

# Are High-Frequency Components Beneficial for Training of Generative Adversarial Networks

Ziqiang Li

Pengfei Xia

Xue Rui

Yanghui Hu

Bin Li

University of Science and Technology of China

{iceli, xpengfei, ruixue27, huyang1}@mail.ustc.edu.cn, binli@ustc.edu.cn

## Abstract

*Advancements in Generative Adversarial Networks (GANs) have the ability to generate realistic images that are visually indistinguishable from real images. However, recent studies of the image spectrum have demonstrated that generated and real images share significant differences at high frequency. Furthermore, the high-frequency components invisible to human eyes affect the decision of CNNs and are related to the robustness of it. Similarly, whether the discriminator will be sensitive to the high-frequency differences, thus reducing the fitting ability of the generator to the low-frequency components is an open problem. In this paper, we demonstrate that the discriminator in GANs is sensitive to such high-frequency differences that can not be distinguished by humans and the high-frequency components of images are not conducive to the training of GANs. Based on these, we propose two pre-processing methods eliminating high-frequency differences in GANs training: High-Frequency Confusion (HFC) and High-Frequency Filter (HFF). The proposed methods are general and can be easily applied to most existing GANs frameworks with a fraction of the cost. The advanced performance of the proposed method is verified on multiple loss functions, network architectures, and datasets.*

## 1. Introduction

Generative Adversarial Networks (GANs) [11] have been widely used in the field of computer vision: attribute editing [33, 30], generation of photo-realistic images [17, 18], image inpainting [13, 31]. Although the remarkable advancements in community, recent works [19, 9, 8, 6, 40, 10]

have demonstrated that generated images are significantly different from real images in the frequency domain, especially in the high frequency. Due to the bias, some works [10, 40, 9] only use very lightweight networks to solve the deep fake problem in the frequency domain. There are two main hypotheses for the artifacts with generated images in the frequency domain: some studies [10, 8, 40] suggest that this difference is mainly due to the employed of upsampling operations; and some studies [19, 9] suggest that the linear dependencies in the Conv filter's spectrum cause spectral limitations. We urge readers to check out these works for more details. Simultaneously, the difference is also shown in the top part of the Figure 1, which illustrates that the difference in power spectrum between real images and fake images appears around the frequency radius ( $R$ ) of 20.

GANs training is a dynamic game process. A good discriminator is prerequisite for training generators, but the proposal of vanilla GAN [11] is based on the non-parametric assumption of infinite capacity of the discriminator. Without any prior, it is easy to use all information to look for shortcuts during the discriminator training. Inspired by CNN can exploit the high-frequency components that are not perceivable to humans [34], this paper analyzes and discusses the impact of high-frequency components on the training of GANs. The experiments indicate that the discriminator is sensitive to the high-frequency components and these high-frequency components are not conducive to the training of GANs.

Based on these facts, frequency bias between real images and generated images, and high-frequency response of discriminator, two aspects can be implemented to improve the performance of GANs. One is to improve the discriminatory ability of the discriminator in the frequency

Bin Li is the corresponding author.  
Under Review

Fake images are generated by SNGAN [26] on CIFAR-10 dataset, the resolution is 32\*32. Humans can not distinguish with  $R = 20$  (see Section 3.2)

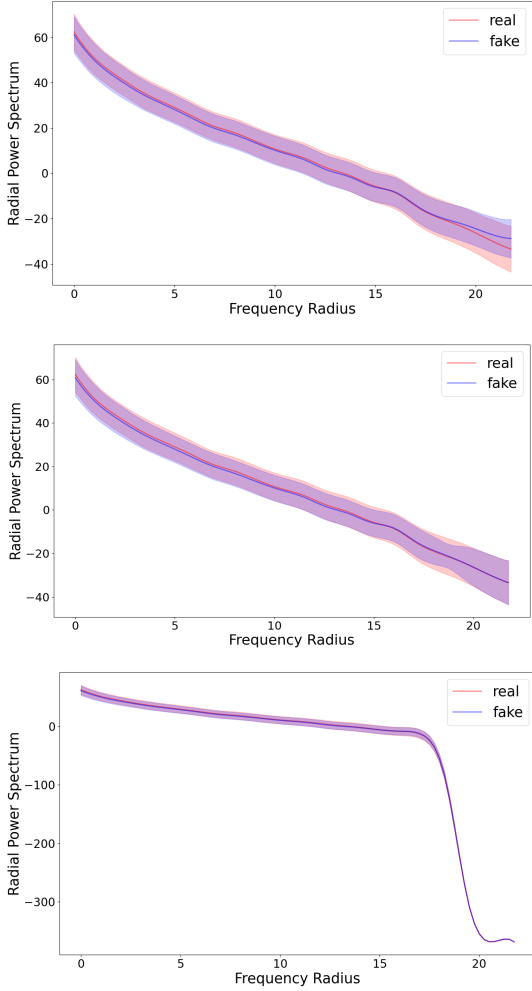


Figure 1: The Radial Power Spectrum (see Section 3.1) of the real images and fake images generated by SNGAN [26] on CIFAR-10 dataset. **Top:** mean and variance after azimuthal integration over the power spectrum of real and generated images. **Middle and Bottom:** results of the same experiments as above after using High Frequency Confusion (HFC) and High Frequency Filter (HFF) preprocessing method, respectively. The calculation of the power spectrum involves logarithmic functions, the high frequency components will mutate while using HFF.

domain, which is introduced by Chen et al. [6]. Specifically, they boost a frequency-aware classifier into the discriminator to measure the realness of the image in both the spatial and spectral domains. But gradient vanishing may occur along with the improvement of discriminative power. And the other one is to remove the high-frequency differences during the training of GANs, which makes the discriminator unrelated to the high-frequency components and

focuses more on low-frequency components. This aspect is robust to the parameters and discussed in this paper. We propose two data preprocessing methods (HFC and HFF) so that there is no difference between the generated images and the real images in high frequency. Among them, HFC replaces the high-frequency components of the generated images with the high-frequency components of the real images, while HFF uses a high-frequency filter to filter out the high-frequency components of both real and generated images. The radial power spectrum after preprocessing with HFC and HFF on the CIFAR-10 dataset can be observed from the middle part and bottom part of Figure 1, respectively.

The main contributions of this paper can be summarized as follows:

- This paper aims for the first pilot study to analyze the effect of different high-frequency components on the discriminator and demonstrate that high-frequency components are not conducive to the training of GANs.
- To balance the high-frequency difference between the generated images and the real images during the training of GANs, two data preprocessing methods are proposed in this paper. The proposed methods improve the quality of generation on many loss functions, architectures, and datasets with little cost.
- Through lots of experiments, we obtain the experimental conclusions about the selection of filter Radius (R).

## 2. Related Work

### 2.1. Generative Adversarial Networks

Generative Adversarial Networks (GANs) are two-player zero-sum games, where the generator  $G(z)$  is a distribution mapping function that transforms low-dimensional latent distribution  $z$  into target distribution  $p_g$ . And the discriminator  $D(x)$  evaluates the distance between generated distribution  $p_g$  and real distribution  $p_r$ . The generator and discriminator minimize and maximize the distribution distance, respectively. This minimax game can be expressed as:

$$\min_{\phi} \max_{\theta} f(\phi, \theta) = \mathbb{E}_{x \sim p_r} [g_1(D_{\theta}(x))] + \mathbb{E}_{z \sim p_z} [g_2(D_{\theta}(G_{\phi}(z)))], \quad (1)$$

where  $\phi$  and  $\theta$  are parameters of the generator  $G$  and discriminator  $D$ , respectively. Specifically, vanilla GAN [11] can be described as  $g_1(t) = g_2(-t) = -\log(1 + e^{-t})$ ;  $f$ -GAN [27] can be written as  $g_1(t) = -e^{-t}$ ,  $g_2(t) = 1 - t$ ; Moreover, Geometric GAN [24] and WGAN [1] are described as  $g_1(t) = g_2(-t) = -\max(0, 1 - t)$  and  $g_1(t) = g_2(-t) = t$ , respectively. Recently, the quality

of the generated images has been improved dramatically. Karras et al. [16] describe a new training methodology in GANs, which grows both the generator and discriminator progressively. The progressive method makes the quality of high-resolution image synthesis improved unprecedentedly. Also, Karras et al. propose StyleGAN [17] and StyleGAN++ [18], which use style-based method to the generation of specific content. Furthermore, some regularization and normalization methods are used to stabilize GANs training [23], WGAN-GP [14] and SNGAN [26] achieve 1-Lipschitz continuity using gradient penalty and spectral normalization, respectively; GAN-DAT [22] minimizes Lipschitz constant to avoid the discriminator being attacked by the adversarial examples; SS-GAN [5] uses self-supervised techniques to avoid discriminator forgetting; Similarly, InfoMAXGAN [21] improves the image generation via information maximization and contrastive learning, which is an unsupervised method for mitigating catastrophic forgetting.

