SPATIAL FREQUENCY BIAS IN CONVOLUTIONAL GENERATIVE ADVERSARIAL NETWORKS

Anonymous authors
Paper under double-blind review

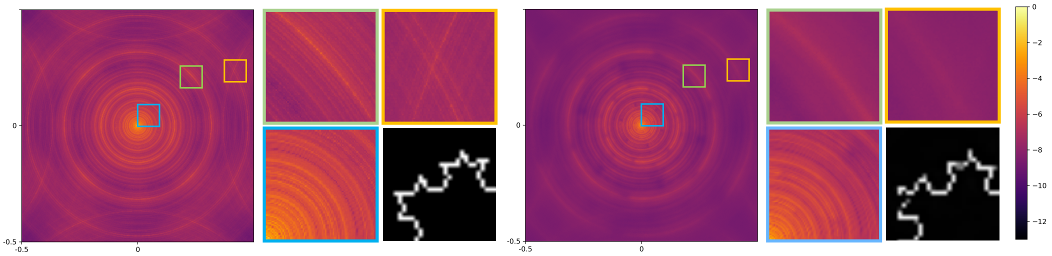
ABSTRACT

As the success of Generative Adversarial Networks (GANs) on natural images quickly propels them into various real-life applications across different domains, it becomes more and more important to clearly understand their limitations. Specifically, understanding GANs' capability across the full spectrum of spatial frequencies, *i.e.* beyond the low-frequency dominant spectrum of natural images, is critical for assessing the reliability of GAN generated data in any detail-sensitive application (*e.g.* denoising, filling and super-resolution in medical and satellite images). In this paper, we show that the ability of convolutional GANs to learn a distribution is significantly affected by the spatial frequency of the underlying carrier signal, that is, GANs have a bias against learning high spatial frequencies. Crucially, we show that this bias is not merely a result of the scarcity of high frequencies in natural images, rather, it is a systemic bias hindering the learning of high frequencies regardless of their prominence in a dataset. Furthermore, we explain why large-scale GANs' ability to generate fine details on natural images does not exclude them from the adverse effects of this bias. Finally, we propose a method for manipulating this bias with minimal computational overhead. This method can be used to explicitly direct computational resources towards any specific spatial frequency of interest in a dataset, extending the flexibility of GANs.

1 INTRODUCTION

The information in an image is carried by a set of spatial frequencies, that is, a set of planar sinusoids with unique frequencies and directions. Intuitively, we associate the high frequencies with the details of an image, and the low frequencies with its general form; however, neither frequencies should be treated as more important by a generative model seeking to learn a distribution. To make this more clear, consider a two-dimensional planar cosine wave defined over a 128×128 image, and assume that we sample the magnitude of this static wave from a Gaussian distribution. Whether this wave completes 64 periods across the image (*i.e.* high spatial frequency), or 3 periods (*i.e.* low spatial frequency), or even no period at all (*i.e.* a constant image), should not affect how well a generative model can learn the underlying Gaussian distribution, otherwise, it may sacrifice the information carried by certain frequencies in favor of learning other frequencies better!

Convolutional Generative Advarsarial Networks (GANs) (Radford et al., 2015; Goodfellow et al., 2014) are the foremost generative models for generating natural image distributions, and while many of their limitations have been studied from the perspective of probability theory and manifold learning (Arjovsky & Bottou, 2017; Arora et al., 2017; Khayatkhoei et al., 2018), their spectral limitations remain mostly unexplored. The theory of GANs (Goodfellow et al., 2014), and its many variants (Arjovsky et al., 2017; Mroueh & Sercu, 2017; Nowozin et al., 2016), do not reveal any spectral limitation. Moreover, the recent success of large-scale GANs in learning fine details in high resolution images seems to support this notion that GANs are not sensitive to the spectral composition (Karras et al., 2020; 2018). In contrast, the progression of GAN research over the recent years reflects a constant effort for generating better *details* while generating *general form and color* seems to be quite easy. Can this apparent difficulty of generating details in practice be linked to an explicit spectral bias? In this work, we first mathematically reveal a spectral bias in the structure of generative convolutional neural networks (CNNs), and then empirically show its manifestation into a spatial frequency bias in GANs. Our findings can help determine which datasets are particularly



True Data

StyleGAN2

Figure 1: Average power spectrum of a large-scale GAN trained on a fractal-based dataset clearly reveals how low frequencies (closer to center) are matched much more accurately than the high frequencies (closer to corners). (Left) Average power spectrum of randomly rotated Koch snowflakes of level 5 and size 1024×1024. (Right) Average power spectrum of StyleGAN2 trained on the latter. A representative patch from the perimeter of true and generated fractals are also displayed.

prone to sub-optimal learning in a GAN training, and perhaps more importantly, which parts of a signal's information are more prone to loss. We summarize our contributions below:

- We present a theorem which quantifies the linear dependency contained in the spectrum of filters used in a generative CNN, and implies a spatial frequency bias (Section 3).
- We show how GANs do not learn high frequencies as well as low frequencies in natural images, through the lens of a feature-based metric with spectral resolution (Section 3.1).
- We show the stark sensitivity of GANs' performance to the spatial frequency of the signal that carries a distribution, providing further evidence for the bias (Section 3.2).
- We explain why large-scale GANs' ability to generate very fine details on natural images does not exclude them from the adverse effects of the spatial frequency bias (Section 3.3).
- Finally, we propose an approach for manipulating the spatial frequency bias and show its effectiveness in enhancing GANs' performance on high spatial frequencies (Section 4).

2 SPATIAL FREQUENCY COMPONENTS

According to Inverse Discrete Fourier Transform (Gonzalez et al., 2002), every periodic discrete 2D signal $I(x, y)$ with $x \in \{0, 1, 2, \ldots, m-1\}$ and $y \in \{0, 1, 2, \ldots n-1\}$, can be written as a sum of several complex sinusoids as follows:

$$I(x,y) = \frac{1}{mn} \sum_{u=0}^{m-1} \sum_{v=0}^{n-1} C(u,v) e^{j2\pi(\frac{ux}{m} + \frac{vy}{n})} = \frac{1}{mn} \sum_{u=0}^{m-1} \sum_{v=0}^{n-1} C(u,v) e^{j2\pi(\hat{u},\hat{v}).(x,y)} \tag{1}$$

We denote each complex sinusoid a *spatial frequency component* which can be expressed by a vector (u, v) over the pixel locations (x, y). In the above equation, $C(u, v)$ is the complex amplitude of each frequency component, $(\hat{u}, \hat{v}) = (\frac{u}{m}, \frac{v}{n})$ defines the direction of propagation on the 2D plane and its magnitude defines the spatial frequency in that direction, and $m, n \in \mathbb{N}$ are the periods of I in x and y direction respectively. Every channel of a digital 2D image can be assumed periodic beyond the image boundaries, and therefore represented by Eq. (1), with periods m and n being the length and width of the image respectively. In that case, the vector $(\hat{u}, \hat{v})$ would define the spatial frequency of a sinusoid in terms of *cycles per pixel*, in x and y direction respectively, with $\hat{u}, \hat{v} \in [0, 1)$. The maximum frequency in each direction is 0.5 corresponding to the Nyquist frequency (the shortest period needs at least two pixels to be represented, hence the maximum frequency is half cycle per pixel). In favor of clarity, and without loss of generality, we will assume $\hat{u}, \hat{v} \in [-0.5, 0.5)$ throughout this paper. Additionally, we loosely refer to the spatial frequency components with $|\hat{u}|$ or $|\hat{v}|$ close to 0.5 as high frequencies, and with $\hat{u}$ or $\hat{v}$ close to 0 as low frequencies. Whenever displaying power spectrums $|C(\hat{u}, \hat{v})|^2$, for better visualization, we drop the dc power, apply Hann window, normalize by the maximum power, and apply $\log$, such that the most powerful frequency always has value 0. Also, $\hat{u}$ and $\hat{v}$ are placed on horizontal and vertical axes respectively, such that low frequencies are placed close to the center, while high frequencies close to the corners.

