Re-evaluation on the BigGAN and SNGAN for image reconstruction with small dataset

Zhuoheng Huang
University of Toronto
angeloschachter.huang@mail.utoronto.ca
Junming Zhang
University of Toronto
junmingpeter.zhang@mail.utoronto.ca
Yuchen Tong
University of Toronto
yuchen.tong@mail.utoronto.ca

Abstract

In this paper, we propose a re-evaluation on two famous GAN networks for their performance on image generation. Normally, considering the desired output qualities, we will need to train large datasets to obtain meaningful experimental data for further investigation. However, with this amount of data, it is not easy for individuals to complete the training since it needs a huge volume of samples to train over thousands of epochs and the process consumes a huge portion of computing power. In the situation where only limited computing resources are provided, we intend to reassess the two networks, BigGAN and SNGAN, using their pre-trained models, in order to reach some conclusions on which model to choose with limited training samples.

1   Introduction

These days, image reconstruction and synthesis have seen their increasing demand in the industry, and quite a lot of generative models are constantly being developed and optimized, but they requirs extremely large datasets to train. To make the process easier, batch statistics adaptation is introduced to reduce the amount of training samples and time needed.

However, the method is applied on specific GAN models, and the selection on the models has a bearing on the comparative image qualities when images from different classes are trained and tested. Therefore, when there is a limited training samples to be used, it is important to make a wise decision on which model to use for sets of images, and it expands another ground for a research to be conducted.

2   Related Work

Referring to Noguchi et al and recent research outcomes, SNGAN and BigGAN could be two candidates for experiments on image reconstruction with small amount of data. They have their own GAN structures and uniqueness and show distinctive advantages on image generation. In the limited data regimes, Kerras et al proposed an adaptive discriminator augmentation mechanism which can significantly stabilizes the training. This mechanism is proven to be applicable both when training from scratch and when fine-tuning an existing GAN [5]. Wang et al. [7] studied domain adaptation applied to image generation with GAN and concluded that using knowledge from pre-trained networks can shorten the convergence time and improve the quality of generated images

when target data size is small. Genin et al. [3] proposed gradient reversal layers for learning domain in variant features. Tzeng et al. [6] focused on studying the correlation between the classes of the source domain and use it to transfer task information.

3 Experiments

We used the pretrained SNGAN and BigGAN models from author's repository and further trained the scale and shift parameters with three sets of images: flowers, animes and human faces. There are only around 30-50 images for each of them but they represent distinctive characteristics from their own.

3.1 Theory

Regarding the scale and shift parameters, they are both trainable and are the only parameters updated during our training process. Here the two parameters have the following allocation:

$$G_{Adapt}^{(l)} = G^{(l)} \cdot \gamma^{(l)} + \beta^{(l)}$$

where $G^{(l)}$ is the feature representation of the lth layer of the generator, $G_{Adapt}^{(l)}$ is the feature of the lth layer after adaptation. γ and β represent scale and shift coefficient respectively, which are initialized as 1 and 0. Within the generator structure, γ infers the activation strength of the filters while β serves as the threshold. When they are both large, the corresponding neuron becomes easy to activate while the neurons are less active with small γ and β values.

3.2 Training

We used the following image data for the BigGAN and SNGAN, where the BigGAN and SNGAN pretrained weights are from the original author's repository. The 50 face images are from FFHQ dataset, 50 anime faces are from the author's repository and we use 32 images as the flower dataset from TF-Flowers.

Our focus is the reconstruction, and for each dataset, we trained for 500, 2000, 5000 iterations. For two models, the generators are trained using the supervised learning. Similar to Noguchi et al, we used the model embedding method to estimate the latent variable for all training data such that the generated data is close to the images in the training dataset.

