energy Engineering, Arizona State University, Tempe, AZ, USA <sup>2</sup>School of Arts, Media and Engineering, Arizona State University, Tempe, AZ, USA <sup>3</sup>Intel Corporation, Chandler, AZ, USA. Correspondence to: Rajhans Singh <rsingh70@asu.edu>, Pavan Turaga <pturaga@asu.edu>, Suren Jayasuriya <sjayasur@asu.edu>, Ravi Garg <ravi.garg@intel.com>, Martin W. Braun <martin.w.braun@intel.com>.

to latent vectors being sampled from the these tails.

In this paper, we propose the use of **non-parametric priors** to address the aforementioned issues. The advantage of a non-parametric prior is that we do not use any modeling assumptions and propose a general optimization approach to determine the prior for the task at hand. In particular, our contributions are as follows:

- We analyze the distribution mismatch problem in latent-space interpolation using basic probability tools, and derive a non-parametric approach to search for a prior which can address the distribution mismatch problem.
- We present algorithms to solve for the prior using off-the-shelf optimizers, and show that obtained priors have interesting multi-lobe structures with fast decaying tails, resulting in mid-point distribution to be close to the prior.
- We show that the resulting non-parametric prior yields better quality and diversity in generated output, with no additional training data nor any added architectural complexity.

More broadly, our approach is a general and flexible method to impose other constraints on latent-space statistics. Our goal is not to outperform all the latest developments in generative models, but to show that our proposed stand-alone formulation can boost performance with no added training or architectural modifications.

## 2. Background and Related Work

**Generative Adversarial Network:** As described in Section 1, a GAN consists of two components: a generator  $G$  and a discriminator  $D$ , which are adversarially trained against one another until the generator can map latent-space points to a high dimensional distribution which the discriminator cannot distinguish from true data samples. Formally, this can be expressed as a min-max game in (1) which the generator tries to minimize and the discriminator tries to maximize (Goodfellow et al., 2014):

$$\min_G \max_D V(D, G) = E_{x \sim \mathcal{P}_x} [\log(D(x))] + E_{z \sim \mathcal{P}_z} [\log(1 - D(G(z)))], \quad (1)$$

where,  $V$  is the objective function,  $x \sim \mathcal{P}_x$  are real data points sampled from a true distribution, and  $z \sim \mathcal{P}_z$  are sampled points from the latent-space distribution. If the training of the GAN is stable and the Nash equilibrium is achieved, then the generator learns to generate samples similar to the true distribution. In general, GAN training is not always stable, thus several methods have been introduced to improve the training (Salimans et al., 2016; Arjovsky & Bottou, 2017). This includes different kinds of divergences and loss functions (Nowozin et al., 2016; Arjovsky et al.,

2017; Gulrajani et al., 2017). Several other works improve generated image quality (Dai et al., 2017; Zhang et al., 2017) or resolution (Denton et al., 2015; Karras et al., 2018).

**Interpolation:** For any two given latent-space points  $z_1, z_2 \sim \mathcal{P}_z$ , a linearly-interpolated point  $z_\lambda$  is given by  $z_\lambda = (1 - \lambda)z_1 + \lambda z_2$  for some  $\lambda \in [0, 1]$ . It has been shown that GANs can generate novel outputs via linear interpolation, and as the line is traversed ( $\lambda : 0 \rightarrow 1$ ), the output images smoothly transition from one to another without visual artifacts (Radford et al., 2016). They further showed that vector arithmetic in the latent space has corresponding semantic meaning in the output space, e.g. latent-space points for “man with glasses” - “man without glasses” + “woman without glasses” generates an image of a woman wearing glasses (c.f. Fig. 7 of (Radford et al., 2016)).

**Distribution Mismatch of Interpolated Points:** Interpolation, while semantically meaningful, presents challenges in ensuring all interpolated points preserve the same data quality (or in the case of images visual quality). Most GANs utilize simple parametric distributions such as normal or uniform as the prior distribution to sample the latent space. However, these two choices of priors cause the interpolated point’s distribution to not match with either the normal or uniform distribution as observed by (Kilcher et al., 2018). We replicate this argument below for the sake of exposition, since this is the core problem we tackle in this paper.

Let  $z_1, z_2 \sim \mathcal{N}(0, \sigma^2 I)$  be two points in the latent space of the GAN’s generator, and let  $z_\lambda = (1 - \lambda)z_1 + \lambda z_2$  be a linearly interpolated point. Note that  $\sigma_{z_\lambda}^2 = (1 - \lambda)^2 \sigma_z^2 + \lambda^2 \sigma_z^2$ . We are interested when  $\sigma_{z_\lambda}^2 = \sigma_z^2$ , which only holds for delta functions for finite moment distributions (we later prove this statement more formally in Section 3). However, we see that the worst case for when  $\sigma_{z_\lambda}^2$  is different from  $\sigma_z^2$  occurs at  $\lambda = 1/2$  or the midpoint denoted  $z' = \frac{z_1 + z_2}{2}$ . Analyzing the Euclidean norm of  $z_1, z_2$  and  $z'$  gives the following equations (Kilcher et al., 2018):

$$\begin{aligned} z_1, z_2 \sim \mathcal{N}(0, \sigma^2 I) &\Rightarrow \|z_1\|^2, \|z_2\|^2 \sim \sigma^2 \mathcal{X}^2(d), \\ z' = \frac{z_1 + z_2}{2} &\Rightarrow \|z'\|^2 \sim \frac{\sigma^2}{2} \mathcal{X}^2(d), \end{aligned} \quad (2)$$