In this paper, two preprocess methods are used in three loss functions: vanilla GAN loss, LSGAN loss, Geometric loss (Hinge loss) and three frameworks: SNGAN, SSGAN, and InfoMAXGAN. Experiments show the consistent advancement of our methods.

## 2.2. Frequency Principle for CNNs

Recently, frequency analysis on Convolution Neural Networks (CNNs) has attracted more and more attention. According to the different analysis objects, Frequency of CNNs can be divided into two categories: Input (Image) Frequency and Response Frequency. Many works have shown that there is a principle in both categories of frequency: **DNNs often fit target functions from low to high frequencies during the training**, which is called **Frequency Principle (F-Principle)** in [38, 37] or **Spectral Bias** in [28]. Input (Image) Frequency corresponds to the rate of change of intensity across neighboring pixels of input images, where sharp textures contribute to the high frequency of the image. Wang et al. [34] demonstrate the F-Principle of Input Frequency and believe that the high-frequency components of the image improve the performance of the model, but reduce its generalization.

Furthermore, Response Frequency is the frequency of general Input-Output mapping  $f$ , which measures the rate of change of outputs with respect to inputs. If  $f$  possesses significant high frequencies, then a small change of inputs in the image might induce a large change of the output (e.g. adversarial example in [12]). F-Principle of Response Frequency is first proposed by Xu et al. [38] through regression problems on synthetic data and real data with principal component analysis. Similarly, Rahaman et al. [28] find empirical evidence of a spectral bias: lower frequencies are learned first. They also show that lower frequencies are more robust to random perturbations of the network pa-

rameters. Subsequently, Xu et al. [35] propose a theoretical analysis framework for one hidden layer neural network with 1-d input, which illustrates activation functions (including Tanh and Relu) in the Fourier domain decays as frequency increases. Furthermore, Xu et al. [37] also propose a Gaussian filtering method that can directly verify the F-Principle in high-dimensional datasets for both regression and classification problems. Certainly, there are still a lot of works [36, 41, 2, 4, 32] to be done in the frequency domain to interpret the success and failure of CNNs. We urge readers to check out these works for more details.

## 2.3. Frequency Bias for GANs

Frequency analysis of GANs is also popular, from which we are mainly concerned with the Input (Image) Frequency. The frequency bias of GANs mainly refers to the bias between the generated images and the real images in the frequency domain, which can be observed in the top part of Figure 1. The same phenomenon has also been found in other works [10, 40, 9, 6, 8, 19]. They all demonstrate that this phenomenon is pervasive and hard to avoid, even if some generated images are flawless from human perception. In addition to the two main hypotheses (the employment of upsampling operations and linear dependencies in the Conv filter’s spectrum) introduced in Introduction, Chen et al. [6] reveal that downsampling layers cause the high frequencies missing in the discriminator. This issue makes the generator lacking the gradient information to model high-frequency content, resulting in a significant spectrum discrepancy between generated images and real images. Specifically, they boost a frequency-aware classifier into the discriminator to measure the realness of the image in both the spatial and spectral domains.

The frequency bias between the generated image and the real image is generally accepted. Based on this bias, many works [10, 40, 9] only use very lightweight networks to solve the deep fake problem in the frequency domain.

## 3. Frequency Decomposition and Analysis

This section demonstrates the Discrete Fourier Transform (DFT) of 2D images and the response of discriminator for different frequency components.

### 3.1. Discrete Fourier Transform of Image Data

The DFT  $\mathcal{F}$  of 2D image data  $I$  with size  $N \times N$  can be shown as:

$$\mathcal{F}(I)(k, l) = \sum_{m=0}^{N-1} \sum_{n=0}^{N-1} I(m, n) \cdot e^{-2\pi j \left( \frac{k}{N}m + \frac{l}{N}n \right)},$$

for  $k = 0, \dots, N-1, \quad l = 0, \dots, N-1.$

(2)

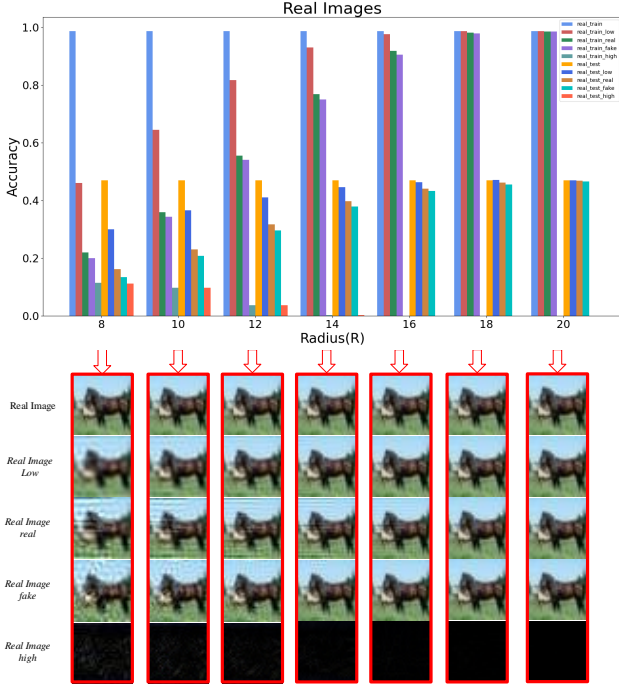


Figure 2: The response of the discriminator to the real images on CIFAR-10 dataset. **Top:** the accuracy of the trained discriminator for the real images (The model is SNGAN trained by Hinge Loss. The output of the discriminator goes through the Sigmoid function. Output of the Sigmoid greater than 0.5 means the image is real and less than 0.5 means the image is fake.). **Bottom:** visualization of different frequency components in the image domain (The image is sampled from the training set.)

And the Power Spectrum of DFT is  $PS(I)(k, l) = 20 \cdot \log |\mathcal{F}(I)(k, l)|$ . At the last, the Radial Power Spectrum (RPS) of DFT via azimuthal integration can be shown as:

$$RPS(\omega_k) = \int_0^{2\pi} PS(I)(\omega_k \cdot \cos(\phi), \omega_k \cdot \sin(\phi)) d\phi, \\ \text{for } k = 0, \dots, N/\sqrt{2}. \quad (3)$$

Figure 1 illustrates the RPS of real and fake images on CIFAR-10 dataset.

Further, frequency decomposition and frequency mixing of the real image  $x_r$  and the generated image  $x_g$  are defined

as follows::

$$\begin{aligned} z_r &= \mathcal{F}(x_r), & z_r^l, z_r^h &= \mathcal{T}(z_r; R), \\ z_g &= \mathcal{F}(x_g), & z_g^l, z_g^h &= \mathcal{T}(z_g; R), \\ x_r^l &= \mathcal{F}^{-1}(z_r^l), & x_r^h &= \mathcal{F}^{-1}(z_r^h), \\ x_r^r &= \mathcal{F}^{-1}(z_r^l, z_r^h), & x_r^f &= \mathcal{F}^{-1}(z_r^l, z_r^h), \\ x_g^l &= \mathcal{F}^{-1}(z_g^l), & x_g^h &= \mathcal{F}^{-1}(z_g^h), \\ x_g^f &= \mathcal{F}^{-1}(z_g^l, z_g^h), & x_g^r &= \mathcal{F}^{-1}(z_g^l, z_g^h), \end{aligned} \quad (4)$$

where  $\hat{z}_r^h$  represents the high-frequency component of another real image that is different from  $x_r$ .  $\hat{z}_g^h$  represents the high-frequency component of another generated image that is different from  $x_g$ . In addition,  $\mathcal{T}(\cdot; R)$  denotes a thresholding function that separates the low and high frequency components from  $z$  according to the radius  $R$ . The formal definition of the equation  $z_l, z_h = \mathcal{T}(z; R)$  is as follows:

$$\begin{aligned} z_l(i, j) &= \begin{cases} z(i, j), & \text{if } d((i, j), (c_i, c_j)) \leq R \\ 0, & \text{otherwise} \end{cases}, \\ z_h(i, j) &= \begin{cases} 0, & \text{if } d((i, j), (c_i, c_j)) \leq R \\ z(i, j), & \text{otherwise} \end{cases}, \end{aligned} \quad (5)$$

where  $z(i, j)$  represents the value of  $z$  at position index  $(i, j)$  and  $c_i, c_j$  indicate the index of the center point of the image. We use  $d(\cdot)$  as the Euclidean distance in this paper.

The equations in Eq (4) define the different frequency components of the image, some are frequency decompositions ( $x_r^l, x_r^h, x_g^l$ , and  $x_g^h$ ) and some are frequency mixtures ( $x_r^r, x_r^f, x_g^f$ , and  $x_g^r$ ). Properties of these frequency components will be discussed in the following sections.