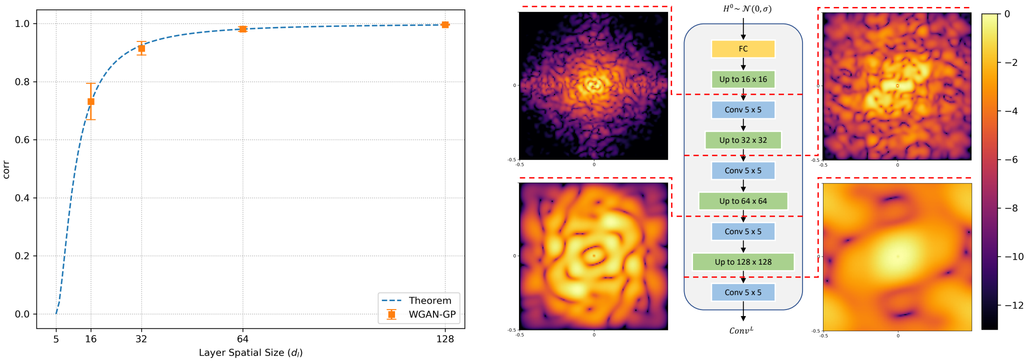


Figure 2: (Left) Dashed blue line shows the predicted correlation between diagonally adjacent spatial frequencies in Eq. (5) for different layers (d_l) given a fixed filter size ($k_l = 5$), and the orange points show its empirical evaluation (average correlation with one standard deviation computed over the filters of WGAN-GP trained on CelebA). (Right) The impulse response spectrum of effective filters operating on each layer in WGAN-GP trained on CelebA. Notice how the spectrum of the effective filters that operate on inner layers, *i.e.* control the generation of low frequencies (top row), are much sharper than the ones that operate on outer layers, *i.e.* control the generation of high frequencies (bottom row). Smoothness is an indication of larger correlation in the effective filters.

3 THE SPATIAL FREQUENCY BIAS

We start by making an observation regarding the structure of a generative convolutional neural network (CNN) with regard to generating spatial frequencies. A CNN generating a 2D signal $I \in \mathbb{R}^{d \times d}$ can be written as a series of layers: $\text{Conv}^L \circ \text{Conv}^{L-1} \cdots \circ \text{Conv}^1(H^1)$, with each Conv layer: [1]

$$\text{Conv}_i^l(H^l) = \sigma\left(\sum_c F_{ic}^l * \text{Up}(H_c^l)\right) \tag{2}$$

where $\text{Conv}_i^l \in \mathbb{R}^{d_l \times d_l}$ denotes the i-th channel in the output of the l-th Conv layer, $F_{ic}^l \in \mathbb{R}^{k_l \times k_l}$ is the 2D filter in the c-th channel of the i-th Conv filter of the layer, H_c^l is the c-th channel of the input to the layer, $\text{Up}(.)$ is the upsampling operator, and $\sigma(.)$ is a non-linearity. If we restrict σ to rectified linear units (ReLU), then the CNN's output space will become piece-wise linear, that is, in a small neighborhood of H^1 we can assume the CNN is linear end-to-end and therefore drop $\sigma(.)$ in Eq. (2). This means we are restricting our analysis to the locality of any single sample. Furthermore, since $\text{Up}(.)$ is a linear shift-invariant operation, we can absorb it into H^l (note that we are not removing Up, we are now considering it as a pre-processing operation on the input to each layer), reaching at:

$$\text{Conv}_i^l(H^l) = \sum_c F_{ic}^l * H_c^l \tag{3}$$

Next, in the following theorem, we will show how the filter size k_l and the spatial dimension d_l of a Conv layer affect the correlation in the spectrum of its filters. Note that more correlation in a filter's spectrum means more linear dependency, and thus reduces its effective capacity, in other words, the filter can not freely adjust specific frequencies without affecting the adjacent correlated frequencies.

Theorem 1. *Let $U = \mathcal{F}\left\{F_{ic}^l\right\}(u_0, v_0)$ and $V = \mathcal{F}\left\{F_{ic}^l\right\}(u_1, v_1)$ be any two spatial frequency components on the spectrum of any 2D filter of the l-th Conv layer with spatial dimension d_l and filter size k_l. Assuming i.i.d. weight initialization, the magnitude of complex correlation coefficient between U and V is given by* [2]*:*

$$\begin{aligned} |corr(U,V)| &= \left|\frac{Sinc(u_0 - u_1, v_0 - v_1)}{k_l^2}\right| \\ s.t. \quad Sinc(u,v) &= \frac{\sin(\frac{\pi u k_l}{d_l})\sin(\frac{\pi v k_l}{d_l})}{\sin(\frac{\pi u}{d_l})\sin(\frac{\pi v}{d_l})} \quad d_l, k_l \in \mathbb{N} \quad and \quad 1 < k_l \le d_l \end{aligned} \tag{4}$$

---

[1] Transposed Conv layers are sufficiently represented by an appropriate choice of the Up(.) operator.

[2] Proof in Appendix.

Corollary 1.1. *If U and V are two diagonally adjacent spatial frequency components of F^l_{ic}, then:*

$$|corr(U,V)| = \frac{\sin(\frac{\pi k_l}{d_l})^2}{k_l^2 \sin(\frac{\pi}{d_l})^2} \tag{5}$$

Now, in each Conv layer, note that the maximum spatial frequency that can be generated is limited by the Nyquist frequency, that is, Conv^l can only adjust image spatial frequencies in $[0, \frac{d_l}{2d}]$ without aliasing[3]. This means that high frequencies are primarily generated by the CNN's outer layers which have larger spatial dimension d_l. According to Eq. (5), given a fixed filter size k_l, the larger the d_l, the larger the correlation in the filter's spectrum (see the curve in Figure 2), and consequently the smaller its effective capacity. Therefore, the outer layers responsible for generating high frequencies are more restricted in their spectrum compared to the inner layers with smaller d_l. Additionally, note that while only outer layers can generate high frequencies without aliasing, the low frequencies can be generated by all layers without aliasing. Therefore, the spatial size of the effective filter operating on low frequencies will always be larger than that operating on high frequencies. That means, even if larger filter size k_l is used in the outer layers to counteract the effect of larger d_l, the low frequencies will still be enjoying a larger end-to-end filter size compared to high frequencies, hence less correlation (see the spectrums of effective filters in Figure 2).

In light of these observations, we hypothesize that convolutional GANs contain a systemic bias against generating high frequencies. In the remainder of this section we will empirically investigate this hypothesis on three popular convolutional GANs: WGAN-GP (Gulrajani et al., 2017) serves as a simple but fundamental GAN model; and Progressively Growing GAN (PG-GAN) (Karras et al., 2018) and StyleGAN2 (Karras et al., 2020) serve as state-of-the-art models with large capacity and complex structure, incorporating state-of-the-art normalization and regularization techniques. In order to have an approximately consistent CNN structure, in all models, the generative CNNs start from a 512 dimensional latent space, and after the 8×8 spatial dimensions, the channel size is consecutively halved every time the spatial dimensions are doubled. See Appendix for more details.

3.1 FID LEVELS

Direct comparison between the average power spectrums of generated and true images is not a very informative measure of spectral performance, since a numerical difference in average power spectrums does not necessarily mean a meaningful difference in discernible image features. Instead, we propose an extended version of Frechet Inception Distance (FID) (Heusel et al., 2017), denoted *FID Levels*, in which we plot FID between two sets of images after gradually removing low frequency bands from both sets. The intuition is to observe the change in the GAN's performance as frequency bands are gradually removed. FID measures the differences between the features extracted from two image distributions: the larger the FID, the larger the mismatch between the two distributions. We chose FID because of its sensitivity to a wide range of high and low frequency features (Borji, 2019; Lucic et al., 2018). Each point on the FID Levels plot shows FID computed after applying a high-pass Gaussian filter, with the cut-off specified on the horizontal axis, to both the generated and the true images. As a baseline for comparison, we also compute FID Levels between two disjoint subsets of the true images, denoted *True FID Levels*, demonstrating the ideal GANs' performance. For completeness, as a direct measure of differences between power spectrums, we also report *Leakage Ratio (LR)* in our experiments, which computes total variation (Gibbs & Su, 2002) between the generated and true average power spectrums normalized into density functions. Intuitively, LR shows what percentage of the power has leaked away between two power spectrums.

Removing frequency bands removes information from both image sets, and at the limit, where all the spatial frequencies are removed, the two image sets would look exactly the same, therefore, we expect a declining plot in FID Levels overall. However, if the GAN has matched certain frequency bands better than the others, removing those bands from images would cause an increase in FID. Figure 3 shows FID Levels of GANs trained on two 128×128 image datasets: CelebA (Liu et al., 2015) and LSUN-Bedrooms (Yu et al., 2015). The GANs exhibit an increase in FID Levels on both

---

[3] Aliasing here refers to the process of generating high frequencies by replicating low frequencies in the expanded spectrum introduced through up-sampling. Since this makes duplicates in several high frequency bands of the spectrum, it's ability to control high frequencies is minimal. Many up-sampling approaches, such as bi-linear or nearest-neighbors, explicitly attenuate such aliased frequencies.