The loss is defined as the following, and we use SGD to update

$$\text{L1 Loss} = \sum_i \frac{1}{c_x h_x \omega_x} ||x_i - G_{Adapt}(z_i + \epsilon)||$$

$$\text{Perceptual Loss} = \sum_i \sum_{l \in layers} \frac{\lambda_C^l}{c_l h_l \omega_l} ||C^{(l)}(x_i) - C^{(l)}(G_{Adapt}(z_i + \epsilon))||$$

$$\textbf{Loss} = \text{L1 Loss} + \text{Perceptual Loss}$$

3.3 Evaluation

We use Frechet Inception Distance (FID) and Kernel Maximum Mean Distance (KMMD) for as evaluation metrics. For each dataset, we calculated the FIDs and KMMDs of reconstructed results for iterations 500, 2000, 5000.

3.3.1 FID

The Frechet Inception Distance is a metric that calculates the distance between feature vectors calculated for real and generated images, and lowers scores indicate higher level of similarity. We used `fid_score` in scipy package to calculate the values. The input and reconstructed images are parsed into the calculation method in batches and corresponding FID scores will be printed.

3.3.2 KMMD

Kernel Maximum Mean Discrepancy measures how similar two distributions are. in the context of image generation via GAN, it evaluates the correlation between the original images and the generated ones. Like FID, a lower KMMD indicates a better generated result. We use a Gaussian kernel for the calculation and use `kmmd` in thr package `kernlab` of R to do the implementation.

4 Results

We tested two models, BigGan and SNGAN for training with different numbers of iterations: 500, 2000, and 5000 with two scales: FID and KMMD. The FID test simply has FID scores as test results, and the KMMD test consists of the following components:

1. Hypothesis test with null hypothesis: two images come from the same distribution [8]
2. Mean and median of 1st and 3rd ordered KMMD scores for input and output images

The FID and KMMD test results are listed in the following two subsections. For detailed illustration of KMMD, such as distribution of KMMD T-test statistics and null hypothesis acceptance and rejection comparison, please read figures in the appendix section.

4.1 FID

The tables of FID scores gained by two models trained with different numbers of iterations on different datasets are drawn. Please see the **Appendix** for table 1, 2, 3

Three tables show that, generally speaking, the FID scores of BigGan are much higher than those of SNGAN, even if in a few cases SNGAN gains higher FID score, but only slightly higher than that gained by BigGAN score. Therefore, from the perspective of FID, the SNGAN is more likely to produce high-quality result compared to BigGAN.

The changing trend of FID scores for two models are invariant with respect to datasets as the iteration number increases. For the SNGAN model, the FID score decreases as the iteration number increases for all three datasets. This decreasing trend is mild on the anime dataset, but rapid on the face dataset.

For the BigGAN model, the FID score decreases initially when iteration number increases from 500 to 2000, and then slightly increases when the iteration number increases from 2000 to 5000. This decreasing trend is mild on the flower dataset, but rapid on the face and anime dataset.

4.2 KMMD

Please refer to **Appendix** for table 4, 5, 6 for KMMD hypothesis test H_0 acceptance rates on three datasets, and table 7, 8, 9 for 1st and 3rd mean and median KMMD statistics on three datasets.

The tables of KMMD statistics gained by two models trained with different numbers of iterations on different datasets are displayed below. The KMMD statistics include the null hypothesis acceptance rate, the mean and the median of first and third order KMMD test statistics. The null hypothesis in the KMMD hypothesis test is that two models are from the same distribution. We reject the null hypothesis when the probabilities computed from the two sample T-test with the KMMD statistics in the tables 7, 8, 9 are lower than the significance level, $\alpha = 0.05$. The less possible the null hypothesis rejected, the more probable the model can generate the high-quality output.

Because of the package implementation, the test function `kmmd` outputs the KMMD statistics, which is used for the hypothesis two sample T-test, rather than the KMMD scores. Therefore, the higher the KMMD statistics, the less possible the null hypothesis is rejected, hence the more probable the model can generate the high-quality output. The two sample T-test statistics is related to the null hypothesis acceptance rate, the higher the T-test statistics, the harder the null hypothesis is rejected.