### 3.2. Responses of the Discriminator to different frequency Components

This subsection will analyze the responses of the discriminator to different frequency components introduced in Eq (4). We calculate the accuracy of the trained SNGAN on the CIFAR-10 dataset for both real and generated images. The SNGAN uses 50000 images for training and 10000 images for testing. For generality, the accuracy on the training set and the testing set are shown at the top part of Figure 2, where real\_train, real\_train\_low, real\_train\_real, real\_train\_fake, and real\_train\_high represent  $x_r, x_r^l, x_r^r, x_r^f$ , and  $x_r^h$  in the training set, respectively. Similarly, real\_test, real\_test\_low, real\_test\_real, real\_test\_fake, and real\_test\_high represent  $x_r, x_r^l, x_r^r, x_r^f$ , and  $x_r^h$  in the testing set. Furthermore, the visualization of different frequency components in the image domain (The image is sampled from the training set.) are also shown at the bottom part

Since images have 3 channels, the below operations are performed separately on each channels

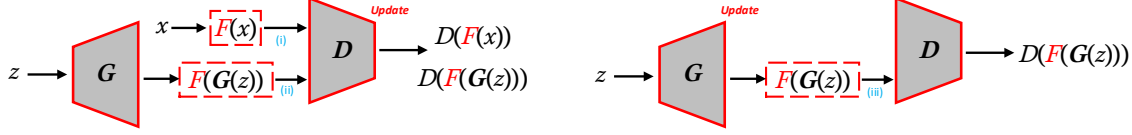


Figure 3: Overview of the image preprocessing (HFF or HFC) for updating discriminator (left) and generator(right).

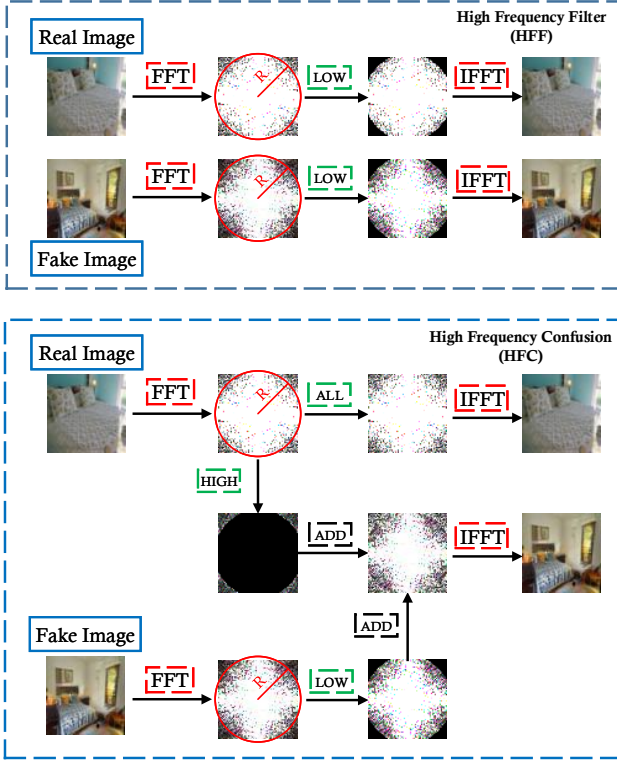


Figure 4: The framework of the proposed methods High Frequency Filter (HFF) and High Frequency Confusion (HFC), in which Fast Fourier Transform (FFT) is a fast method to compute the Discrete Fourier Transform (DFT) of a sequence and IFFT is Inverse FFT. The operations LOW, HIGH and ALL are used to select the low-frequency component ( $x^l$ ), high-frequency component ( $x^h$ ) and all-frequency components ( $x^l, x^h$ ) of the image ( $x$ ).

of Figure 2. The visualization illustrates that the high-frequency component is not visible to the human eyes when the frequency threshold radius ( $R$ ) is greater than or equal to 16. Specifically,  $x_r$ ,  $x_r^l$ ,  $x_r^r$ , and  $x_r^f$  can not be distinguished by humans when  $R \geq 16$ . However, the trained discriminator has differences in response to different frequency components, even when  $R \geq 16$ . It is worth noting that under different threshold radius ( $R$ ), the accuracy of  $x_r^f$  is always lower than that of  $x_r^r$ , which indicates that the discrimina-

tor of GANs uses the high-frequency information of images during the training, and it also demonstrates that there is a difference between real images and generated images at high frequency.

Similarly, responses of the discriminator to the generated images are demonstrated in the Subsection 1.1 of **Supplemental material**. It is clear that the accuracy of  $x_g^f$  is greater than that of  $x_g^r$ , no matter what  $R$  is.

#### 4. High Frequency Filter (HFF) And High Frequency Confusion (HFC)

In this section, we propose two image preprocessing methods to eliminate the bias between the generated images and the real images during the training of GANs. Figure 3 illustrates the overview of the HFF and HFC for updating discriminator and generator during GANs training, the loss functions are:

$$\min_{\phi} \max_{\theta} f(\phi, \theta) = \mathbb{E}_{x \sim p_r} [g_1(D_{\theta}(F(x)))] + \mathbb{E}_{z \sim p_z} [g_2(D_{\theta}(F(G_{\phi}(z))))], \quad (6)$$

where  $F$  is required to be the same function (HFF or HFC) and the same threshold radius ( $R$ ) across the three places illustrated in Figure 3. The framework of HFF and HFC can be seen in Figure 4, where HFF uses a filter to filter out the high-frequency components of the images, while HFC mixes two types of images by replacing the high-frequency components of the generated images with the high-frequency components of the real images. The form can be expressed as:

$$\begin{aligned} HFF(x_r, x_g; R) &= x_r^l, x_g^l \\ HFC(x_r, x_g; R) &= x_r, (x_g^l, x_r^h) \end{aligned} \quad (7)$$

#### 5. Experiment

In the experiment, we will analyse the hypothesis: **high-frequency components are not conducive to the training of GANs** and demonstrate the advanced performance of our methods for various GANs frameworks of different loss functions, network architectures, and datasets. The details of datasets and experiment settings are shown in Subsection 1.2 of **Supplemental material**.



Table 1: Training FID with using HFF in different places for training the GANs on CIFAR-10 dataset. "Real only" applies HFF to real (i) (see Figure 3); "Fake only" applies HFF to fake (ii); "Dis only" applies HFF to both real (i) and fake (ii), but not G (iii); "All" applies HFF to real (i), fake (ii), and G (iii). All FID scores are calculated on 50K training samples, and we select the best FID score for each method during the training.

| Method                | Where $F$ ? |      |       | Radius (R) |       |       |       |       |       |
|-----------------------|-------------|------|-------|------------|-------|-------|-------|-------|-------|
|                       | (i)         | (ii) | (iii) | 12         | 14    | 16    | 18    | 20    | 22    |
| SNGAN(Baseline)       |             |      |       | 20.05      | 20.05 | 20.05 | 20.05 | 20.05 | 20.05 |
| SNGAN-HFF (Real only) | ✓           |      |       | 26.65      | 23.59 | 20.02 | 20.3  | 19.68 | 20.16 |
| SNGAN-HFF (Fake only) |             | ✓    |       | 350        | 350   | 100   | 68    | 20.07 | 20.67 |
| SNGAN-HFF (Dis only)  | ✓           | ✓    |       | 350        | 52.2  | 20.07 | 20.59 | 19.89 | 19.53 |
| SNGAN-HFF (All)       | ✓           | ✓    | ✓     | 21.29      | 21.06 | 18.74 | 18.5  | 18.54 | 19.59 |

### 5.1. High-frequency Components are not Conducive to the Training of GANs

This subsection analyses the hypothesis: **high-frequency components are not conducive to the training of GANs**. Train FID and Test FID with using HFF in different places ((i), (ii), (iii) in Figure 3) for training the GANs on the CIFAR-10 dataset are shown in Table 1 and Figure 4, respectively. The results illustrate: (1) Excessive filtering of high-frequency components makes the training of "Real only", "Fake only", and "Dis only" methods fail, and reduces the performance of the "All" method; (2) When the threshold radius (R) is relatively large, performance of "Real only", "Fake only", and "Dis only" methods are similar to that of normal GANs (baseline), while performance of the "All" method is better than that of baseline; (3) HFF operation on generated images has a greater impact than the same operation on real images. These results indicate that adding artificial high-frequency bias between the real images and the generated images ("Real only", "Fake only" and "Dis only") has little effect on the training of GANs while removing the high-frequency bias between the generated images and the real images ("All") improves the performance of GANs, which serve as strong evidences for the hypothesis: **high-frequency components are not conducive to the training of GANs**.