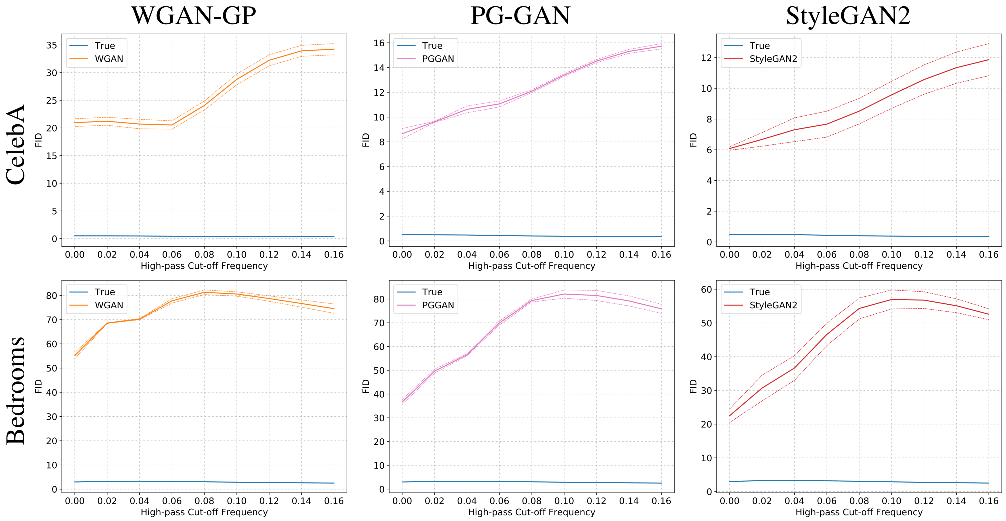


Figure 3: FID Levels of GANs trained on CelebA and LSUN-Bedrooms. The farther to the right on the horizontal axis, the more low frequencies are removed prior to FID computation. Notice the transient increase in FID (worsening performance) as low frequencies are removed. All plots will eventually decline to zero if we continue removing frequencies. In all figures, the *blue curve* depicts the True FID Levels of the corresponding dataset as a baseline. All figures show average FID with one standard deviation error bars (*dashed line*), over three random training runs.

datasets, which shows that removing low frequencies has emphasized the mismatch between the true and the generated images, hence a bias against learning high frequencies. Without such a bias, we would only observe a decline in FID as low frequencies are gradually removed. However, since much of the information in natural images is concentrated at low frequencies (Field, 1987), this bias might be attributed to the scarcity of high frequencies during training. We will show that this point is not valid in the following subsection.

3.2 HIGH FREQUENCY DATASETS

If GANs are not biased against high frequencies, their performance should remain indifferent to shifting the frequency contents of the datasets. In other words, whether the information in a dataset is primarily carried by high frequencies or low frequencies should not affect how well GANs can learn the underlying image distribution. In order to test this hypothesis, we can create high frequency versions of CelebA and LSUN-Bedrooms by applying a frequency shift operator, that is, multiplying every image in each dataset with $\cos(\pi(x + y))$ channel-wise, to create Shifted CelebA (SCelebA) and Shifted LSUN-Bedrooms (SBedrooms) respectively. In effect, all we are doing is swapping the low and high frequency contents of these datasets. Note that the frequency shift operator is a homeomorphism and therefore the distributions of SCelebA and SBedrooms have the same topological properties as CelebA's and LSUN-Bedroom's, and therefore the GANs' performance should remain unchanged from a purely probabilistic perspective.

Table 1 compares the GANs' performance on SCelebA and SBedrooms versus the original CelebA and LSUN-Bedrooms.[4] Their performance has worsened significantly (larger FID and LR) on the high frequency datasets, showing that the GANs perform considerably better when the same image distribution is carried primarily by low frequencies. This observation rejects our earlier hypothesis, that is, the GANs' performance is not indifferent to frequency shift. Additionally, this shows that the bias against high frequencies we observed in Section 3.1 cannot be explained by the scarcity of high frequencies in natural images: even though the unbalancedness in the distribution of power has remained unchanged in the high frequency versions of the datasets, the GANs' performance has

---

[4] In SCelebA and SBedrooms, true and generated images are re-shifted before computing FID so that the values are comparable with the FID results on CelebA and LSUN-Bedrooms

Table 1: Performance drop (increase in both FID and LR) when GANs are trained on the high frequency versions of CelebA and LSUN-Bedrooms, denoted SCelebA and SBedrooms respectively. Average measures with standard deviation (sd) are reported over three random training runs.

Model	Measure	CelebA	SCelebA	Bedrooms	SBedrooms
WGAN-GP	FID	20.97 sd 0.70	328.72 sd 9.70	55.14 sd 1.29	283.02 sd 7.06
	LR (%)	2.29 sd 0.31	59.04 sd 5.09	1.99 sd 0.48	42.42 sd 4.32
PG-GAN	FID	8.66 sd 0.41	23.12 sd 2.08	36.65 sd 0.97	69.03 sd 10.28
	LR (%)	1.06 sd 0.21	3.93 sd 0.70	1.51 sd 0.29	3.12 sd 0.16
StyleGAN2	FID	6.08 sd 0.11	343.57 sd 53.59	22.49 sd 2.00	260.84 sd 4.03
	LR (%)	1.55 sd 0.42	43.03 sd 34.10	1.28 sd 0.19	7.32 sd 1.86

worsened significantly. We conclude that this bias is indeed a "spatial frequency bias" against high frequencies, regardless of how abundant or scarce they are in the dataset.

3.3 A HIGH RESOLUTION ESCAPE

It is key to note that whether a signal contains high spatial frequencies or not is directly related to its sampling rate. For example, consider the continuous-valued image of a bird formed on a camera's sensor, whose feathers change color from white to black 64 times over the length of the image. If this image is sampled into a 128×128 picture, the feathers would form a high frequency ($\frac{1}{2}$ cycles per pixel). If we instead sample the same image into a 1024×1024 picture, the feathers, still changing color 64 times over the length of the image, now create pixel value changes every 16 pixels, forming a low frequency ($\frac{1}{16}$ cycles per pixel). Therefore, one solution to the spatial frequency bias is to simply use data at a very high resolution such that no high frequency component remains. We hypothesize that this can partly explain the success of large-scale GANs at generating fine details in natural image datasets (Karras et al., 2020; 2018; Brock et al., 2019). These models, by introducing novel techniques, are able to train very large networks on high resolution images, in which details are no longer "fine" as far as the generator is concerned. However, this does not mean that they perform similarly on all frequencies and do not suffer from the spatial frequency bias. In Figure 1, we show that a large-scale StyleGAN2, when trained on a dataset that does contain high frequencies, clearly still exhibits the signs of a bias against high frequencies. Thus, sampling at a high resolution is essentially escaping the spatial frequency bias, not solving the bias, and such an escape is not always practical. Training GANs on higher resolution images requires exponentially more training resources, which can quickly make the training impractical if the data inherently contains very high frequencies. Therefore, it is important to explore alternative approaches for dealing with the bias.

4 FREQUENCY SHIFTED GENERATORS (FSG)

In section 3, we observed that GANs have a spatial frequency bias, favoring the learning of low frequencies, however, *is it possible to manipulate this bias such that it favors other frequencies?* If so, this would make it possible to explicitly target specific frequencies of interest in a dataset. In this section, we show how this can be achieved without any increase in training resources. Instead of inherently generating high frequencies, a generative model $G(x, y)$ can first generate a signal with prominent low frequencies and then transform the signal such that these prominent frequencies are shifted towards a desired frequency $(\hat{u}_t, \hat{v}_t)$. This can be achieved by a frequency shift operator:

$$G(x,y)e^{j2\pi(\hat{u}_t x+\hat{v}_t y)} = \frac{1}{mn}\sum_{u=0}^{m-1}\sum_{v=0}^{n-1} C(u,v)e^{j2\pi(\hat{u}+\hat{u}_t,\hat{v}+\hat{v}_t).(x,y)} \tag{6}$$

where $\hat{u}_t, \hat{v}_t \in [-0.5, 0.5)$. After the frequency shift, the frequency components previously close to $(0, 0)$ are now placed close to $(\hat{u}_t, \hat{v}_t)$. However, since G is generating a real signal and the spectrum of real signals are symmetric, it can not sufficiently represent a high frequency band, that is, G can

Table 2: Performance gain (decrease in both FID and LR) on the high frequency datasets achieved by using FSG in GANs. Average measures with sd are reported over three random training runs.