where,  $d$  is the dimension of the latent-space and  $\mathcal{X}^2$  is the chi-squared distribution. The GAN is trained with latent-space whose norm squared distance follows a  $\sigma^2 \mathcal{X}^2(d)$  distribution according to (2). However, the mid-point will have a distribution  $\frac{\sigma^2}{2} \mathcal{X}^2(d)$ . Clearly, there is distribution mismatch between the points at which the GAN is trained to generate realistic samples and the mid-point where we want to do interpolation. Further, this distribution mismatch becomes worse if we increase the dimension of the latent

space. Finally, it has been shown that the Normal distribution in higher dimensions forms a ‘soap bubble’ structure, i.e. most of its probability mass concentrates in an annulus around the mean, rather than near the mean itself (Hall et al., 2005). This implies that interpolations that traverse near the origin of the latent space will suffer degradation in output fidelity/quality, which has been confirmed for GANs in (White, 2016). A similar proof for a distribution mismatch can be shown for the uniform distribution.

We note that we are not the first to propose a solution to this problem. Several prior approaches have proposed solutions, either through new interpolation schemes or new prior distributions different from normal or uniform that suffer less distribution mismatch. White et al. proposes spherical linear interpolation by following the geodesic curve on a hypersphere to avoid interpolating near the origin (and thereby minimize distribution mismatch) if the latent points are sampled uniformly on a sphere with finite radius (White, 2016). However, this interpolation is not semantically meaningful if the path becomes too long and it passes through unnecessary images as noted in (Kilcher et al., 2018). Similar to spherical interpolation, (Agustsson et al., 2019) propose a normalized interpolation. Yet, it inherits similar issues as in spherical interpolation in not being the shortest path.

Other approaches have attempted to define new prior distributions to ensure the interpolated points have low distribution mismatch. This is similar to the method we employ in our paper, except ours is non-parametric. Kilcher et al. propose a new prior distribution defined as follows:

$$v \sim \text{Uniform}(S^{d-1}), r \sim \Gamma(\frac{1}{2}, \theta), z = \sqrt{r}v \quad (3)$$

where  $d$  is the dimension of the latent space,  $\Gamma(\frac{1}{2}, \theta)$  is the Gamma distribution and  $z$  is a latent vector (Kilcher et al., 2018). Latent spaces defined using this prior distribution do not suffer as much from mid-point distribution mismatch. Further work by Lesniak et al. showed that the Cauchy distribution  $P(z) = \frac{1}{\pi(1+z^2)}$  induces a midpoint distribution which is the same as itself (Leśniak et al., 2019). However, the Cauchy distribution has undefined moments, which makes analysis difficult, and also is heavy-tailed, which can lead to undesirable outputs.

### 3. Design of non-parametric priors for GANs

The primary motivation to design non-parametric priors is that in order to have a prior whose distribution of mid-points is close to the original prior, we need to optimize for an appropriate cost over the space of density functions. This optimization is easily done for the non-parametric case, and requires rather few assumptions. Our terminology of *non-parametric* stems from classical density estimation ap-

proaches, rather than the more modern usage in Bayesian non-parametric.

Let  $f_X(x)$  be the chosen prior distribution. Let  $x_1$  and  $x_2$  be two samples drawn from  $f_X(x)$ . Let an interpolated point be given by:  $(1 - \lambda)x_1 + \lambda x_2$ , for  $0 < \lambda < 1$ . The precise relation between the distribution of this interpolated point and  $f_X(x)$  is given analytically as follows.

**Property 1:** If  $x_1$  and  $x_2$  are two independent samples drawn from  $f_X(x)$ , then the density function of  $(1 - \lambda)x_1 + \lambda x_2$ , for  $0 < \lambda < 1$  is given by  $\frac{1}{|\lambda(1-\lambda)|} f_X(\frac{x}{\lambda}) * f_X(\frac{x}{1-\lambda})$ .

**Proof:** The proof is a direct application of the following two results from probability theory.

- $R_1$ : If  $X_1$  and  $X_2$  are two independent random variables, with density functions  $f_{X_1}(x)$  and  $f_{X_2}(x)$ , then the density of their sum  $X_1 + X_2$  is given by the linear convolution  $f_{X_1}(x) * f_{X_2}(x)$ .
- $R_2$ : If random variable  $X$  has density function  $f_X(x)$ , then for  $\alpha \in \mathbb{R}$ , the density of  $\alpha X$  is given by  $\frac{1}{|\alpha|} f_X(\frac{x}{\alpha})$ .

Apply  $R_2$  to  $(1 - \lambda)x_1$  and  $\lambda x_2$  separately, then convolve the results using  $R_1$ . QED.

Following from here, the distribution mismatch problem can be expressed as the search for a prior distribution  $f_X(x)$  such that the distribution of any other interpolated point is close to  $f_X(x)$ . That is, we would like  $f_X(x)$  to satisfy:

$$f_X(x) = \frac{1}{|\lambda(1-\lambda)|} f_X(\frac{x}{\lambda}) * f_X(\frac{x}{1-\lambda}), \quad (4)$$

where,  $\lambda \in (0, 1)$ . The only distributions we are aware of that satisfy this condition are the Cauchy (undefined moments, heavy-tailed), and delta functions (zero variance).

**Property 2:** The only density functions with finite moments that satisfy condition (4) are delta functions.

**Proof:** A density function that satisfies condition (4), will also satisfy the equality of all moments of the left and right side densities. By specifically applying this to the equality of variances, we will show that the only solution is a delta function (among the class of finite moment densities). The following two results come handy.