### 5.2. The Results on CIFAR-10 Dataset with Different Loss Functions

This subsection validates whether our proposed preprocessing methods are robust to different loss functions. Three different loss functions (gan [11], lsgan [25], hinge gan [24]) are used to train SNGAN on the CIFAR-10 dataset. The Training FID and Inception Score (IS) are shown in Figure 6 and Table 2, respectively. Compared to HFF, it is easy to understand that HFC fails to train at small radii. HFC replaces the high-frequency components of the generated images with the high-frequency components of the

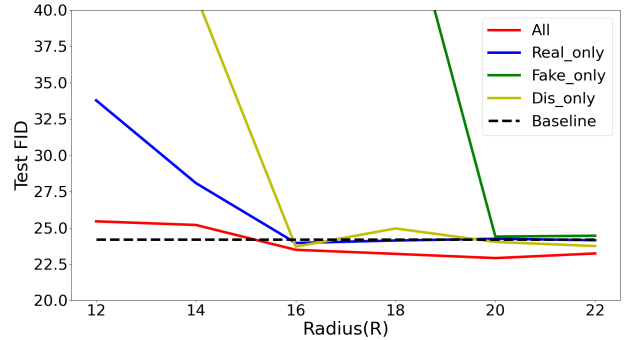


Figure 5: Testing FID with using HFF in different places for training the SNGAN on the CIFAR-10 dataset. All FID scores are calculated on 10K testing samples, and we select the best FID score for each method. The other settings are the same as in Table 1.

real images, which leads to a significant discontinuity in the spectrum domain of the generated images as shown in Figure 4. The spectral discontinuity of the generated images leads to deviations between them and the real images, which leads to the fail of training. When R is large, this spectral discontinuity is minor, and the deviations between real images and generated images are imperceptible. Hence HFC is only suitable for larger radii.

The proposed methods HFC and HFF both improve the performance of the baseline. Furthermore, the best performances of HFF and HFC are achieved when R equals to 20 (The vast majority of cases), which corresponds to the phenomenon illustrated in the Top part of the Figure 1 (High-frequency bias occurs at the radius of 20 in the CIFAR-10 dataset).

Table 2: Inception Score (IS) on CIFAR-10 dataset with different loss functions in SNGAN. All Inception scores are calculated on 50K training samples, and we select the best IS for each method.

| Loss      | Method | Radius (R) |      |      |      |             |             |                     |
|-----------|--------|------------|------|------|------|-------------|-------------|---------------------|
|           |        | 12         | 14   | 16   | 18   | 20          | 22          | $\infty$ (Baseline) |
| gan       | HFF    | 7.445      | 7.56 | 7.58 | 7.66 | <b>7.79</b> | 7.78        | 7.68                |
|           | HFC    | 3.2        | 3.5  | 4.5  | 6.36 | 7.42        | <b>7.76</b> |                     |
| lsgan     | HFF    | 7.63       | 7.76 | 7.76 | 7.66 | <b>7.89</b> | 7.89        | 7.74                |
|           | HFC    | 3.2        | 5.51 | 6.22 | 7.47 | <b>7.82</b> | 7.74        |                     |
| hinge gan | HFF    | 7.79       | 7.77 | 7.86 | 7.85 | <b>7.95</b> | 7.86        | 7.84                |
|           | HFC    | 3.5        | 4.3  | 5.67 | 7.89 | <b>8.01</b> | 7.9         |                     |

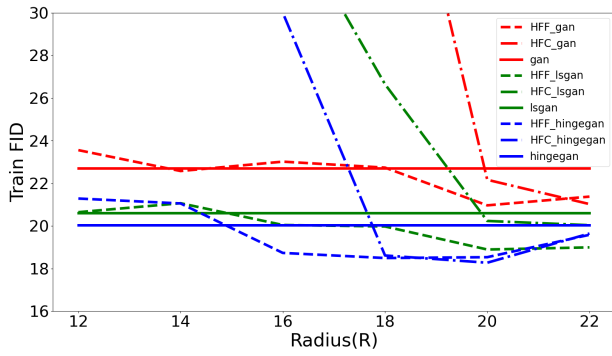


Figure 6: Training FID on CIFAR-10 dataset with different loss functions in SNGAN. All FID scores are calculated on 50K training samples, and we select the best FID score for each method during the training.

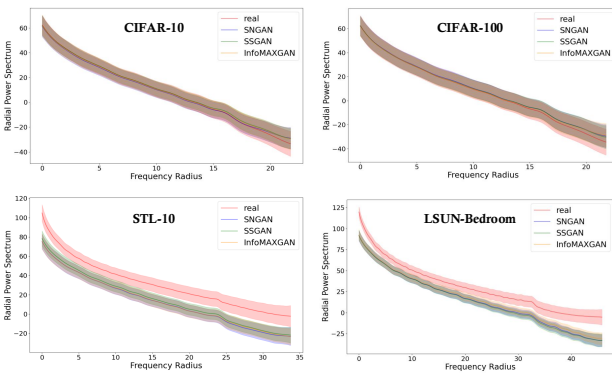


Figure 7: The bias of Radial Power Spectrum between real and generated images on four datasets.

### 5.3. Evaluation on Different Datasets and Different Architectures

In this subsection, HFF and HFC are adopted on some popular models, such as SNGAN, SSGAN, and InfoMAXGAN. Hinge loss is selected for all experiments. The baseline code can be available on Github. We evaluate our methods on four different datasets at multiple resolutions: CIFAR-10 ( $32 \times 32$ ) [20], CIFAR-100 ( $32 \times 32$ ) [20], STL-10 ( $48 \times 48$ ) [7], and LSUN-Bedroom ( $64 \times 64$ ) [39]. Also, three metrics are adopted to evaluate the quality of generated images: Fréchet Inception Distance (FID) [15], Kernel Inception Distance (KID) [3], and Inception Score (IS) [29]. In general, FID and KID measure the diversity between real and generated images. IS only uses generated images to measure the quality. Therefore, to validate the methods, FID and KID are calculated on both training and testing datasets.

Radius (R) is an important parameter to control the range of frequency during GANs training. Here, different R settings are evaluated on four datasets to investigate their effect on the training of GANs. Firstly, the bias of Radial Power Spectrum between real and generated images on four datasets are demonstrated in Figure 7. Figure 7 illustrates that the bias between real and generated images on simple datasets such as CIFAR-10 and CIFAR-100 mainly exist in the high-frequency components. But for complex datasets such as STL-10 and LSUN-Bedroom, all-frequency components exist the bias, which introduces uncertainty into the setting of the optimal Radius (R). To investigate the effect of different settings of Radius (R), Figure 8 illustrates the Training FID of different radii on four datasets. It is clear that: (1) the proposed methods HFF and HFC have a small improvement for InfoMAXGAN, but the improvement is significant for SSGAN and SNGAN. (2) the optimal Ra-

<https://github.com/kwotsin/mimicry>

Due to the lack of computing resources, we are unable to do validation experiments at higher resolutions

Table 3: FID, KID, and IS scores of popular models on different datasets. FID and KID: lower is better. IS: higher is better.

| Metric | Dataset      | Evaluation set | Models |        |        |        |        |        |             |        |        |
|--------|--------------|----------------|--------|--------|--------|--------|--------|--------|-------------|--------|--------|
|        |              |                | SNGAN  |        |        | SSGAN  |        |        | InfoMAX-GAN |        |        |
|        |              |                | None   | HFF    | HFC    | None   | HFF    | HFC    | None        | HFF    | HFC    |
| FID    | LSUN-Bedroom | Train          | 40.47  | 37.71  | 38.77  | 40.01  | 35.79  | 38.36  | 40.32       | 39.37  | 39.79  |
|        |              | Test           | 43.22  | 40.31  | 41.8   | 43.07  | 38.83  | 41.06  | 42.89       | 42.39  | 42.37  |
|        | STL-10       | Train          | 40.77  | 38.99  | 38.43  | 38.78  | 35.13  | 35.14  | 38.55       | 37.44  | 38.64  |
|        |              | Test           | 58.5   | 57.82  | 56.45  | 55.2   | 52.56  | 52.49  | 56.92       | 55.98  | 56.7   |
|        | CIFAR-100    | Train          | 23.70  | 21.45  | 21.98  | 22.43  | 20.27  | 21.51  | 21.36       | 20.41  | 19.16  |
|        |              | Test           | 28.48  | 26.4   | 26.7   | 27.61  | 25.26  | 26.61  | 26.36       | 25.32  | 25.41  |
|        | CIFAR-10     | Train          | 20.05  | 18.54  | 18.28  | 16.75  | 14.71  | 14.88  | 17.71       | 16.19  | 17.17  |
|        |              | Test           | 24.19  | 22.92  | 22.34  | 21.23  | 19.14  | 19.31  | 22.03       | 20.43  | 21.32  |
| KID    | LSUN-Bedroom | Train          | 0.0486 | 0.0422 | 0.0465 | 0.0475 | 0.0436 | 0.0453 | 0.0481      | 0.0466 | 0.0472 |
|        |              | Test           | 0.0489 | 0.0438 | 0.0471 | 0.0482 | 0.0442 | 0.0457 | 0.0484      | 0.0462 | 0.0474 |
|        | STL-10       | Train          | 0.0384 | 0.0368 | 0.0377 | 0.0362 | 0.0328 | 0.0331 | 0.0369      | 0.0354 | 0.0372 |
|        |              | Test           | 0.0409 | 0.0391 | 0.04   | 0.0385 | 0.0352 | 0.0356 | 0.0385      | 0.0374 | 0.04   |
|        | CIFAR-100    | Train          | 0.0158 | 0.0131 | 0.0135 | 0.0155 | 0.014  | 0.0156 | 0.0152      | 0.0139 | 0.0133 |
|        |              | Test           | 0.0164 | 0.0134 | 0.0139 | 0.0160 | 0.0143 | 0.0157 | 0.0153      | 0.0145 | 0.0134 |
|        | CIFAR-10     | Train          | 0.0149 | 0.0138 | 0.0135 | 0.0125 | 0.0113 | 0.0113 | 0.0134      | 0.0112 | 0.0124 |
|        |              | Test           | 0.0150 | 0.0144 | 0.0138 | 0.0128 | 0.012  | 0.0118 | 0.0139      | 0.0119 | 0.0124 |
| IS     | LSUN-Bedroom | None           | -      | -      | -      | -      | -      | -      | -           | -      | -      |
|        | STL-10       | None           | 8.43   | 8.58   | 8.54   | 8.58   | 8.79   | 8.76   | 8.63        | 8.74   | 8.69   |
|        | CIFAR-100    | None           | 7.66   | 7.85   | 7.84   | 7.74   | 7.85   | 7.96   | 8.02        | 8.02   | 8.1    |
|        | CIFAR-10     | None           | 7.84   | 7.95   | 8.01   | 8.13   | 8.1    | 8.09   | 8.01        | 8.1    | 8.0    |