	SCelebA		SBedrooms	
Model	FID	LR (%)	FID	LR (%)
WGAN-GP	328.72 sd 9.70	59.04 sd 5.09	283.02 sd 7.06	42.42 sd 4.32
WGAN-FSG	20.70 sd 0.44	1.93 sd 0.57	59.81 sd 1.64	1.80 sd 0.28
PG-GAN	23.12 sd 2.08	3.93 sd 0.70	69.03 sd 10.28	3.12 sd 0.16
PG-GAN-FSG	17.91 sd 0.74	2.96 sd 0.55	54.64 sd 0.26	2.67 sd 0.75
StyleGAN2	343.57 sd 53.59	43.03 sd 34.10	260.84 sd 4.03	7.32 sd 1.86
StyleGAN2-FSG	7.17 sd 0.07	1.41 sd 0.10	67.85 sd 2.38	1.82 sd 0.27

only represent symmetric frequency bands. Note that while natural images are real signals and have symmetric spectrum with respect to zero, a specific band of their spectrum is not necessarily symmetric. In order to generate a non-symmetric frequency band, we can use two neural networks to generate a real image (G_r) and an imaginary image (G_i), which together compose the complex generated image (G_c). The complex image is then shifted to $(\hat{u}_t, \hat{v}_t)$ according to Eq. (6) to construct the shifted generator G_s as follows:

$$G_s(x, y) = G_c(x, y)e^{j2\pi(\hat{u}_t x+\hat{v}_t y)} = [G_r(x, y) + jG_i(x, y)]\, e^{j2\pi(\hat{u}_t x+\hat{v}_t y)} \tag{7}$$

The real part of G_s is now generating an image which can sufficiently represent any frequency band, and has a spatial frequency bias favoring the desired component $(\hat{u}_t, \hat{v}_t)$:

$$\Re\left[G_s(x, y)\right] = G_r(x, y)\cos(2\pi(\hat{u}_t x + \hat{v}_t y)) - G_i(x, y)\sin(2\pi(\hat{u}_t x + \hat{v}_t y)) \tag{8}$$

Frequency Shifted Generators (FSGs) can be used to efficiently target specific spatial frequency components in a dataset. Table 2 shows the results of training GANs using FSG with $(\hat{u}_t, \hat{v}_t) = (\frac{1}{2}, \frac{1}{2})$ on SCelebA and SBedrooms. The use of FSG has considerably improved the GANs' performance on these high frequency datasets, without any increase in training resources. This also provides an interesting insight: the discriminator is able to effectively guide a capable generator towards learning high frequencies, therefore, the spatial frequency bias must be primarily rooted in the GAN's generator and not the discriminator. Moreover, multiple shifted generators, with smaller network capacity, can be added to the main generator of GANs to improve performance on specific frequencies (*e.g.* see Figure 4). Interestingly, the added FSGs specialize towards their respective target frequency $(\hat{u}_t, \hat{v}_t)$, without any explicit supervision during training. This provides further evidence of the spatial frequency bias: if unbiased, the added FSGs would have no incentive to specialize towards any specific frequency.

5 RELATED WORKS

Spectral Theorems of Neural Networks. Rahaman et al. (2019) have recently studied the spectral properties of fully connected neural networks that use ReLU activations, and discovered a bias against learning high frequencies in a regression task. However, while Rahaman *et al.* define a frequency component as a periodic change in a single output of the network with respect to changes in the input space, we define a frequency component as a periodic change across the adjacent outputs of the network. Note that these two notions of frequency are independent by definition, that is, one notion can be mathematically defined while the other is not and vice versa (by restricting the input or output space to a single point), therefore a bias in one can not imply a bias in the other.

Spectral Limitations. Spectral limitations have been observed in different tasks when using generative CNNs. In L2-reconstruction tasks, there is a bias against low-power frequencies (which coincide with high frequencies in natural images) primarily due to the vanishing gradient of the L2 loss near zero, which is often counteracted by emphasizing the low-power frequencies through additional loss terms (Deng et al., 2020; Li & Barbastathis, 2018; Ulyanov et al., 2018). In Auto

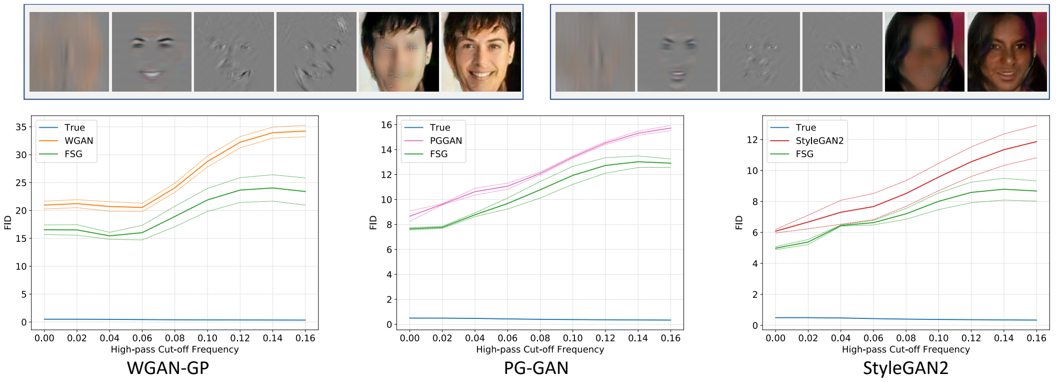


Figure 4: (Top) Two samples from WGAN-GP when enhanced by adding multiple FSGs. In each sample, from left to right, the outputs correspond to the FSG with $(\hat{u}_t, \hat{v}_t)$ at $(\frac{1}{16}, 0)$, $(0, \frac{1}{16})$, $(-\frac{1}{16}, \frac{1}{16})$, and $(\frac{1}{16}, \frac{1}{16})$, the WGAN-GP's main generator, and the final compound output (sum of all the preceding generators). Notice how each FSG has learned to focus on specific spatial frequencies, without any explicit supervision during training. See Appendix for more samples. (Bottom) Improvement in the FID Levels of GANs when enhanced by multiple FSGs, trained on CelebA.

Encoders and Variational Auto Encoders (VAEs), there is a similar bias, primarily due to the distribution assumptions in their objective (Huang et al., 2018; Larsen et al., 2016). In contrast, GAN's objective does not impose any such spectral limitations. In theory, GANs must be able to learn any suitable distribution regardless of the carrier spatial frequencies. Therefore, while a systemic spectral bias in generative CNNs could be obscured by the inherent spectral biases of AEs, VAEs and L2 tasks, the GANs provide a clear lens for observing such biases. Most relevant to our work, two concurrent works (Dzanic & Witherden, 2019; Durall et al., 2020) have recently shown that the discrepancies in high frequencies can be utilized to easily distinguish GAN generated images from natural images. Durall et al. (2020) connect the discrepancies to the aliasing introduced by the up-sampling operator in the expanded spectrum. However, note that if the up-sampling does not introduce aliased power in the expanded frequency spectrum, the following layer will not have any power to filter-out and shape the desired higher frequencies. In contrast, in Theorem 1, we show that it is the linear dependencies in the Conv filter's spectrum that causes spectral limitations, and explain why this affects high frequencies more severely than low frequencies, causing a systemic bias against high frequencies, not just artifacts in the power spectrum.

Quantitative Metrics. The prevalent metrics for evaluating GANs, most notably Inception Score (Salimans et al., 2016), FID (Heusel et al., 2017), and MS-SSIM (Odena et al., 2017), consider all spatial frequency components at once when comparing GAN generated and true images, thus lacking any spectral resolution. Most relevant to our proposed metric, Karras et al. (2018) propose computing sliced Wasserstein distance between patches extracted from true and generated images at different levels of a Laplacian pyramid (SWD). Interestingly, evaluating GANs with SWD shows approximately similar performance across frequency bands (Karras et al., 2018). We conjecture that this discrepancy comes from the fact that small differences between patches in the pixel space, can result in large differences in the more meaningful feature space used by FID. As such, FID Levels is a more informative measure of GANs' performance across spatial frequencies.