- $R_3$ : If  $X_1$  and  $X_2$  are two independent random variables, with respective variance  $\sigma_1^2$  and  $\sigma_2^2$ , then the variance of their sum  $X_1 + X_2$  is given by  $\sigma_1^2 + \sigma_2^2$ .
- $R_4$ : If random variable  $X$  has variance  $\sigma^2$ , then for  $\alpha \in \mathbb{R}$ , the variance of  $\alpha X$  is given by  $\alpha^2 \sigma^2$ .

If (4) holds, it must imply the equality of the variance of the prior, and the variance of any intermediate-point. i.e.

$\sigma_X^2 = (1 - \lambda)^2 \sigma_X^2 + \lambda^2 \sigma_X^2$ . If  $\sigma_X$  is finite, equality happens if and only if  $\sigma_X = 0$ . This implies that  $f_X(x)$  is a delta function. QED.

The search for a density function that satisfies (4) is thus not meaningful in the context of generative models, because a delta function as prior implies constant output. Cauchy, on the other hand, is too specific a choice, and suffers from pathologies such as undefined moments, which renders imposing any additional constraints on latent-space statistics impossible. It also is heavy-tailed, which may cause generation of undesirable outputs for samples from the tails.

What if we relax condition (4), such that we do not seek exact equality, but closeness of the left and right sides? The next section shows that this relaxed search results in a problem which can be solved using standard off-the-shelf function minimizers. Using this approach we obtain distributions with many useful properties. If we let  $P(x) = f_X(x)$ , and  $Q(x; \lambda) = \frac{1}{\lambda(1-\lambda)} f_X(\frac{x}{\lambda}) * f_X(\frac{x}{1-\lambda})$ , we seek to minimize some form of distance or divergence between  $P(x)$  and  $Q(x)$  among densities with finite-variance. This optimization problem is defined in the next section.

### 3.1. Searching for the optimal prior distribution

As mentioned earlier, instead of enforcing exact equality as in condition (4), we would like to minimize the discrepancy between the left and right sides. A natural choice would be to minimize the KL divergence between  $P(x) = f_X(x)$ , and  $Q(x; \lambda) = \frac{1}{\lambda(1-\lambda)} f_X(\frac{x}{\lambda}) * f_X(\frac{x}{1-\lambda})$ . Ideally, it might make sense to minimize this over the entire range of  $\lambda$ 's, i.e.

$$\underset{P}{\text{minimize}} \int f(P(x) \| Q(x; \lambda)) d\lambda, \quad (5)$$

where  $f$  is the chosen divergence/distance between the densities. However, this is likely an intractable problem due to integration over  $\lambda$ . In order to make this tractable, we observe that for a given  $\lambda$ , the mean of  $Q(x; \lambda)$  is the same as the mean of  $P(x)$ . However, the variance of  $Q(x; \lambda)$  goes as  $(1 - \lambda)^2 \sigma_X^2 + \lambda^2 \sigma_X^2$ . Thus, for interpolation problems, where  $\lambda \in (0, 1)$ , the largest discrepancy in variance between  $P$  and  $Q$  occurs at a value of  $\lambda = 0.5$ . Thus, for interpolation problems, we suggest that minimizing for the worst-case error is sufficient.

### 3.2. Optimization problem for interpolation priors

For interpolation priors, we minimize the KL divergence between  $P(x)$  and  $Q(x; \lambda = 0.5)$ . To create a tractable problem, we restrict  $P(x)$  to be defined over a compact domain, without loss of generality, we choose it to be  $[0, 1]$ . We discretize the domain with sufficient fineness, typically we choose  $2^{10}$  bins in  $[0, 1]$ . The distribution is now dis-

cretely represented by the bin-centers  $P = \{p_i\}_{i=1}^n$ . The optimization problem now becomes:

$$\underset{P}{\min} f(P \| Q) \text{ s. t. } \sum_{i=1}^n p_i = 1, \text{ and } p_i \geq 0. \quad (6)$$

In (6),  $f(P \| Q)$  is a divergence/distance function between  $P$  and  $Q$ . We use KL divergence because not only is it a natural choice, but we also find that it produces smooth distributions than when using the  $\ell_2$  distance. Without a variance constraint, the solution of (6) is simply a discrete delta function, which we would like to avoid. The variance constraint is equivalently expressed as a quadratic-term involving  $p_i$ 's, based on which we have:

$$\begin{aligned} & \underset{P}{\min} f(P \| Q) \\ \text{s.t. } & \sum_{i=1}^n p_i = 1, \frac{1}{n} \left( \sum_{i=1}^n i^2 p_i - \left( \sum_{i=1}^n i p_i \right)^2 \right) \geq \xi, p_i \geq 0, \end{aligned} \quad (7)$$

where,  $\xi > 0$ . In general the KL divergence is convex on the space of density functions, but in our case  $P$  and  $Q$  are related to each other; our objective function is not convex. We solve (7) using *fmincon* in Matlab, which uses an interior-point algorithm with barrier functions. We note that the solution from *fmincon* may only be a locally optimal solution, yet we find the obtained solution is quite robust to large variation in initialization. We also note that using any of  $KL(P \| Q)$ ,  $KL(Q \| P)$  and  $KL(P \| \frac{P+Q}{2}) + KL(Q \| \frac{P+Q}{2})$  as objective in (7) gives us the same result. We use the following settings for *fmincon*: interior-point as algorithm, *Max Function Evaluations* =  $4 \times 10^5$ , *Max Iterations* =  $10^5$  and  $n = 1024$ . We use  $\xi = 0.75$  in our experiments because it provides the best FID score (Heusel et al., 2017). Figure 1 shows the trace of the optimizer cost value for the above settings; we observe convergence to a local minimum in less than 500 iterations.