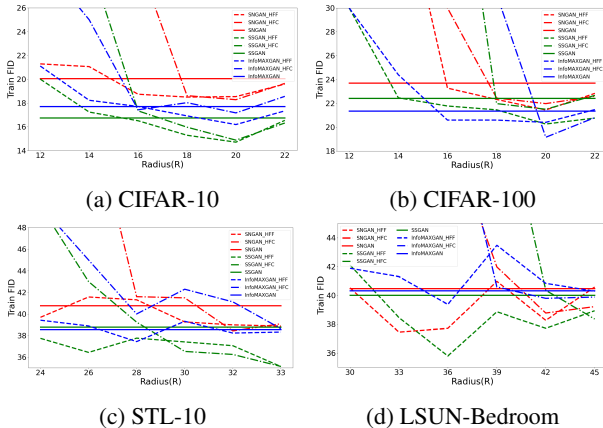


Figure 8: Training FID results for different Radius (R) on different datasets

dius of HFF is set at approximately the same location as the appearance of frequency bias on CIFAR-10 and CIFAR-100 datasets. Due to the underfitting on complex datasets (STL-10 and LSUN-Bedroom), the frequency bias exists in the whole frequency domain. There is uncertainty in the optimal Radius for different models. Furthermore, the Test KID

The reason we mainly consider HFF is that HFC can cause model collapse when the Radius is small

results of different radii on four datasets are shown in Subsection 1.3 of **Supplemental material**.

At the end, we demonstrate the whole results under the optimal Radius (R) in Table 3. Specifically, the optimal Radius for both HFF and HFC is chosen to be 20 on the CIFAR-10, and CIFAR-100 datasets, the optimal Radius for HFF of SNGAN, SSGAN, and InfoMAXGAN on STL-10 dataset are chosen to be 32, 33, and 28 respectively, the optimal Radius for HFC of SNGAN, SSGAN, and InfoMAXGAN on STL-10 dataset are chosen to be 32, 33, and 33 respectively, the optimal Radius for HFF of all models on LSUN-Bedroom dataset are chosen to be 36, and the optimal Radius for HFC of SNGAN, SSGAN, and InfoMAXGAN on LSUN-Bedroom dataset are chosen to be 42, 45, and 42 respectively.

For qualitative comparisons, we present randomly sampled, non-cherry picked images generated by SSGAN and SSGAN with HFF for CIFAR-10, CIFAR-100, STL-10, and LSUN-Bedroom datasets in Subsection 1.4 of **Supplemental material**.

## 6. Conclusion and Outlook

This paper indicates that discriminator is sensitive to the high-frequency components and high-frequency components of images are not conducive to the training of GANs. To avoid the high-frequency differences between the gener-



ated images and the real images during the GANs training, two data preprocessing methods (HFF and HFC) are introduced in this paper. We provide empirical evidences that the proposed HFF and HFC improve the performance of the generation with a little cost.

Frequency analysis is an interesting and new perspective on neural networks. Recent works have shown that the high-frequency component is related to the generalization and robustness of CNNs. This paper aims for the first pilot study of different frequency components on GANs training. Although significant conclusions and remarkable results have been obtained, the modeling process of GANs for different frequency components is still unclear. In the future, we will verify the frequency principles (DNNs often fit target functions from low to high frequencies during the training) of generators and discriminators, and also explore the relationship between different frequency components and robustness and generalization of GANs training. We hope our work set forth towards a new perspective: Frequency analysis in GANs training.

## 7. Acknowledgment

The work is partially supported by the National Natural Science Foundation of China under grand No.U19B2044, No.61836011.

## References

- [1] Martin Arjovsky, Soumith Chintala, and Léon Bottou. Wasserstein gan. *arXiv preprint arXiv:1701.07875*, 2017. [2](#)
- [2] Ronen Basri, David Jacobs, Yoni Kasten, and Shira Kritchman. The convergence rate of neural networks for learned functions of different frequencies. *arXiv preprint arXiv:1906.00425*, 2019. [3](#)
- [3] Mikołaj Bińkowski, Dougal J Sutherland, Michael Arbel, and Arthur Gretton. Demystifying mmd gans. *arXiv preprint arXiv:1801.01401*, 2018. [7](#)
- [4] Yuan Cao, Zhiying Fang, Yue Wu, Ding-Xuan Zhou, and Quanquan Gu. Towards understanding the spectral bias of deep learning. *arXiv preprint arXiv:1912.01198*, 2019. [3](#)
- [5] Ting Chen, Xiaohua Zhai, Marvin Ritter, Mario Lucic, and Neil Houlsby. Self-supervised gans via auxiliary rotation loss. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 12154–12163, 2019. [3](#)
- [6] Yuanqi Chen, Ge Li, Cece Jin, Shan Liu, and Thomas Li. Ssd-gan: Measuring the realness in the spatial and spectral domains. *arXiv preprint arXiv:2012.05535*, 2020. [1, 2, 3](#)
- [7] Adam Coates, Andrew Ng, and Honglak Lee. An analysis of single-layer networks in unsupervised feature learning. In *Proceedings of the International Conference on Artificial Intelligence and Statistics*, pages 215–223, 2011. [7](#)
- [8] 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*, pages 7890–7899, 2020. [1, 3](#)
- [9] Tarik Dzanic, Karan Shah, and Freddie Witherden. Fourier spectrum discrepancies in deep network generated images. *arXiv preprint arXiv:1911.06465*, 2019. [1, 3](#)
- [10] Joel Frank, Thorsten Eisenhofer, Lea Schönherr, Asja Fischer, Dorothea Kolossa, and Thorsten Holz. Leveraging frequency analysis for deep fake image recognition. *arXiv preprint arXiv:2003.08685*, 2020. [1, 3](#)
- [11] 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*, pages 2672–2680, 2014. [1, 2, 6](#)
- [12] Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572*, 2014. [3](#)
- [13] Jinjin Gu, Yujun Shen, and Bolei Zhou. Image processing using multi-code gan prior. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3012–3021, 2020. [1](#)
- [14] Ishaan Gulrajani, Faruk Ahmed, Martin Arjovsky, Vincent Dumoulin, and Aaron C Courville. Improved training of wasserstein gans. In *Advances in Neural Information Processing Systems*, pages 5767–5777, 2017. [3](#)
- [15] 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. *Advances in Neural Information Processing Systems*, 30:6626–6637, 2017. [7](#)
- [16] Tero Karras, Timo Aila, Samuli Laine, and Jaakko Lehtinen. Progressive growing of gans for improved quality, stability, and variation. *arXiv preprint arXiv:1710.10196*, 2017. [3](#)
- [17] Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4401–4410, 2019. [1, 3](#)
- [18] 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*, pages 8110–8119, 2020. [1, 3](#)
- [19] Mahyar Khayatkhoei and Ahmed Elgammal. Spatial frequency bias in convolutional generative adversarial networks. *arXiv preprint arXiv:2010.01473*, 2020. [1, 3](#)
- [20] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009. [7](#)
- [21] Kwot Sin Lee, Ngoc-Trung Tran, and Ngai-Man Cheung. Infomax-gan: Improved adversarial image generation via information maximization and contrastive learning. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 3942–3952, 2020. [3](#)
- [22] Ziqiang Li. Direct adversarial training for gans. *arXiv preprint arXiv:2008.09041*, 2020. [3](#)
- [23] Ziqiang Li, Rentuo Tao, and Bin Li. Regularization and normalization for generative adversarial networks: A review. *arXiv preprint arXiv:2008.08930*, 2020. [3](#)

[24] Jae Hyun Lim and Jong Chul Ye. Geometric gan. *arXiv preprint arXiv:1705.02894*, 2017. 2, 6

[25] Xudong Mao, Qing Li, Haoran Xie, Raymond YK Lau, Zhen Wang, and Stephen Paul Smolley. Least squares generative adversarial networks. In *Proceedings of the IEEE international conference on computer vision*, pages 2794–2802, 2017. 6