There are first and third order statistics, they are computed in different ways, but they have the same result for the two sample T-test.

The tables show that, in general, the images generated by BigGAN are more possible to pass the KMMD test than SNGAN. Since the null hypothesis acceptance rates and mean and median KMMD

statiscs of BigGAN are generally higher (sometimes much higher) than those of SNGAN (there is only one case the SNGAN derives outputs with higher acceptance rate and KMMD statistics than BigGAN). Hence, the BigGAN model may generate images with closer distribution to the original images than SNGAN.

For the flower dataset, the acceptance rates (and the test statistics) of both models decrease as the iteration number increases. The test results from the face dataset are not ideal, since the acceptance rates for all cases are nearly 0, and the test statistics for both models decrease as the number of iteration increases, this means the performance on KMMD is even worse as the model is trained more. For the anime dataset, the null hypothesis acceptance rate and test statistics for both models decrease at first, and then increase as the iteration number increases. The acceptance rate and test statistics of BigGAN increases even more significantly than SNGAN. Therefore, the distribution of output images generated by both models may depend on the distribution of input images.

5   Conclusion

In our research, we compared and analyzed the performance of SNGAN and BigGAN for image reconstruction on three datasets. Our results show that, with the help of the batch statistics adaptation, the shift and scale parameters are tuned. The SNGAN and BigGAN could both generate high-quality reconstructed images by further training the models with small amount of source images. Observing the outputs of two evaluation metrics, it suggests that SNGAN yields results with better qualities evaluated by FID, while BigGAN outperforms SNGAN with KMMD being the quantified indicator.

By visual judgment on the reconstructed results, SNGAN is the winning candidate over BigGAN under fixed iterations. SNGAN generated more visually plausible outcomes at higher iterations since the outlines and contours of the target objects are clearer and more semantically interpretable.

6   Attributions

Zhuoheng Huang completed the experiment workflow implementation including the code structure, experiments outline and etc. He also refactored the BigGAN implementation, implemented part of the FID calculation method as well as helped train and obtain the BigGAN reconstruction.

Junming Zhang looked for models to implement and refactored part of SNGAN and BigGAN to produce images suitable for testing. He also designed and completed the testing part: script of KMMD and FID, and made computations and calculations.

Yuchen Tong has refectored the evaluation function in SNGAN to output images with desired format, and helped train and tested the SNGAN for image reconstruction. He was also responsible for labelling, grouping and formatting before the result evaluation stage.

Please see

Complete reseach source codes, notebooks, graphs, tables and generated results
`https://github.com/JunmingZhang/CSC413_PROJECT`

Source codes of refactored GAN models
`https://github.com/JunmingZhang/BigGAN-image-generation`
`https://github.com/JunmingZhang/SNGAN-image-generation`

References

[1] Alexander, J.A. & Mozer, M.C. (1995) Template-based algorithms for connectionist rule extraction. In G. Tesauro, D.S. Touretzky and T.K. Leen (eds.), *Advances in Neural Information Processing Systems 7*, pp. 609–616. Cambridge, MA: MIT Press.

[2] Bower, J.M. & Beeman, D. (1995) *The Book of GENESIS: Exploring Realistic Neural Models with the GEneral NEural SImulation System.* New York: TELOS/Springer–Verlag.

[3] Ganin, Y., Ustinova, E., Ajakan, H., Germain, P., Larochelle, H., Laviolette, F., Marchand, N., Lempitsky, V. (2016) Domain-adversarial training of neural networks. JMLR **17** (1), 2096-2030.

[4] Hasselmo, M.E., Schnell, E. & Barkai, E. (1995) Dynamics of learning and recall at excitatory recurrent synapses and cholinergic modulation in rat hippocampal region CA3. *Journal of Neuroscience* **15**(7):5249-5262.