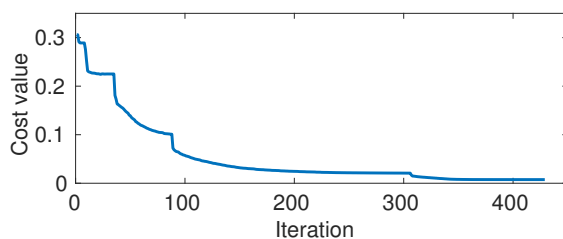


Figure 1. Figure shows a sample trace of the cost function (7) over iterations, showing fast convergence.



**Remarks on the shape of the obtained distribution:**

Here we make brief remarks on the shape of the obtained distribution. Firstly, we note that the exact shape of the obtained distribution varies slightly each time we run the solver. This is of course expected. However, we find that all obtained solutions seem to share the same general shape: they have a large main-lobe, appear to be symmetric, and have small but significant side-lobes. This is more clearly shown in Figure 2. We note that we did not impose any symmetry condition during optimization, yet these solutions emerged despite different initialization.

**Dependence on initialization:** We initialized our solver with a uniform density, delta functions centered at different locations, and truncated Gaussians with varying means and standard-deviations. For all these, the final solution still converges to a shape very similar to that shown in Figure 2. Further, all obtained solutions seem to perform equally well in the final evaluation of GAN output quality.

**Role of side-lobes:** We are not aware of any simple parametric distribution that can describe the shape seen in Figure 2, except perhaps a Gaussian mixture model. However, the shape of the side-lobes is intricate, and not simple Gaussian-like. The existence of these side-lobes seems to allow us to strike a balance between the fast tail-decay of distributions like the Gamma, and the heavy tail of the Cauchy. It is almost as if the obtained shape fuses the best properties of the two classes of distributions, enabling us to generate good quality GAN output, all the while minimizing the divergence to the interpolated samples.

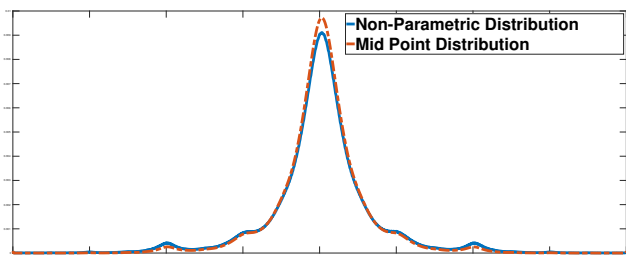


Figure 2. The distribution obtained by solving (7) (shown in blue), and its mid-point distribution (red). While there is small variability in the solutions obtained, we find that no matter how we initialize the solver, all our obtained distributions share the three following traits: a large main-lobe, symmetry, and small side-lobes. Also, note the strong overlap between the distribution and the mid-point distribution. This is further quantified in table 1 and compared with other distributions in figure 4.

**Continuous samples from discretized density:** At first glance it may appear that since we discretize the domain  $[0, 1]$  while solving (7), that our prior is capable of generating only discrete samples. This is easily dealt with as follows. Once we generate a sample from the discretized density, what we get is really an index corresponding to the

Table 1. KL-divergence between prior and mid-point distribution.

Distribution	KL divergence
Uniform Distribution	0.3065
Normal Distribution	0.1544
Proposed Non-Parametric Distribution	<b>0.0075</b>

bin-center, but the bin itself has non-zero width given by how finely we partition  $[0, 1]$ . From the corresponding bin, we simply generate a uniform random variable restricted to the width of the bin. This approach implicitly corresponds to sampling from a continuous density constructed by a zeroth-order interpolation over the obtained discrete one. One can be more sophisticated than this, but the sampling algorithm will no longer be as simple. We find that the approach described above is quite sufficient in practice.

**Quantification of mid-point mismatch:** Table 1 shows the actual KL divergence between the prior and mid-point distribution. It is clear that for the normal and uniform distributions, the mid-point distribution is very divergent from the actual prior distribution; whereas the distribution obtained from solving (7) has a much lower divergence from the mid-point. In (7) we are only minimizing the distribution mismatch for the one-dimensional case. The idea is that if the 1-D distribution is similar to its mid-point distribution, then the divergence between the corresponding Euclidean norm distribution will be low even for higher dimensions.

Figure 3 shows the mid-point distribution mismatch for different priors for the one dimensional case. Figure 4 shows the Euclidean norm distribution for prior and mid-points for different dimensions, computed from a set of  $5 \times 10^4$  samples. For the mid-points, we sample two sets of  $5 \times 10^4$  points, and calculate the Euclidean norm of the corresponding mid-points. We see that for the normal distributions, at low dimensions ( $d = 5$  and  $d = 10$ ) the mid-point distribution overlaps well with the prior distribution. As the dimension increases ( $d = 50, 100, 200$ ), the two distributions start to diverge. For  $d = 100, 200$  there is almost no overlap between the prior and mid-point distribution. We observe similar trend for the uniform distribution. On the contrary, in our case the mid-point distribution and the prior distribution (of Euclidean norm) overlap well with each other even in higher dimensions. In Figure 4 we can notice that our non-parametric distribution does bring the Euclidean norm distribution very close to the origin compared to the normal and uniform.