6 DISCUSSION

In this work, we identified a bias against high spatial frequencies in convolutional GANs, and then proposed an approach for manipulating this bias and reducing its adverse effects. Being aware of this bias is of paramount importance when using GANs in applications concerned with intricate patterns, *e.g.* de-nosing and reconstruction in medical or satellite imaging. Moreover, our findings suggest that the information carried by high frequencies is much more likely to be missed by GANs, a critical consideration when using GANs for data-augmentation or in semi-supervised learning. We also observed that the spatial frequency bias primarily affects GAN's generator and not its discriminator. This gives the discriminator an advantage which can be the root of certain instabilities in GAN training. Investigating this connection between the spatial frequency bias and unstable GAN training, as well as extending Theorem 1 to incorporate the effect of various normalizations and stabilization techniques, are interesting directions for future research. Finally, we would like to emphasize that while FSG exposes certain behaviors of the spatial frequency bias, and can be used to match the bias of GANs to the known biases of a dataset, it is not a proper solution to the bias, in that it does not remove the bias, rather exploits it. Devising proper solutions for the bias is yet another exciting and important direction for future research.

REFERENCES

Martin Arjovsky and Léon Bottou. Towards principled methods for training generative adversarial networks. *arXiv preprint arXiv:1701.04862*, 2017.

Martin Arjovsky, Soumith Chintala, and Léon Bottou. Wasserstein generative adversarial networks. In *International Conference on Machine Learning*, pp. 214–223, 2017.

Sanjeev Arora, Rong Ge, Yingyu Liang, Tengyu Ma, and Yi Zhang. Generalization and equilibrium in generative adversarial nets (gans). In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pp. 224–232. JMLR. org, 2017.

Ali Borji. Pros and cons of gan evaluation measures. *Computer Vision and Image Understanding*, 179:41–65, 2019.

Andrew Brock, Jeff Donahue, and Karen Simonyan. Large scale GAN training for high fidelity natural image synthesis. In *International Conference on Learning Representations*, 2019.

Mo Deng, Shuai Li, Alexandre Goy, Iksung Kang, and George Barbastathis. Learning to synthesize: robust phase retrieval at low photon counts. *Light: Science & Applications*, 9(1):1–16, 2020.

Ricard Durall, Margret Keuper, and Janis Keuper. Watch your up-convolution: Cnn based generative deep neural networks are failing to reproduce spectral distributions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7890–7899, 2020.

Tarik Dzanic and Freddie Witherden. Fourier spectrum discrepancies in deep network generated images. *arXiv preprint arXiv:1911.06465*, 2019.

David J Field. Relations between the statistics of natural images and the response properties of cortical cells. *Josa a*, 4(12):2379–2394, 1987.

Alison L Gibbs and Francis Edward Su. On choosing and bounding probability metrics. *International statistical review*, 70(3):419–435, 2002.

Rafael C Gonzalez, Richard E Woods, et al. Digital image processing, 2002.

Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in neural information processing systems*, pp. 2672–2680, 2014.

Ishaan Gulrajani, Faruk Ahmed, Martin Arjovsky, Vincent Dumoulin, and Aaron C Courville. Improved training of wasserstein gans. In *Advances in Neural Information Processing Systems*, pp. 5769–5779, 2017.

Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. 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.

Huaibo Huang, Ran He, Zhenan Sun, Tieniu Tan, et al. Introvae: Introspective variational autoencoders for photographic image synthesis. In *Advances in neural information processing systems*, pp. 52–63, 2018.

Tero Karras, Timo Aila, Samuli Laine, and Jaakko Lehtinen. Progressive growing of GANs for improved quality, stability, and variation. In *International Conference on Learning Representations*, 2018.

Tero Karras, Samuli Laine, Miika Aittala, Janne Hellsten, Jaakko Lehtinen, and Timo Aila. Analyzing and improving the image quality of stylegan. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 8110–8119, 2020.

Mahyar Khayatkhoei, Maneesh K Singh, and Ahmed Elgammal. Disconnected manifold learning for generative adversarial networks. In *Advances in Neural Information Processing Systems*, pp. 7343–7353, 2018.

Anders Boesen Lindbo Larsen, Søren Kaae Sønderby, Hugo Larochelle, and Ole Winther. Autoencoding beyond pixels using a learned similarity metric. In *International conference on machine learning*, pp. 1558–1566. PMLR, 2016.

Shuai Li and George Barbastathis. Spectral pre-modulation of training examples enhances the spatial resolution of the phase extraction neural network (phenn). *Optics express*, 26(22):29340–29352, 2018.

Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Deep learning face attributes in the wild. In *Proceedings of International Conference on Computer Vision (ICCV)*, December 2015.

Mario Lucic, Karol Kurach, Marcin Michalski, Sylvain Gelly, and Olivier Bousquet. Are gans created equal? a large-scale study. In *Advances in neural information processing systems*, pp. 700–709, 2018.

Youssef Mroueh and Tom Sercu. Fisher gan. In *Advances in Neural Information Processing Systems*, pp. 2513–2523, 2017.

Sebastian Nowozin, Botond Cseke, and Ryota Tomioka. f-gan: Training generative neural samplers using variational divergence minimization. In *Advances in neural information processing systems*, pp. 271–279, 2016.

Augustus Odena, Christopher Olah, and Jonathon Shlens. Conditional image synthesis with auxiliary classifier gans. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pp. 2642–2651. JMLR. org, 2017.

Kun L. Park. *Fundamentals of Probability and Stochastic Processes with Applications to Communications*. Springer, 2018.

Alec Radford, Luke Metz, and Soumith Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks. *arXiv preprint arXiv:1511.06434*, 2015.

Nasim Rahaman, Aristide Baratin, Devansh Arpit, Felix Draxler, Min Lin, Fred Hamprecht, Yoshua Bengio, and Aaron Courville. On the spectral bias of neural networks. In Kamalika Chaudhuri and Ruslan Salakhutdinov (eds.), *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pp. 5301–5310, Long Beach, California, USA, 09–15 Jun 2019. PMLR.

Tim Salimans, Ian Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford, and Xi Chen. Improved techniques for training gans. In *Advances in neural information processing systems*, pp. 2234–2242, 2016.

Dmitry Ulyanov, Andrea Vedaldi, and Victor Lempitsky. Deep image prior. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 9446–9454, 2018.

Fisher Yu, Yinda Zhang, Shuran Song, Ari Seff, and Jianxiong Xiao. Lsun: Construction of a large-scale image dataset using deep learning with humans in the loop. *arXiv preprint arXiv:1506.03365*, 2015.

APPENDIX A PROOF OF THEOREM 1

Proof. Let us consider a single 2D filter $F^l \in \mathbb{R}^{k_l \times k_l}$ in the l-th Conv layer. Since the spatial dimension of the layer's output is d_l, a filter $G^l \in \mathbb{R}^{d_l \times d_l}$ can sufficiently model any spectrum in the layer's output space. So we can write F^l as a restriction of G^l, that is, the multiplication of G^l with the 2D pulse $P \in \mathbb{R}^{d_l \times d_l}$ of area k_l^2:

$$F^l = P.G^l \tag{9}$$

$$P(x, y) = \begin{cases} 1 & 0 \leq x, y < k_l \\ 0 & \text{otherwise} \end{cases} \tag{10}$$

Therefore, by convolution theorem, the spectrum of F^l is equal to the spectrum of G^l convolved by the spectrum of P:

$$\mathcal{F}\left\{F^l\right\} = \mathcal{F}\left\{P.G^l\right\} = \mathcal{F}\left\{P\right\} * \mathcal{F}\left\{G^l\right\} = \text{Sinc} * \mathcal{F}\left\{G^l\right\} \tag{11}$$

where $\mathcal{F}\{.\}$ denotes the d_l-point DFT, $*$ is the circular convolution, and Sinc is the aliased sinc function:

$$\text{Sinc}(u, v) = \frac{\sin(\frac{\pi u k_l}{d_l}) \sin(\frac{\pi v k_l}{d_l})}{\sin(\frac{\pi u}{d_l}) \sin(\frac{\pi v}{d_l})} e^{-j\pi(u+v)\frac{(k_l-1)}{d_l}} \quad d_l, k_l \in \mathbb{N} \quad \text{and} \quad 1 < k_l \leq d_l \tag{12}$$