[5] Karras, T., Aittala, M., Hellsten, J., Laine, S., Lehtinen, J., & Aila, T. (2020). Training generative adversarial networks with limited data.

[6] Tzeng, E., Hoffman, J., Darrell, T., Saenko, K. (2015) Simultaneous deep transfer across domains and tasks. In: CVPR. pp. 4068-4076.

[7] Wang, Y., Wu, C., Herranz, L. & Weijer, J. (2018) Transferring GANs: generating images from limited data.

[8] Gretton, A., Borgwardt, K., Rasch, M., Scholkopf, B., Smola, A. (2006) A Kernel Method for the Two-Sample-Problem

Appendix

Samples generated by SNGAN
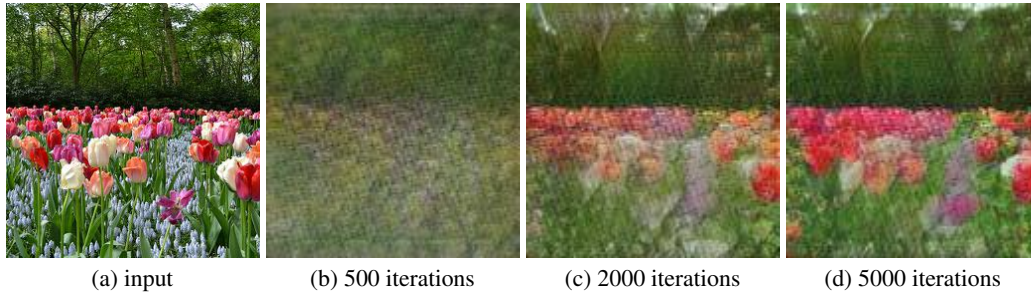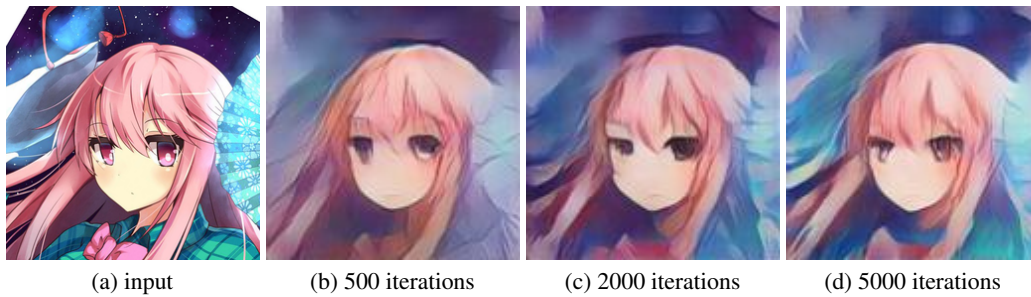
Figure 1: Dataset Flower