[26] Takeru Miyato, Toshiki Kataoka, Masanori Koyama, and Yuichi Yoshida. Spectral normalization for generative adversarial networks. *arXiv preprint arXiv:1802.05957*, 2018. 1, 2, 3

[27] Sebastian Nowozin, Botond Cseke, and Ryota Tomioka. f-gan: Training generative neural samplers using variational divergence minimization. In *Advances in Neural Information Processing Systems*, pages 271–279, 2016. 2

[28] 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 *International Conference on Machine Learning*, pages 5301–5310. PMLR, 2019. 3

[29] Tim Salimans, Ian Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford, and Xi Chen. Improved techniques for training gans. *Advances in Neural Information Processing Systems*, 29:2234–2242, 2016. 7

[30] Yujun Shen, Jinjin Gu, Xiaou Tang, and Bolei Zhou. Interpreting the latent space of gans for semantic face editing. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 9243–9252, 2020. 1

[31] Yong-Goo Shin, Min-Cheol Sagong, Yoon-Jae Yeo, Seung-Wook Kim, and Sung-Jea Ko. Pepsi++: Fast and lightweight network for image inpainting. *IEEE Transactions on Neural Networks and Learning Systems*, 2020. 1

[32] Matthew Tancik, Pratul P Srinivasan, Ben Mildenhall, Sara Fridovich-Keil, Nithin Raghavan, Utkarsh Singhal, Ravi Ramamoorthi, Jonathan T Barron, and Ren Ng. Fourier features let networks learn high frequency functions in low dimensional domains. *arXiv preprint arXiv:2006.10739*, 2020. 3

[33] Rentuo Tao, Ziqiang Li, Renshuai Tao, and Bin Li. Resattrgan: Unpaired deep residual attributes learning for multi-domain face image translation. *IEEE Access*, 7:132594–132608, 2019. 1

[34] Haohan Wang, Xindi Wu, Zeyi Huang, and Eric P Xing. High-frequency component helps explain the generalization of convolutional neural networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8684–8694, 2020. 1, 3

[35] Zhiqin John Xu. Understanding training and generalization in deep learning by fourier analysis. *arXiv preprint arXiv:1808.04295*, 2018. 3

[36] Zhi-Qin John Xu. Frequency principle in deep learning with general loss functions and its potential application. *arXiv preprint arXiv:1811.10146*, 2018. 3

[37] Zhi-Qin John Xu, Yaoyu Zhang, Tao Luo, Yanyang Xiao, and Zheng Ma. Frequency principle: Fourier analysis sheds light on deep neural networks. *arXiv preprint arXiv:1901.06523*, 2019. 3

[38] Zhi-Qin John Xu, Yaoyu Zhang, and Yanyang Xiao. Training behavior of deep neural network in frequency domain. In

*International Conference on Neural Information Processing*, pages 264–274. Springer, 2019. 3

[39] Fisher Yu, Ari Seff, Yinda Zhang, Shuran Song, Thomas Funkhouser, 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. 7

[40] Xu Zhang, Svebor Karaman, and Shih-Fu Chang. Detecting and simulating artifacts in gan fake images. In *2019 IEEE International Workshop on Information Forensics and Security (WIFS)*, pages 1–6. IEEE, 2019. 1, 3

[41] Yaoyu Zhang, Zhi-Qin John Xu, Tao Luo, and Zheng Ma. Explicitizing an implicit bias of the frequency principle in two-layer neural networks. *arXiv preprint arXiv:1905.10264*, 2019. 3

## References

[1] Martin Arjovsky, Soumith Chintala, and Léon Bottou. Wasserstein gan. *arXiv preprint arXiv:1701.07875*, 2017. 2

[2] Ronen Basri, David Jacobs, Yoni Kasten, and Shira Kritchman. The convergence rate of neural networks for learned functions of different frequencies. *arXiv preprint arXiv:1906.00425*, 2019. 3

[3] Mikołaj Bińkowski, Dougal J Sutherland, Michael Arbel, and Arthur Gretton. Demystifying mmd gans. *arXiv preprint arXiv:1801.01401*, 2018. 7

[4] Yuan Cao, Zhiying Fang, Yue Wu, Ding-Xuan Zhou, and Quanquan Gu. Towards understanding the spectral bias of deep learning. *arXiv preprint arXiv:1912.01198*, 2019. 3

[5] Ting Chen, Xiaohua Zhai, Marvin Ritter, Mario Lucic, and Neil Houlsby. Self-supervised gans via auxiliary rotation loss. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 12154–12163, 2019. 3

[6] Yuanqi Chen, Ge Li, Cece Jin, Shan Liu, and Thomas Li. Ssd-gan: Measuring the realness in the spatial and spectral domains. *arXiv preprint arXiv:2012.05535*, 2020. 1, 2, 3

[7] Adam Coates, Andrew Ng, and Honglak Lee. An analysis of single-layer networks in unsupervised feature learning. In *Proceedings of the International Conference on Artificial Intelligence and Statistics*, pages 215–223, 2011. 7

[8] 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*, pages 7890–7899, 2020. 1, 3

[9] Tarik Dzanic, Karan Shah, and Freddie Witherden. Fourier spectrum discrepancies in deep network generated images. *arXiv preprint arXiv:1911.06465*, 2019. 1, 3

[10] Joel Frank, Thorsten Eisenhofer, Lea Schönherr, Asja Fischer, Dorothea Kolossa, and Thorsten Holz. Leveraging frequency analysis for deep fake image recognition. *arXiv preprint arXiv:2003.08685*, 2020. 1, 3

[11] 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, pages 2672–2680, 2014. 1, 2, 6
- [12] Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572*, 2014. 3
- [13] Jinjin Gu, Yujun Shen, and Bolei Zhou. Image processing using multi-code gan prior. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3012–3021, 2020. 1
- [14] Ishaan Gulrajani, Faruk Ahmed, Martin Arjovsky, Vincent Dumoulin, and Aaron C Courville. Improved training of wasserstein gans. In *Advances in Neural Information Processing Systems*, pages 5767–5777, 2017. 3
- [15] 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. *Advances in Neural Information Processing Systems*, 30:6626–6637, 2017. 7
- [16] Tero Karras, Timo Aila, Samuli Laine, and Jaakko Lehtinen. Progressive growing of gans for improved quality, stability, and variation. *arXiv preprint arXiv:1710.10196*, 2017. 3
- [17] Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4401–4410, 2019. 1, 3
- [18] 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*, pages 8110–8119, 2020. 1, 3
- [19] Mahyar Khayatkhoei and Ahmed Elgammal. Spatial frequency bias in convolutional generative adversarial networks. *arXiv preprint arXiv:2010.01473*, 2020. 1, 3
- [20] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009. 7
- [21] Kwot Sin Lee, Ngoc-Trung Tran, and Ngai-Man Cheung. Infomax-gan: Improved adversarial image generation via information maximization and contrastive learning. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 3942–3952, 2020. 3
- [22] Ziqiang Li. Direct adversarial training for gans. *arXiv preprint arXiv:2008.09041*, 2020. 3
- [23] Ziqiang Li, Rentuo Tao, and Bin Li. Regularization and normalization for generative adversarial networks: A review. *arXiv preprint arXiv:2008.08930*, 2020. 3
- [24] Jae Hyun Lim and Jong Chul Ye. Geometric gan. *arXiv preprint arXiv:1705.02894*, 2017. 2, 6
- [25] Xudong Mao, Qing Li, Haoran Xie, Raymond YK Lau, Zhen Wang, and Stephen Paul Smolley. Least squares generative adversarial networks. In *Proceedings of the IEEE international conference on computer vision*, pages 2794–2802, 2017. 6
- [26] Takeru Miyato, Toshiki Kataoka, Masanori Koyama, and Yuichi Yoshida. Spectral normalization for generative adversarial networks. *arXiv preprint arXiv:1802.05957*, 2018. 1, 2, 3
- [27] Sebastian Nowozin, Botond Cseke, and Ryota Tomioka. f-gan: Training generative neural samplers using variational divergence minimization. In *Advances in Neural Information Processing Systems*, pages 271–279, 2016. 2
- [28] 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 *International Conference on Machine Learning*, pages 5301–5310. PMLR, 2019. 3
- [29] Tim Salimans, Ian Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford, and Xi Chen. Improved techniques for training gans. *Advances in Neural Information Processing Systems*, 29:2234–2242, 2016. 7
- [30] Yujun Shen, Jinjin Gu, Xiaoou Tang, and Bolei Zhou. Interpreting the latent space of gans for semantic face editing. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 9243–9252, 2020. 1
- [31] Yong-Goo Shin, Min-Cheol Sagong, Yoon-Jae Yeo, Seung-Wook Kim, and Sung-Jea Ko. Pepsi++: Fast and lightweight network for image inpainting. *IEEE Transactions on Neural Networks and Learning Systems*, 2020. 1
- [32] Matthew Tancik, Pratul P Srinivasan, Ben Mildenhall, Sara Fridovich-Keil, Nithin Raghavan, Utkarsh Singhal, Ravi Ramamoorthi, Jonathan T Barron, and Ren Ng. Fourier features let networks learn high frequency functions in low dimensional domains. *arXiv preprint arXiv:2006.10739*, 2020. 3
- [33] Rentuo Tao, Ziqiang Li, Renshuai Tao, and Bin Li. Resattrgan: Unpaired deep residual attributes learning for multi-domain face image translation. *IEEE Access*, 7:132594–132608, 2019. 1
- [34] Haohan Wang, Xindi Wu, Zeyi Huang, and Eric P Xing. High-frequency component helps explain the generalization of convolutional neural networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8684–8694, 2020. 1, 3
- [35] Zhiqin John Xu. Understanding training and generalization in deep learning by fourier analysis. *arXiv preprint arXiv:1808.04295*, 2018. 3
- [36] Zhi-Qin John Xu. Frequency principle in deep learning with general loss functions and its potential application. *arXiv preprint arXiv:1811.10146*, 2018. 3
- [37] Zhi-Qin John Xu, Yaoyu Zhang, Tao Luo, Yanyang Xiao, and Zheng Ma. Frequency principle: Fourier analysis sheds light on deep neural networks. *arXiv preprint arXiv:1901.06523*, 2019. 3
- [38] Zhi-Qin John Xu, Yaoyu Zhang, and Yanyang Xiao. Training behavior of deep neural network in frequency domain. In *International Conference on Neural Information Processing*, pages 264–274. Springer, 2019. 3
- [39] Fisher Yu, Ari Seff, Yinda Zhang, Shuran Song, Thomas Funkhouser, 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. 7
- [40] Xu Zhang, Svebor Karaman, and Shih-Fu Chang. Detecting and simulating artifacts in gan fake images. In *2019 IEEE International Workshop on Information Forensics and Security (WIFS)*, pages 1–6. IEEE, 2019. 1, 3