Now, let $U = \mathcal{F}\left\{F^l\right\}(u_0, v_0)$ and $V = \mathcal{F}\left\{F^l\right\}(u_1, v_1)$ be two spatial frequency components on the spectrum of the filter, located at the $(\frac{u_0}{d_l}, \frac{v_0}{d_l})$ and $(\frac{u_1}{d_l}, \frac{v_1}{d_l})$ frequencies respectively. We are interested in the correlation between these two components. Note that we are defining G^l as the hypothetical unrestricted filter, such that any possible filter that can be learnt during training becomes a restriction of G^l. Therefore, without loss of generality, we can assume G^l has independent spatial frequency components with variance σ^2. Considering these assumptions, we can find the complex correlation coefficient (Park, 2018) between U and V as follows:

$$\text{Cov}\left[U, V\right] = \text{Cov}\left[\text{Sinc} * \mathcal{F}\left\{G^l\right\}(u_0, v_0), \text{Sinc} * \mathcal{F}\left\{G^l\right\}(u_1, v_1)\right] \tag{13}$$

$$= \text{Cov}\left[\sum_{u,v} \text{Sinc}(u, v)\mathcal{F}\left\{G^l\right\}(u_0 - u, v_0 - v), \sum_{\hat{u},\hat{v}} \text{Sinc}(\hat{u}, \hat{v}).\mathcal{F}\left\{G^l\right\}(u_1 - \hat{u}, v_1 - \hat{v})\right] \tag{14}$$

$$= \sum_{u,v}\sum_{\hat{u},\hat{v}} \text{Sinc}(u, v)\text{Sinc}^*(\hat{u}, \hat{v})\text{Cov}\left[\mathcal{F}\left\{G^l\right\}(u_0 - u, v_0 - v), \mathcal{F}\left\{G^l\right\}(u_1 - \hat{u}, v_1 - \hat{v})\right] \tag{15}$$

$$= \sum_{u,v} \text{Sinc}(u, v)\text{Sinc}(u_0 - u_1 - u, v_0 - v_1 - v)\text{Var}\left[\mathcal{F}\left\{G^l\right\}(u_0 - u, v_0 - v)\right] \tag{16}$$

$$= \sigma^2 \text{Sinc} * \text{Sinc}(u_0 - u_1, v_0 - v_1) = \sigma^2 d_l^2 \text{Sinc}(u_0 - u_1, v_0 - v_1) \tag{17}$$

$$\tag{18}$$

$$\text{Var}\left[U\right] = \text{Var}\left[\text{Sinc} * \mathcal{F}\left\{G^l\right\}(u_0, v_0)\right] \tag{19}$$

$$= \text{Var}\left[\sum_{u,v} \text{Sinc}(u, v)\mathcal{F}\left\{G^l\right\}(u_0 - u, v_0 - v)\right] \tag{20}$$

$$= \sum_{u,v} |\text{Sinc}(u, v)|^2 \text{Var}\left[\mathcal{F}\left\{G^l\right\}(u_0 - u, v_0 - v)\right] = \sigma^2 \sum_{u,v} |\text{Sinc}(u, v)|^2 = \sigma^2 d_l^2 k_l^2 \tag{21}$$

$$\tag{22}$$

$$\text{corr}(U, V) = \frac{\text{Cov}\left[U, V\right]}{\sqrt{\text{Var}\left[U\right]\text{Var}\left[V\right]}} = \frac{\text{Sinc}(u_0 - u_1, v_0 - v_1)}{k_l^2} \tag{23}$$

□

Note that all the expectations are taken over the probability space of all unrestricted 2D filters of a specific output dimension d_l, that is the probability space of G_l when considered as a random variable, and that Sinc^* represents the complex conjugate of Sinc.

APPENDIX B GAN GENERATED SAMPLES

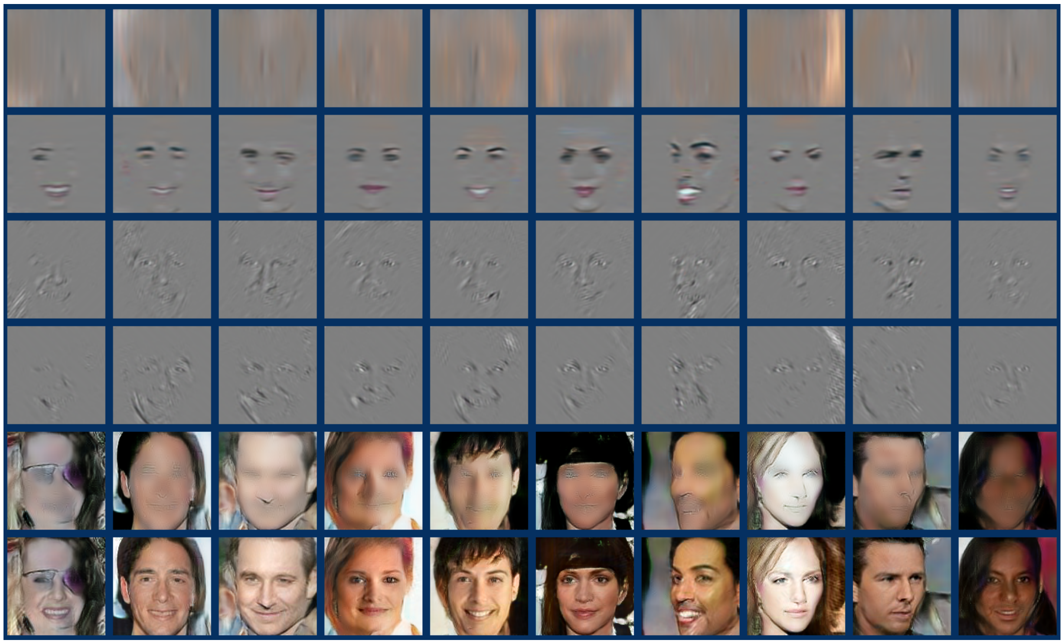


Figure 5: Each column corresponds to one sample of a WGAN-GP enhanced by adding multiple FSGs, trained on CelebA. The first four rows, from top to bottom, show the outputs of the FSGs with $(\hat{u}_t, \hat{v}_t)$ at $(\frac{1}{16}, 0)$, $(0, \frac{1}{16})$, $(-\frac{1}{16}, \frac{1}{16})$, and $(\frac{1}{16}, \frac{1}{16})$ respectively. The fifth row shows the main generator, and the final row the output of the compound generator (sum of all the preceding rows).

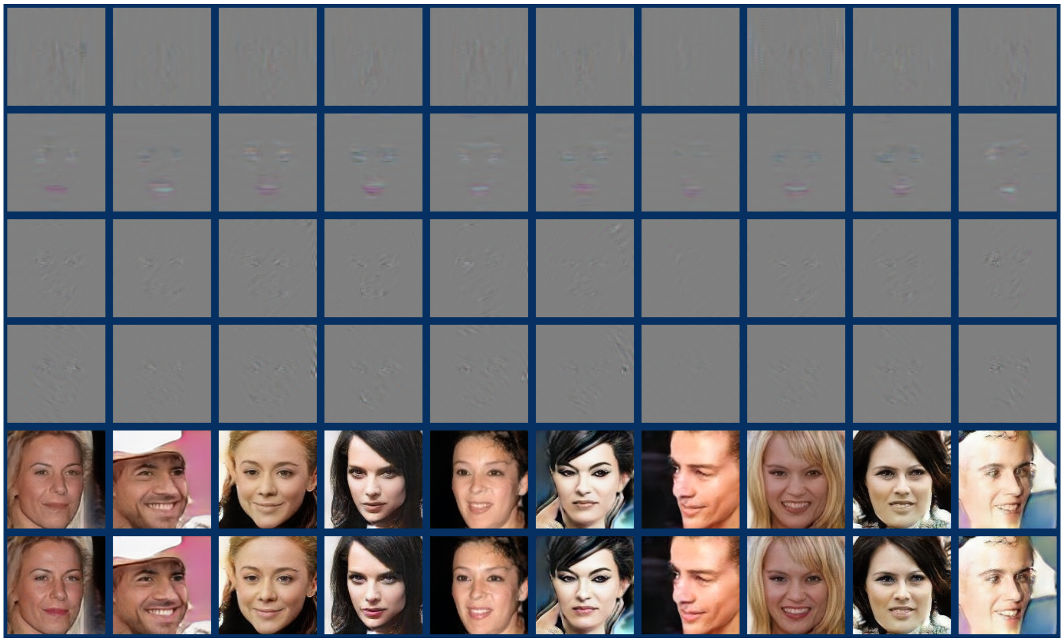


Figure 6: Each column corresponds to one sample of a PG-GAN enhanced by adding multiple FSGs, trained on CelebA. The first four rows, from top to bottom, show the outputs of the FSGs with $(\hat{u}_t, \hat{v}_t)$ at $(\frac{1}{16}, 0)$, $(0, \frac{1}{16})$, $(-\frac{1}{16}, \frac{1}{16})$, and $(\frac{1}{16}, \frac{1}{16})$ respectively. The fifth row shows the main generator, and the final row the output of the compound generator (sum of all the preceding rows).