(a) input   (b) 500 iterations   (c) 2000 iterations   (d) 5000 iterations

Figure 2: Dataset Face

(a) input   (b) 500 iterations   (c) 2000 iterations   (d) 5000 iterations

Figure 3: Dataset Anime

(a) input   (b) 500 iterations   (c) 2000 iterations   (d) 5000 iterations

Samples generated by BigGAN

Figure 4: Dataset Flower

(a) input   (b) 500 iterations   (c) 2000 iterations   (d) 5000 iterations

Figure 5: Dataset Face

(a) input   (b) 500 iterations   (c) 2000 iterations   (d) 5000 iterations

Figure 6: Dataset Anime

(a) input   (b) 500 iterations   (c) 2000 iterations   (d) 5000 iterations
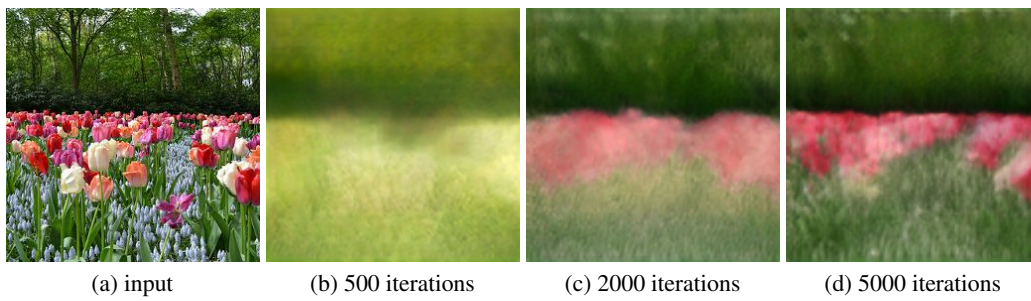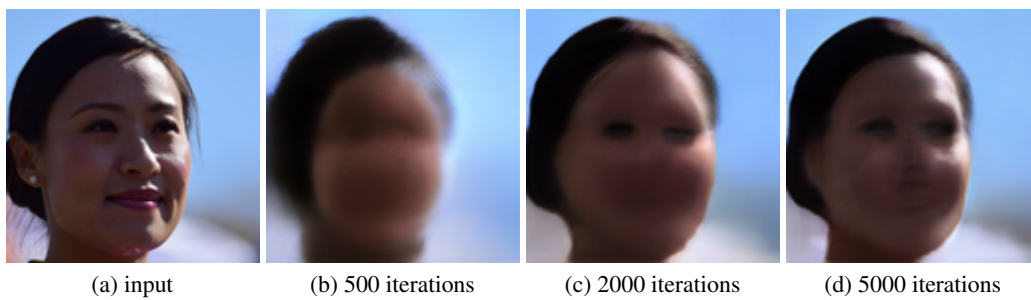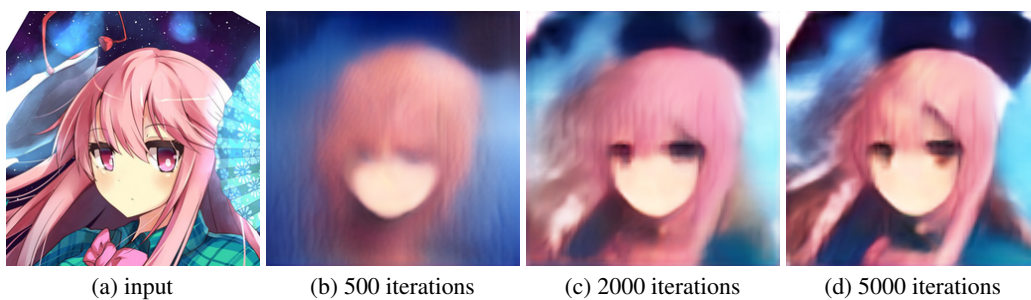
FID scores of three datasets by models

Table 1: FID scores on dataset flower

	500 iterations	2000 iterations	5000 iterations
SNGAN	332.54	254.81	222.53
BigGAN	354.25	297.03	302.93

Table 2: FID scores on dataset face

	500 iterations	2000 iterations	5000 iterations
SNGAN	336.60	226.01	218.61
BigGAN	331.01	220.89	260.48

Table 3: FID scores on dataset anime

	500 iterations	2000 iterations	5000 iterations
SNGAN	201.42	187.98	182.03
BigGAN	284.15	188.65	242.54

KMMD hypothesis test H_0 acceptance rates of three datasets by models

Table 4: KMMD hypothesis test H_0 acceptance rates on dataset flower

	500 iterations	2000 iterations	5000 iterations
SNGAN	0.75	0.50	0.25
BigGAN	0.84	0.56	0.31

Table 5: KMMD hypothesis test H_0 acceptance rates on dataset face

	500 iterations	2000 iterations	5000 iterations
SNGAN	0.00	0.02	0.00
BigGAN	0.04	0.00	0.00