**4. Experiments and Results**

**Datasets, models, and baselines:** To validate the effectiveness of the proposed approach, we train the standard DCGAN model (Radford et al., 2016) on four different

## Non-Parametric Priors For GAN

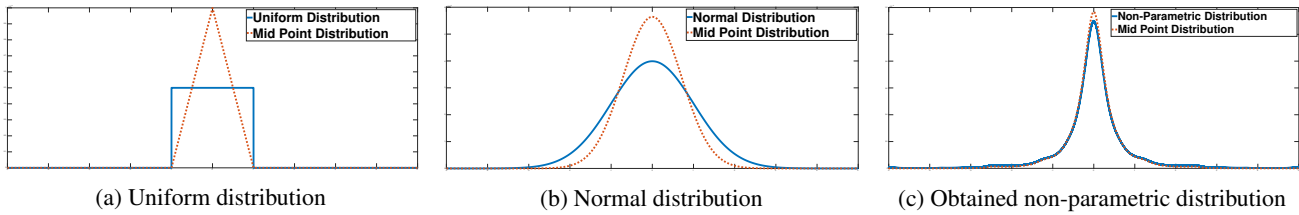


Figure 3. The figures show various choices of priors (blue) and their corresponding mid-point distribution (red). Note that one can observe a large discrepancy between the prior and mid-point distributions, for typical choices such as the uniform and Normal. The prior we develop shows significantly less discrepancy. These discrepancies are also quantified via the KL-divergence in table 1.

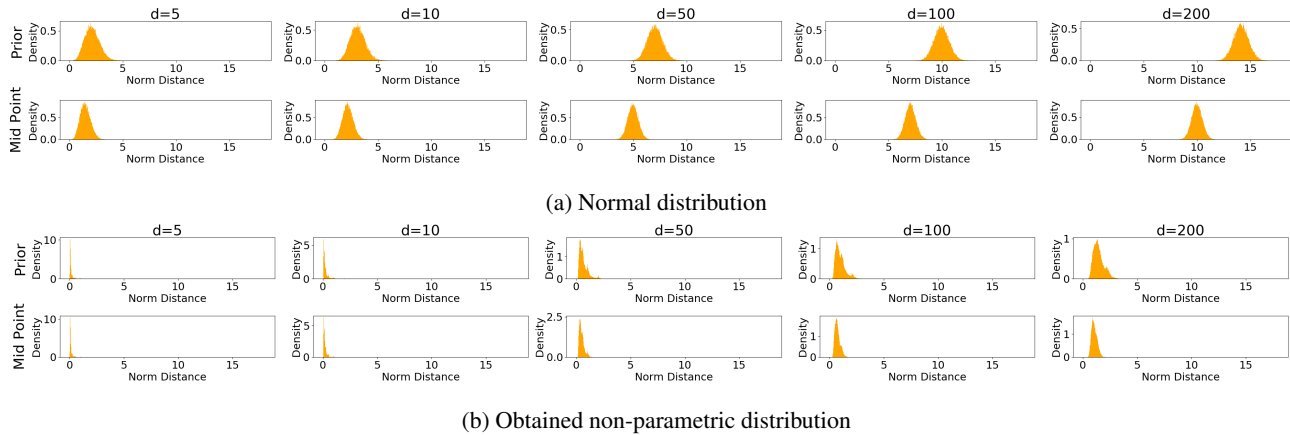


Figure 4. Euclidean norm distribution for samples drawn from different priors and their corresponding mid-point Euclidean norm distribution for different dimensions  $d$ . Note the mid-point norm distribution for the normal prior moves further away from the prior norm distribution as the dimension increases to  $d = 200$ , whereas with our non-parametric prior, the mid-point norm distribution overlaps with the prior norm distribution even at  $d = 200$ .

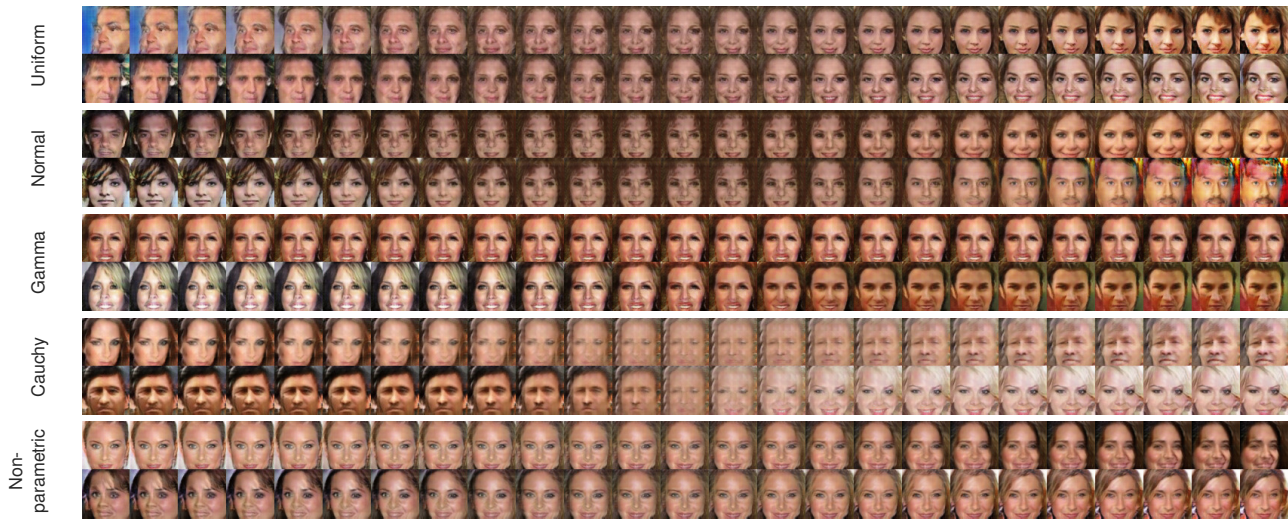


Figure 5. Interpolation (left to right) through the origin on CelebA dataset using different priors with  $d = 100$ . Note the degradation in image quality around the center of the panel (origin space) for many standard priors.