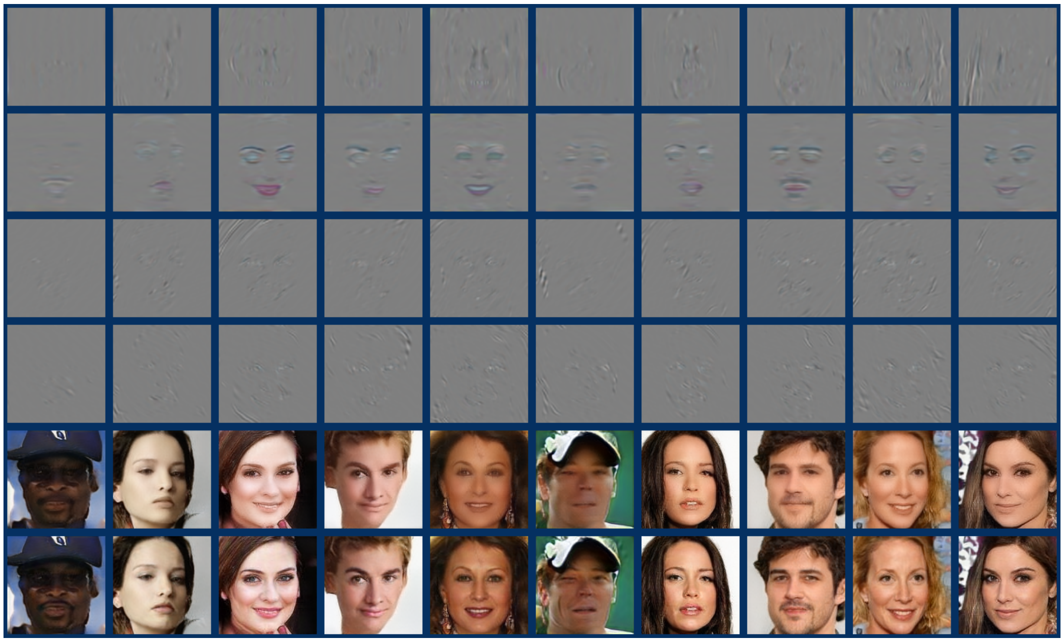
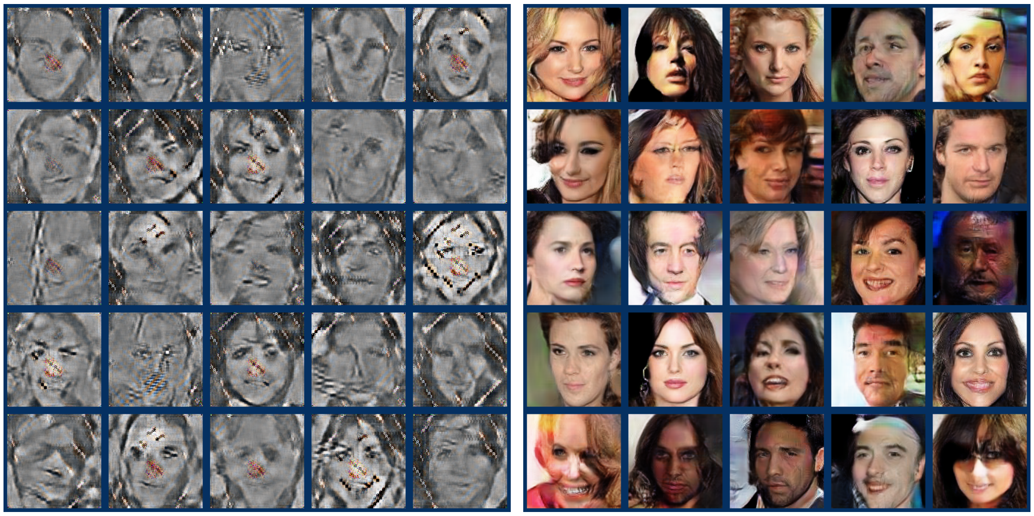
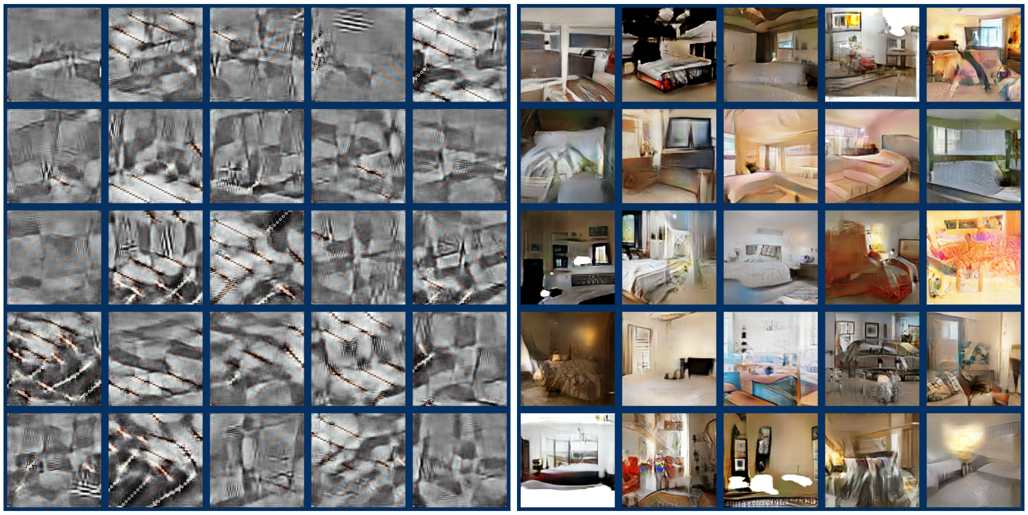


Figure 7: Each column corresponds to one sample of a StyleGAN2 enhanced by adding multiple FSGs, trained on CelebA. The first four rows, from top to bottom, show the outputs of the FSGs with $(\hat{u}_t, \hat{v}_t)$ at $(\frac{1}{16}, 0)$, $(0, \frac{1}{16})$, $(-\frac{1}{16}, \frac{1}{16})$, and $(\frac{1}{16}, \frac{1}{16})$ respectively. The fifth row shows the main generator, and the final row the output of the compound generator (sum of all the preceding rows).



(a) WGAN-GP on SCelebA

(b) WGAN-GP-FSG on SCelebA



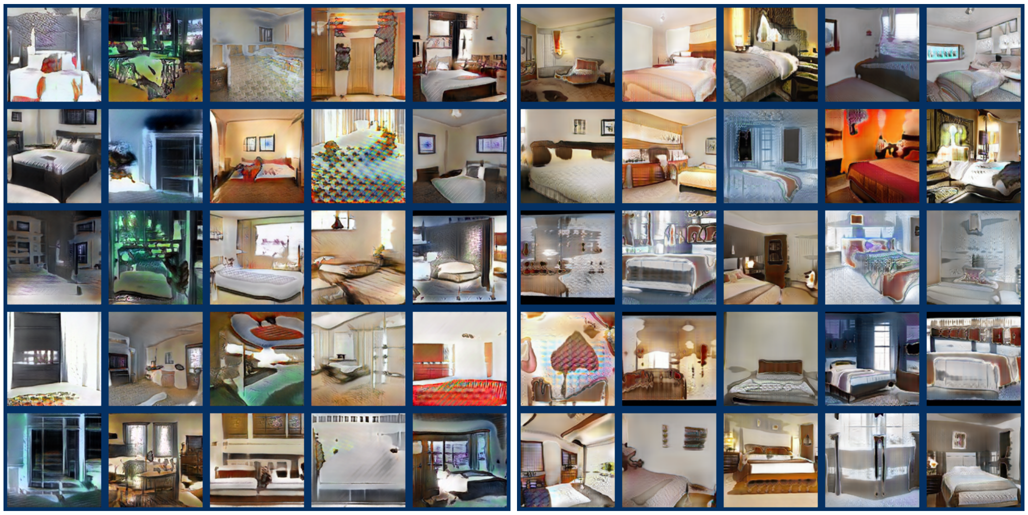
(c) WGAN-GP on SBedrooms

(d) WGAN-GP-FSG on SBedrooms

Figure 8: WGAN-GP random samples compared to WGAN-GP-FSG on SCelebA and SBedrooms. Samples are re-shifted for visualization.