- [41] Yaoyu Zhang, Zhi-Qin John Xu, Tao Luo, and Zheng Ma. Explicitizing an implicit bias of the frequency principle in two-layer neural networks. *arXiv preprint arXiv:1905.10264*, 2019. 3

## A. Supplemental material

### A.1. The response of the discriminator to the generated images on CIFAR-10 dataset.

We demonstrate the response of the discriminator to the generated images on CIFAR-10 dataset in Figure 9, where generated, generated\_low, generated\_fake, generated\_real, and generated\_high represent  $x_g$ ,  $x_g^l$ ,  $x_g^f$ ,  $x_g^r$ , and  $x_g^h$  of the Eq (4). Similar to the response of real images, the accuracy of  $x_g^f$  is greater than that of  $x_g^r$ , no matter what  $R$  is.

### A.2. The Details of Datasets and Experiment Settings

CIFAR-10 and CIFAR-100 comprise 50K training images and 10K test images with a spatial resolution of  $32 \times 32$ . Thus we train GAN models and compute Train FID on the training dataset with 50K images. Certainly, Test FID is calculated on the test dataset with 10K images. STL-10 is a similar dataset that contains 100K unlabeled images and 8K test images at  $96 \times 96$  resolution. We resize them to  $48 \times 48$ . Thus we train GAN models on the whole unlabeled dataset and compute Train FID on the unlabeled dataset with random 50K images. Certainly, Test FID is calculated on the test dataset with 8K images. LSUN-Bedroom has approximately 3M images. We took out 10K of them as a test set which is used to calculate the Test FID. Certainly, we compute Train FID on the training dataset with 50K random images.

Furthermore, all training parameters are chosen to give the best results in the baseline network.. The details can be found in Table 4, where the Adam parameters are Learning Rate (LR=2e-4),  $\beta_1=0$  and  $\beta_2=0.9$ ;  $n_{dis}$  is number of discriminator steps per generator step.

### A.3. Testing KID of popular models on different datasets.

To investigate the effect of different setting of Radius (R), Figure 10 illustrates the Testing KID results of different Radius (R) on four datasets. Figure 10 shows that the proposed methods (HFF and HFC) improve the quality of the generated images on different datasets and different models which is similar to Figure 8.

### A.4. The Samples of Generated Images

For qualitative comparisons, we present randomly sampled, non-cherry picked images generated by SSGAN and SSGAN with HFF for CIFAR-10, CIFAR-100, STL-10, and LSUN-Bedroom in Figures 11 and 12. We qualitatively observe that the images are more diverse and have higher quality after the usage of HFF.

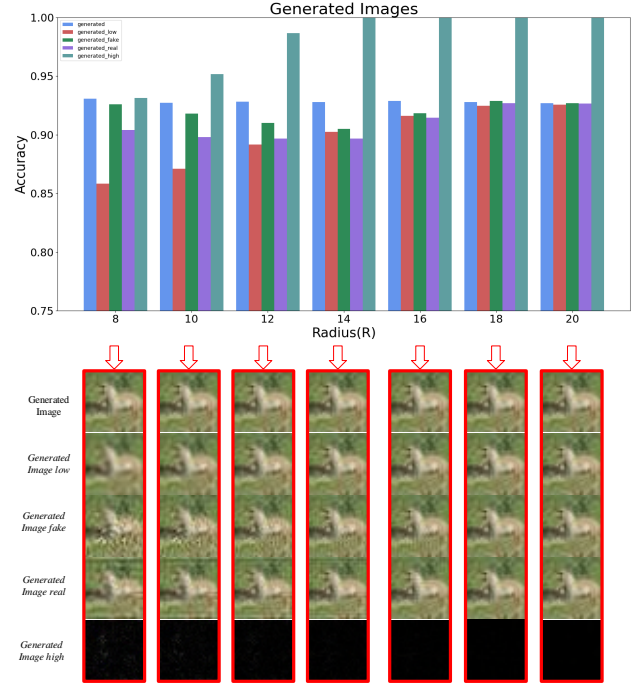


Figure 9: The response of the discriminator to the generated images on CIFAR-10 dataset. **Top:** the accuracy of the trained discriminator for the generated images (The model is SNGAN trained by Hinge Loss. The output of the discriminator goes through the Sigmoid function. Output of the Sigmoid greater than 0.5 means the image is real and less than 0.5 means the image is fake.). **Bottom:** visualization of different frequency components in the generated images



Table 4: Training parameters on different datasets

| Dataset      | Resolution | Batch Size | Learning Rate | $\beta_1$ | $\beta_2$ | Decay Policy | $n_{dis}$ |
|--------------|------------|------------|---------------|-----------|-----------|--------------|-----------|
| CIFAR-10     | 32 x 32    | 64         | 2e-4          | 0.0       | 0.9       | Linear       | 5         |
| CIFAR-100    | 32 x 32    | 64         | 2e-4          | 0.0       | 0.9       | Linear       | 5         |
| STL-10       | 48 x 48    | 64         | 2e-4          | 0.0       | 0.9       | None         | 5         |
| LSUN-Bedroom | 64 x 64    | 64         | 2e-4          | 0.0       | 0.9       | Linear       | 5         |