datasets: a) CelebA dataset (Liu et al., 2015), b) CIFAR10 (Krizhevsky & Hinton, 2009), c) LSUN Bedroom, and d) LSUN Kitchen (Yu et al., 2015)). We train our model to the same number of epochs and all the hyper-parameters of the

training are kept same for all the cases. We train each model three times and report the average scores. Details about the network architecture and the training method are provided in the supplemental material. We compare our proposed



Figure 6. Interpolation (left to right) between two random points on LSUN bedroom dataset using different priors with  $d = 100$ .

non-parametric prior distribution against standard ones like the normal, uniform and the priors designed to minimize the mid-point distribution mismatch like Gamma (Kilcher et al., 2018), and Cauchy (Leśniak et al., 2019). For Gamma and Cauchy, we use the same parameters as suggested in the corresponding references.

**Qualitative tests:** In Figure 5, we show the effect of interpolation through origin in high latent-space dimension ( $d = 100$ ) for different priors on CelebA dataset. Here, we interpolate between two random points such that the interpolation line passes through the origin. Similar to (Kilcher et al., 2018), we also observe that with standard priors like the normal, in high latent dimension, the GAN generates non-realistic images around the origin. Note the difference in quality near the images in the center of the panels (space around origin). With our non-parametric distribution obtained from the solution of (7), the GAN generates more realistic images around the origin even at higher dimensions. It was pointed out by Lesniak et. al. (Leśniak et al., 2019) that if a GAN is trained for more epochs, then it learns to fill the space around the origin even with the standard priors. We observe similar trend with the normal and uniform priors. However, with our non-parametric prior the GAN learns to fill the space around the origin very early in training compared with the standard priors. We also present qualitative comparisons on LSUN bedroom dataset in Figure 6: comparing the results with the standard priors and the priors proposed in (Kilcher et al., 2018) and (Leśniak et al., 2019), highlighting the favorable performance of the proposed approach. While we note that it is difficult to perceptually appreciate whether we outperform the other priors, we show that we do obtain competitive visual quality with a conceptually general approach. In the supplemental material, we show additional qualitative results for LSUN

Bedroom/Kitchen and CIFAR10 datasets. We note that the Cauchy distribution had difficulty converging on these datasets, exhibiting possible mode collapse and instability during GAN training.

**Quantitative evaluation:** For quantitative analysis, we use the Inception Score (IS) (Salimans et al., 2016) and the Frechet Inception Distance (FID) (Heusel et al., 2017) which are the standard metrics used to evaluate GAN performance. The inception score *correlates* with the visual quality of the generated image – higher the better. However, recent studies suggest that the inception score does not compare the statistics of the generated dataset with the real-world data (Heusel et al., 2017; Zhang et al., 2018), and thus is not always a reliable indicator of visual quality. This drawback of the IS is overcome by the FID score, which compares the statistics of the generated data with the real data with respect to features. The lower the FID score, the better. We will see in Tables 2 to 6, that our non-parametric prior performs better in terms of FID score on both the prior and mid-point by at least 2 points. In terms of IS we are the best in most of the cases, when we do not perform better, we come quite close to the best performing one.

To get the IS and FID score we sample  $5 \times 10^4$  points from the prior, and estimate the scores on the corresponding image samples. For mid-point, we sample two sets of  $5 \times 10^4$  points from the prior, and an image is generated by the GAN, with the corresponding average points as inputs. Results are summarized in Tables 2 to 6 for different datasets. Table 2 compares our non-parametric prior with other standard priors on the CelebA dataset, at latent space dimension  $d = 100$ . The non-parametric prior outperforms all other priors on both the metrics. Our prior has better FID score by more than 6 points on both the prior and the mid-point.



Table 2. Comparison of IS and FID scores for different prior distributions on CelebA dataset with  $d = 100$

Distribution	Inception Score		FID Score	
	Prior	Mid-Point	Prior	Mid-Point
Uniform	1.843	1.369	24.055	40.371
Normal	1.805	1.371	26.173	42.136
Gamma	1.776	1.618	29.912	28.608
Cauchy	1.625	1.628	59.601	60.128
<b>Non-parametric</b>	<b>1.933</b>	<b>1.681</b>	<b>17.735</b>	<b>19.115</b>

Table 3. Comparison of IS and FID scores for different prior distributions on CelebA dataset with  $d = 200$

Distribution	Inception Score		FID Score	
	Prior	Mid-Point	Prior	Mid-Point
Uniform	1.908	1.407	25.586	44.837
Normal	1.857	1.434	25.035	43.596
Gamma	1.738	1.608	33.816	32.241
Cauchy	1.734	<b>1.743</b>	86.286	86.278
<b>Non-parametric</b>	<b>1.973</b>	1.636	<b>14.953</b>	<b>19.322</b>

As expected the IS and FID scores for the Cauchy is almost the same for the prior and the mid-point. With Gamma, we observe that the score on mid-points is slightly better than the prior since the gamma distribution is highly dense around the origin. In Table 3, we show the IS and FID for the prior and the mid-point at latent space dimension  $d = 200$ . We note that our non-parametric distribution performs better in all the cases compared with all other priors. Scores for our non-parametric prior is almost similar to the scores in Table 2, which indicates its robustness toward the increase in latent space dimension. Cauchy prior sometimes leads to mode collapse during the training which is indicated by its poor FID scores. From Table 2 and 3, we also note that the IS and FID score become worse for the mid-point compared to the prior point as we increase the latent space dimension.