Table 6: KMMD hypothesis test H_0 acceptance rates on dataset anime

	500 iterations	2000 iterations	5000 iterations
SNGAN	0.30	0.12	0.18
BigGAN	0.50	0.16	0.84

1st and 3rd mean and median KMMD statistics of three datasets by models

Table 7: 1st and 3rd mean and median KMMD statistics on dataset flower

	500 iterations	2000 iterations	5000 iterations
SNGAN 1st order	0.4678, 0.4767	0.3525, 0.3428	0.2732, 0.2450
SNGAN 3rd order	0.2376, 0.2246	0.1400, 0.1149	0.0931, 0.0581
BigGAN 1st order	0.5321, 0.4479	0.4120, 0.4007	0.2999, 0.2525
BigGAN 3rd order	0.3171, 0.1972	0.2063, 0.1575	0.1196, 0.0620

Table 8: 1st and 3rd mean and median KMMD statistics on dataset face

	500 iterations	2000 iterations	5000 iterations
SNGAN 1st order	0.2525, 0.2429	0.2082, 0.2037	0.1949, 0.1852
SNGAN 3rd order	0.0660, 0.0558	0.0459, 0.0390	0.0400, 0.0323
BigGAN 1st order	0.2749, 0.2501	0.1789, 0.1657	0.1377, 0.1240
BigGAN 3rd order	0.0817, 0.0587	0.0329, 0.0255	0.0205, 0.0138

Table 9: 1st and 3rd mean and median KMMD statistics on dataset anime

	500 iterations	2000 iterations	5000 iterations
SNGAN 1st order	0.2923, 0.2832	0.2372, 0.2326	0.2537, 0.2542
SNGAN 3rd order	0.0920, 0.0781	0.0613, 0.0523	0.0688, 0.0630
BigGAN 1st order	0.3319, 0.3414	0.2367, 0.2156	0.6939, 0.6815
BigGAN 3rd order	0.1165, 0.1140	0.0604, 0.0444	0.5035, 0.4570