(a) PG-GAN on SCelebA

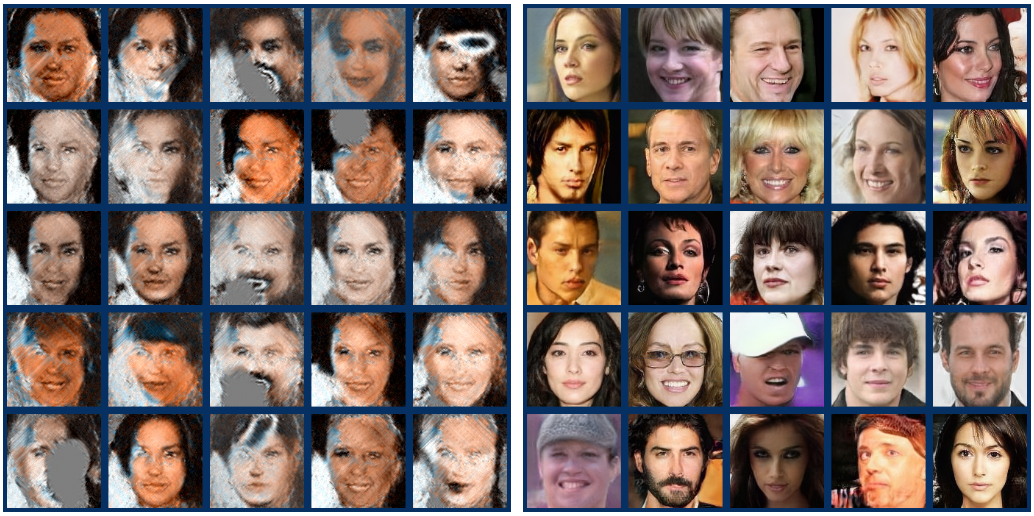
(b) PG-GAN-FSG on SCelebA



(c) PG-GAN on SBedrooms

(d) PG-GAN-FSG on SBedrooms

Figure 9: PG-GAN random samples compared to PG-GAN-FSG on SCelebA and SBedrooms. Samples are re-shifted for visualization.



(a) StyleGAN2 on SCelebA (b) StyleGAN2-FSG on SCelebA

(c) StyleGAN2 on SBedrooms (d) StyleGAN2-FSG on SBedrooms

Figure 10: StyleGAN2 random samples compared to StyleGAN2-FSG on SCelebA and SBedrooms. Samples are re-shifted for visualization.

APPENDIX C EXPERIMENT DETAILS

For WGAN-GP, we use the network in Table 3, and train according to the specifications of Gulrajani et al. (2017) for 200 epochs. For PG-GAN, we use the network in Table 4 and train according to the specifications of Karras et al. (2018) for 1M images. For StyleGAN2, we use the network and training setup of config f in Karras et al. (2020), except that we use the same progression of CNN filter channel sizes as in Table 4, and train for 1M images. On the 1024×1024 fractal dataset of Figure 1, the exact network and setup of config f is used for the large-scale StyleGAN2.

In multiple FSGs experiments, for WGAN-GP, each FSG has the same network as WGAN-GP itself, except it has only up-sampling instead of the last layer, so that FSGs have less capacity than the main generator. For PG-GAN and StyleGAN2, each FSG shares all the Conv blocks of the corresponding main generator network except the two outer most blocks, which are replaced by a single Conv block followed by up-sampling to the output spatial dimension, that is, the FSGs are essentially branching out of the main generator at the start of the 64×64 Conv layer. In StyleGAN2, the added FSGs share the mixing input to the last block of the main generator.

For FID calculation, 50K images are sampled per distribution in CelebA and SCelebA experiments, and 10K images are sampled in LSUN-Bedroom and SBedrooms experiments, due to the limited number of available true images. For spectrum visualization, 1K images are sampled.

Table 3: WGAN-GP's networks. $G(z)$ and $D(x)$ refer to the generator and discriminator respectively, where $z \in [-1, 1]^{128}$ is the randomly sampled latent vector and $x \in [-1, 1]^{128 \times 128 \times 3}$ is the input image.

Operation	Kernel	Strides	Output Shape	Activation
G(z): $z \sim \text{Uniform}(-1, 1)$			128	
Fully Connected			$8 \times 8 \times 512$	ReLU
Nearest Upsample			$16 \times 16 \times 512$	
Convolution	5×5	1×1	$16 \times 16 \times 256$	ReLU
Nearest Upsample			$32 \times 32 \times 256$	-
Convolution	5×5	1×1	$32 \times 32 \times 128$	ReLU
Nearest Upsample			$64 \times 64 \times 128$	-
Convolution	5×5	1×1	$64 \times 64 \times 64$	ReLU
Nearest Upsample			$128 \times 128 \times 64$	-
Convolution	5×5	1×1	$128 \times 128 \times 3$	Tanh
D(x): x			$128 \times 128 \times 3$	
Convolution	5×5	2×2	$64 \times 64 \times 64$	Leaky ReLU
Convolution	5×5	2×2	$32 \times 32 \times 128$	Leaky ReLU
Convolution	5×5	2×2	$16 \times 16 \times 256$	Leaky ReLU
Convolution	5×5	2×2	$8 \times 8 \times 512$	Leaky ReLU
Fully Connected			1	

Table 4: PG-GAN's networks. $G(z)$ and $D(x)$ refer to the generator and discriminator respectively, where $z \in \mathbb{R}^{512}$ is the randomly sampled latent vector and $x \in [-1, 1]^{128\times128\times3}$ is the input image.

Operation	Kernel	Strides	Output Shape	Activation
G(z): $z \sim \mathcal{N}(0,1)$			512	
Convolution	4×4	1×1	$4\times4\times512$	Leaky ReLU
Convolution	3×3	1×1	$4\times4\times512$	Leaky ReLU
Nearest Upsample			$8\times8\times512$	
Convolution	3×3	1×1	$8\times8\times512$	Leaky ReLU
Convolution	3×3	1×1	$8\times8\times512$	Leaky ReLU
Nearest Upsample			$16\times16\times512$	
Convolution	3×3	1×1	$16\times16\times256$	Leaky ReLU
Convolution	3×3	1×1	$16\times16\times256$	Leaky ReLU
Nearest Upsample			$32\times32\times256$	
Convolution	3×3	1×1	$32\times32\times128$	Leaky ReLU
Convolution	3×3	1×1	$32\times32\times128$	Leaky ReLU
Nearest Upsample			$64\times64\times128$	
Convolution	3×3	1×1	$64\times64\times64$	Leaky ReLU
Convolution	3×3	1×1	$64\times64\times64$	Leaky ReLU
Nearest Upsample			$128\times128\times64$	
Convolution	3×3	1×1	$128\times128\times32$	Leaky ReLU
Convolution	3×3	1×1	$128\times128\times32$	Leaky ReLU
Convolution	1×1	1×1	$128\times128\times3$	
D(x): x			$128\times128\times3$	
Convolution	1×1	1×1	$128\times128\times32$	Leaky ReLU
Convolution	3×3	1×1	$128\times128\times32$	Leaky ReLU
Convolution	3×3	1×1	$128\times128\times64$	Leaky ReLU
Box Downsample			$64\times64\times64$	
Convolution	3×3	1×1	$64\times64\times64$	Leaky ReLU
Convolution	3×3	1×1	$64\times64\times128$	Leaky ReLU
Box Downsample			$32\times32\times128$	
Convolution	3×3	1×1	$32\times32\times128$	Leaky ReLU
Convolution	3×3	1×1	$32\times32\times256$	Leaky ReLU
Box Downsample			$16\times16\times256$	
Convolution	3×3	1×1	$16\times16\times256$	Leaky ReLU
Convolution	3×3	1×1	$16\times16\times512$	Leaky ReLU
Box Downsample			$8\times8\times512$	
Convolution	3×3	1×1	$8\times8\times512$	Leaky ReLU
Convolution	3×3	1×1	$8\times8\times512$	Leaky ReLU
Box Downsample			$4\times4\times512$	
Minibatch stddev			$4\times4\times513$	
Convolution	3×3	1×1	$4\times4\times512$	Leaky ReLU
Convolution	4×4	4×4	$1\times1\times512$	Leaky ReLU
Fully Connected			1	