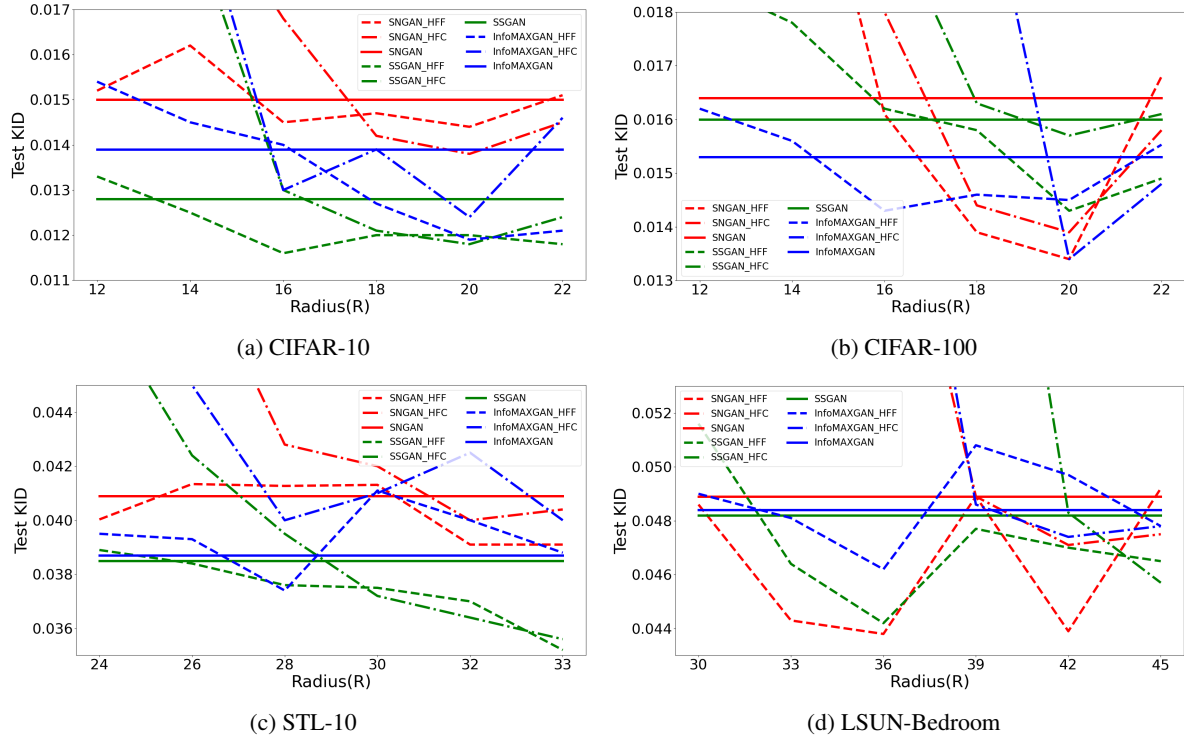
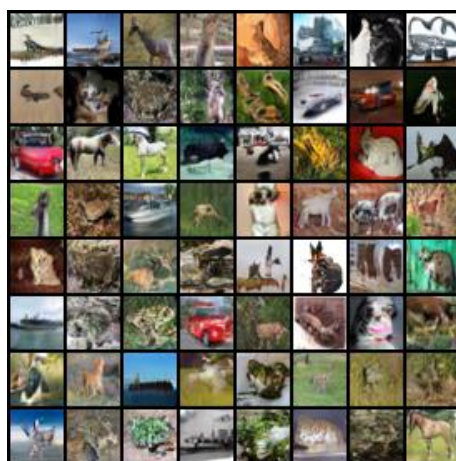
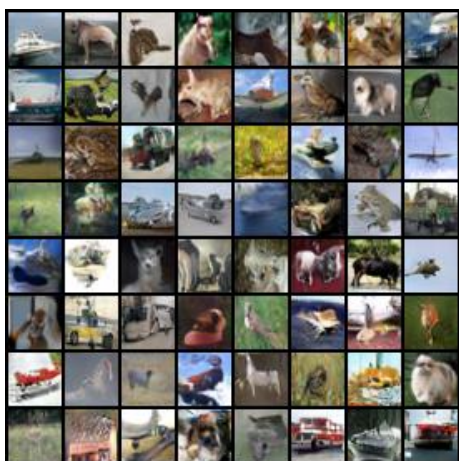
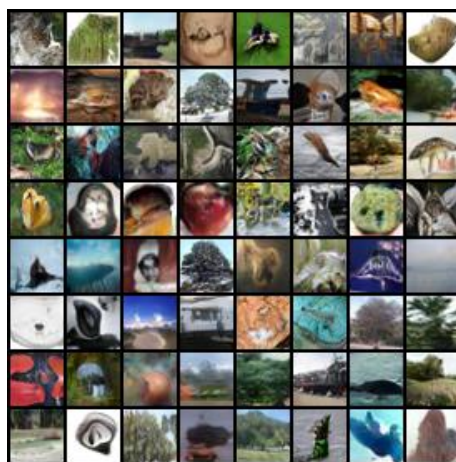


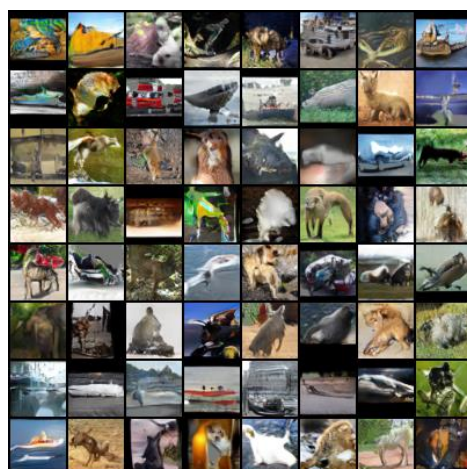
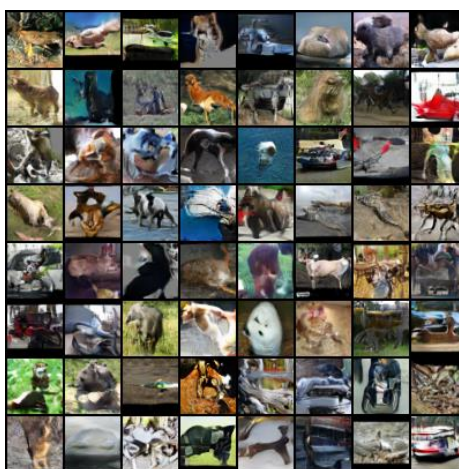
Figure 10: Testing KID results for different Radius (R) on different datasets



(a) CIFAR10 ( $R = 20$ )



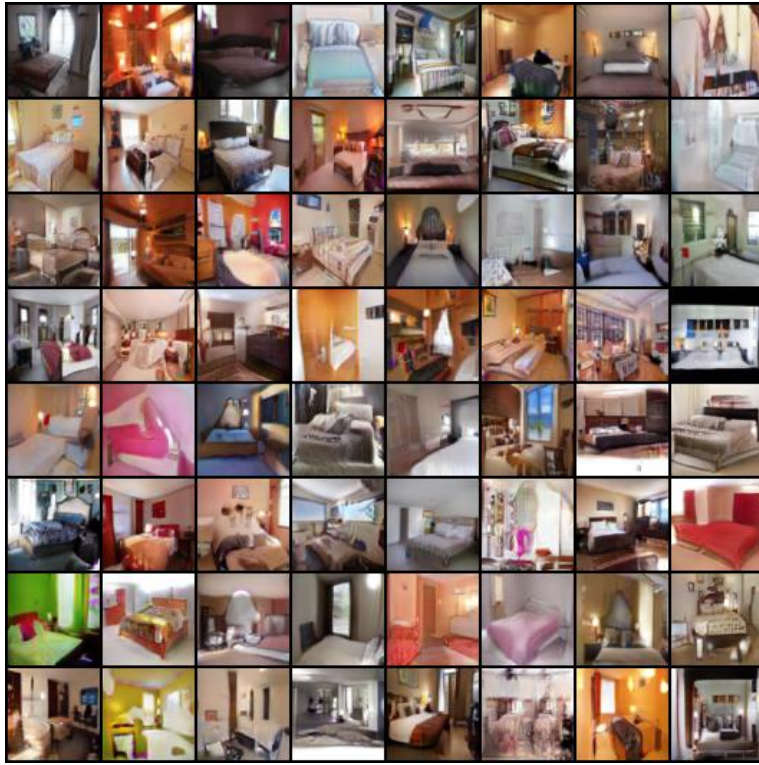
(b) CIFAR100 ( $R = 20$ )



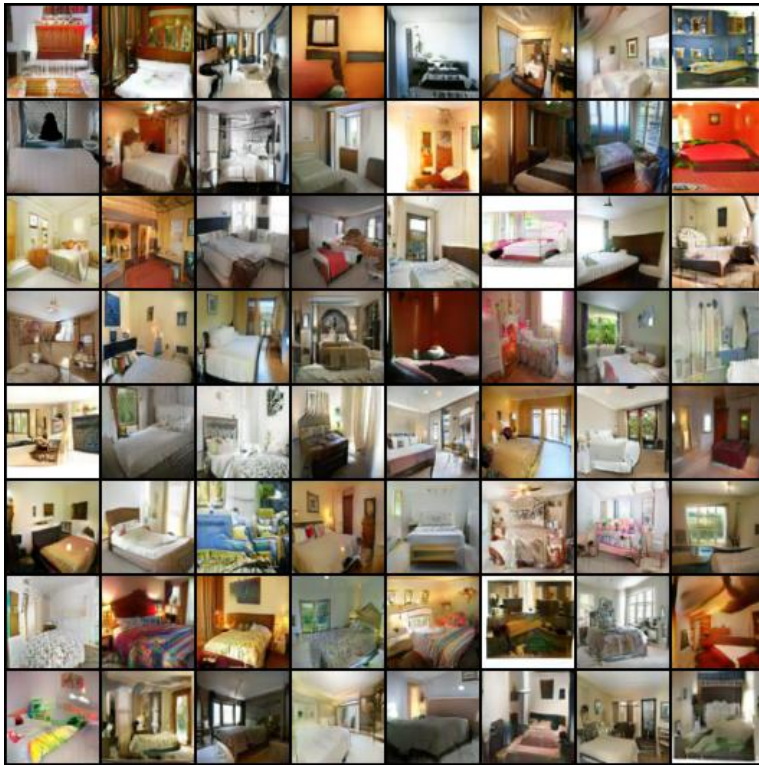
(c) STL10 ( $R = 33$ )

Figure 11: Randomly sampled and non-cherry picked images for SSGAN (left) and SSGAN with HFF (right) on CIFAR-10, CIFAR-100, and STL-10 datasets, where the Radius ( $R$ ) of HFF is 20 on CIFAR-10 dataset, 20 on CIFAR-100 dataset and 33 on STL-10 dataset.





(a) SSGAN



(b) SSGAN with HFF ( $R = 33$ )

Figure 12: Randomly sampled and non-cherry picked images for SSGAN (top) and SSGAN with HFF (bottom) for LSUN-Bedroom dataset.