Table 4. Comparison of IS and FID scores for different prior distributions on CIFAR10 dataset with  $d = 100$

Distribution	Inception Score		FID Score	
	Prior	Mid-Point	Prior	Mid-Point
Uniform	6.411	5.204	43.501	76.913
Normal	6.836	5.656	39.235	65.525
Gamma	6.449	6.798	48.334	39.262
Cauchy	2.972	2.964	180.37	180.40
<b>Non-parametric</b>	<b>6.871</b>	<b>6.809</b>	<b>34.803</b>	<b>37.112</b>

Table 4 shows the IS and FID scores for CIFAR10. With our non-parametric prior, the GAN performs better than other priors on both the prior and mid-point by at least 2 points on FID score. Similar to CelebA dataset, we observe the training with Cauchy prior is highly unstable. In Table 5 and 6 we compare our non-parametric prior with other priors on the LSUN bedroom and LSUN kitchen datasets. We observe that our non-parametric prior outperform other priors on the FID score by at least 6 points. We observe that the Gamma and Cauchy priors perform worse on both the prior and mid-point in terms of FID score. These priors often lead to mode collapse during the training. Note that the LSUN

Table 5. Comparison of IS and FID scores with different prior distributions on LSUN Bedroom dataset with  $d = 100$

Distribution	Inception Score		FID Score	
	Prior	Mid-Point	Prior	Mid-Point
Uniform	2.969	2.649	42.998	76.412
Normal	2.812	2.591	64.682	108.49
Gamma	2.930	2.808	162.44	161.37
Cauchy	<b>3.148</b>	<b>3.149</b>	97.057	97.109
<b>Non-parametric</b>	3.028	2.769	<b>27.857</b>	<b>31.472</b>

Table 6. Comparison of IS and FID scores with different prior distributions on LSUN Kitchen dataset with  $d = 100$

Distribution	Inception Score		FID Score	
	Prior	Mid-Point	Prior	Mid-Point
Uniform	2.656	2.549	40.041	51.119
Normal	2.844	2.867	39.909	53.448
Gamma	2.183	2.147	181.81	187.00
Cauchy	1.182	1.179	242.27	242.87
<b>Non-parametric</b>	<b>3.109</b>	<b>3.031</b>	<b>33.194</b>	<b>35.074</b>

dataset has a larger variation in images, and also a larger training-set than the CelebA dataset. The non-parametric prior performs best in both cases as measured by the FID on both prior and mid-point, showing its benefits on large datasets with large variation. **A few salient observations** from the results are:

- The quantitative results on different datasets show that standard priors like the normal and uniform perform better on the prior point but worse on the mid-point.
- The priors proposed to minimize the mid-point distribution mismatch in (Kilcher et al., 2018) and (Leśniak et al., 2019) achieve better results on the mid-point but perform worse on the prior point.
- Gamma and Cauchy do not perform consistently across datasets. In some cases they are the best, but when they are not, their performance can be far from the best.
- The non-parametric distribution is far more consistent, and is either the best, or pretty close to the best performing one, on all four datasets.

## 5. Conclusions

In this paper, we propose a generalized approach to solve distribution mismatch for interpolation in GANs. We show the qualitative and quantitative effectiveness of our approach over the standard priors. We note that often times, our proposed method is in fact the best one, and in cases when it is not, it comes quite close to the best performing technique. Our goal is not necessarily outperform all other GANs, but to suggest the use of non-trivial priors, which might improve image quality without any additional training-data or architectural complexity. Additionally, it would be an interesting avenue of future work to extend this approach to extrapolation problems, or impose other interesting statistical or physically-motivated constraints over latent spaces.



## Acknowledgements

This work was supported by a Intel HVMRC grant. PT was supported by ARO grant number W911NF-17-1-0293. SJ was jointly supported from both the Herberger Research Initiative in the Herberger Institute for Design and the Arts (HIDA) and the Fulton Schools of Engineering (FSE) at Arizona State University.