Null hypothesis reject vs cannot reject for SNGAN

Figure 7: Dataset Flower

(a) 500 iterations

(b) 2000 iterations

(c) 5000 iterations

Figure 8: Dataset Face

(a) 500 iterations

(b) 2000 iterations

(c) 5000 iterations

Figure 9: Dataset Anime

(a) 500 iterations

(b) 2000 iterations

(c) 5000 iterations
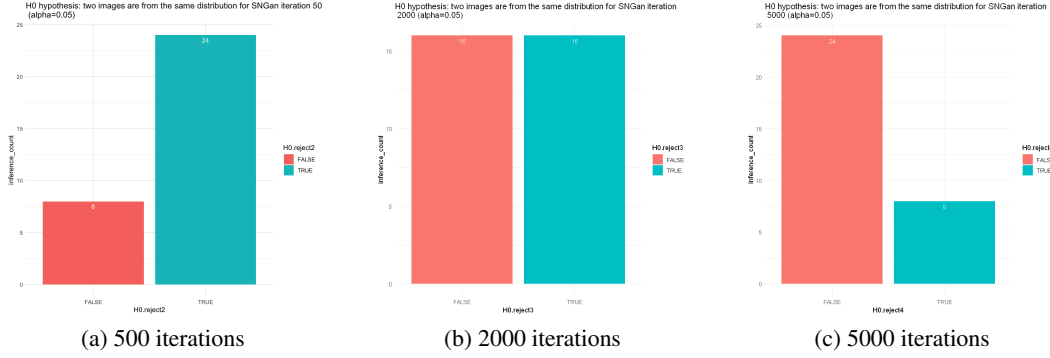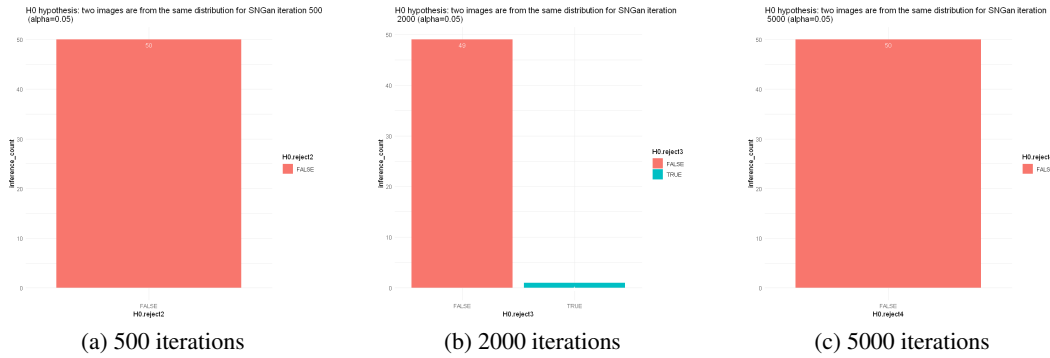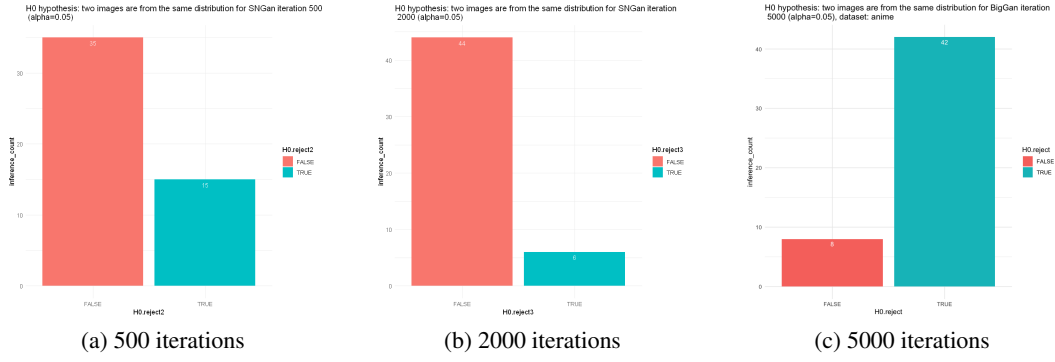
Null hypothesis reject vs cannot reject for BigGAN

Figure 10: Dataset Flower
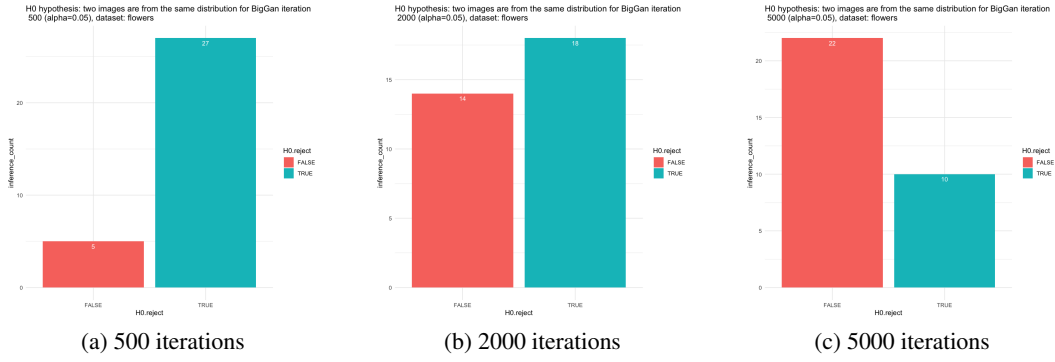
(a) 500 iterations