## References

- Agustsson, E., Sage, A., Timofte, R., and Van Gool, L. Optimal transport maps for distribution preserving operations on latent spaces of generative models. In *International Conference on Learning Representations (ICLR)*, 2019.
- Arjovsky, M. and Bottou, L. Towards principled methods for training generative adversarial networks. In *International Conference on Learning Representations (ICLR)*, 2017.
- Arjovsky, M., Chintala, S., and Bottou, L. Wasserstein gan. In *International Conference on Machine Learning (ICML)*, 2017.
- Dai, B., Fidler, S., Urtasun, R., and Lin, D. Towards diverse and natural image descriptions via a conditional gan. In *IEEE International Conference on Computer Vision (ICCV)*, pp. 2989–2998, 2017.
- Denton, E. L., Chintala, S., Fergus, R., et al. Deep generative image models using a laplacian pyramid of adversarial networks. In *Advances in neural information processing systems*, pp. 1486–1494, 2015.
- Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y. Generative adversarial nets. In *Advances in neural information processing systems*, pp. 2672–2680, 2014.
- Gulrajani, I., Ahmed, F., Arjovsky, M., Dumoulin, V., and Courville, A. C. Improved training of wasserstein gans. In *Advances in Neural Information Processing Systems*, pp. 5767–5777, 2017.
- Hall, P., Marron, J. S., and Neeman, A. Geometric representation of high dimension, low sample size data. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 67(3):427–444, 2005.
- Heusel, M., Ramsauer, H., Unterthiner, T., Nessler, B., and Hochreiter, S. Gans trained by a two time-scale update rule converge to a local nash equilibrium. In *Advances in Neural Information Processing Systems*, pp. 6626–6637, 2017.
- Isola, P., Zhu, J.-Y., Zhou, T., and Efros, A. A. Image-to-image translation with conditional adversarial networks. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5967–5976. IEEE, 2017.
- Karras, T., Aila, T., Laine, S., and Lehtinen, J. Progressive growing of gans for improved quality, stability, and variation. In *International Conference on Learning Representations (ICLR)*, 2018.
- Kilcher, Y., Lucchi, A., and Hofmann, T. Semantic interpolation in implicit models. In *International Conference on Learning Representations (ICLR)*, 2018.
- Krizhevsky, A. and Hinton, G. Learning multiple layers of features from tiny images. Technical report, University of Toronto, 2009.
- Kumar, A. and Chellappa, R. Disentangling 3D Pose in A Dendritic CNN for Unconstrained 2D Face Alignment. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 430–439, 2018.
- Ledig, C., Theis, L., Huszár, F., Caballero, J., Cunningham, A., Acosta, A., Aitken, A. P., Tejani, A., Totz, J., Wang, Z., et al. Photo-realistic single image super-resolution using a generative adversarial network. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, volume 2, pp. 4, 2017.
- Leśniak, D., Sieradzki, I., and Podolak, I. Distribution-interpolation trade off in generative models. In *International Conference on Learning Representations (ICLR)*, 2019.
- Liu, Y., Wei, F., Shao, J., Sheng, L., Yan, J., and Wang, X. Exploring disentangled feature representation beyond face identification. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2080–2089, 2018a.
- Liu, Y., Yeh, Y., Fu, T., Wang, S., Chiu, W., and Wang, Y. F. Detach and Adapt: Learning Cross-Domain Disentangled Deep Representation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 8867–8876, 2018b.
- Liu, Z., Luo, P., Wang, X., and Tang, X. Deep learning face attributes in the wild. In *Proceedings of the IEEE International Conference on Computer Vision*, pp. 3730–3738, 2015.
- Nowozin, S., Cseke, B., and Tomioka, R. f-gan: Training generative neural samplers using variational divergence minimization. In *Advances in Neural Information Processing Systems*, pp. 271–279, 2016.
- Pathak, D., Krahenbuhl, P., Donahue, J., Darrell, T., and Efros, A. A. Context encoders: Feature learning by inpainting. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2536–2544, 2016.

- Radford, A., Metz, L., and Chintala, S. Unsupervised representation learning with deep convolutional generative adversarial networks. In *International Conference on Learning Representations (ICLR)*, 2016.
- Reed, S., Akata, Z., Yan, X., Logeswaran, L., Schiele, B., and Lee, H. Generative adversarial text to image synthesis. In *33rd International Conference on Machine Learning*, pp. 1060–1069, 2016.
- Salimans, T., Goodfellow, I., Zaremba, W., Cheung, V., Radford, A., and Chen, X. Improved techniques for training gans. In *Advances in Neural Information Processing Systems*, pp. 2234–2242, 2016.
- Shrivastava, A., Pfister, T., Tuzel, O., Susskind, J., Wang, W., and Webb, R. Learning from simulated and unsupervised images through adversarial training. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, volume 2, pp. 5, 2017.
- Tzeng, E., Hoffman, J., Saenko, K., and Darrell, T. Adversarial discriminative domain adaptation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, volume 1, pp. 4, 2017.
- White, T. Sampling generative networks. *arXiv preprint arXiv:1609.04468*, 2016.
- Yin, W., Fu, Y., Sigal, L., and Xue, X. Semi-latent GAN: Learning to generate and modify facial images from attributes. *arXiv preprint arXiv:1704.02166*, 2017.
- Yu, F., Seff, A., Zhang, Y., Song, S., Funkhouser, T., and Xiao, J. LSUN: Construction of a large-scale image dataset using deep learning with humans in the loop. *arXiv preprint arXiv:1506.03365*, 2015.
- Zhang, H., Xu, T., Li, H., Zhang, S., Wang, X., Huang, X., and Metaxas, D. N. StackGAN: Text to photo-realistic image synthesis with stacked generative adversarial networks. In *IEEE International Conference on Computer Vision (ICCV)*, pp. 5907–5915, 2017.
- Zhang, H., Goodfellow, I., Metaxas, D., and Odena, A. Self-attention generative adversarial networks. *arXiv preprint arXiv:1805.08318*, 2018.
- Zhu, J.-Y., Krähenbühl, P., Shechtman, E., and Efros, A. A. Generative visual manipulation on the natural image manifold. In *European Conference on Computer Vision (ECCV)*, pp. 597–613. Springer, 2016.
- Zhu, J.-Y., Park, T., Isola, P., and Efros, A. A. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *IEEE International Conference on Computer Vision (ICCV)*, 2017.