(b) 2000 iterations

(c) 5000 iterations

Figure 11: Dataset Face

(a) 500 iterations

(b) 2000 iterations

(c) 5000 iterations

Figure 12: Dataset Anime

(a) 500 iterations

(b) 2000 iterations

(c) 5000 iterations
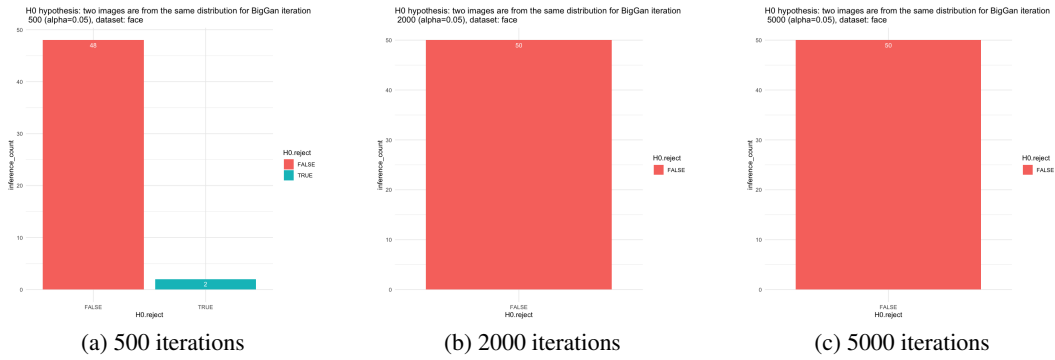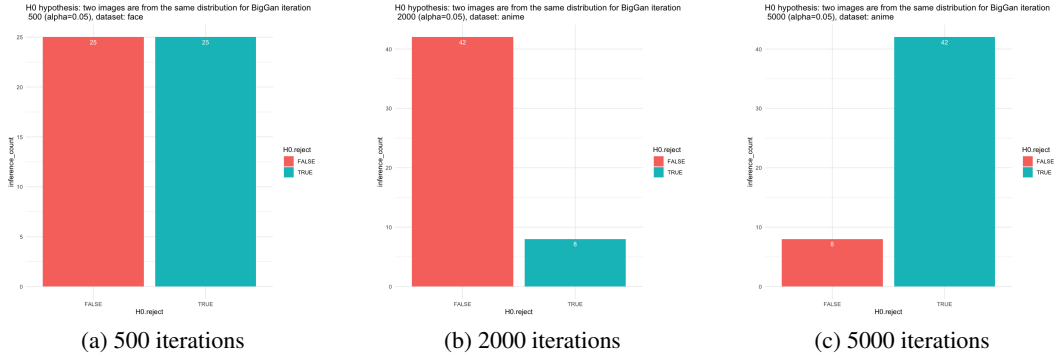
KMMD first order test statistics distribution for SNGAN

Figure 13: Dataset Flower
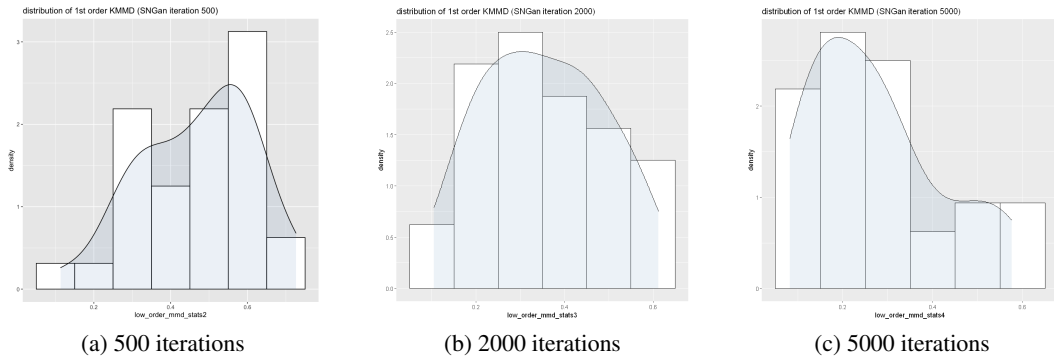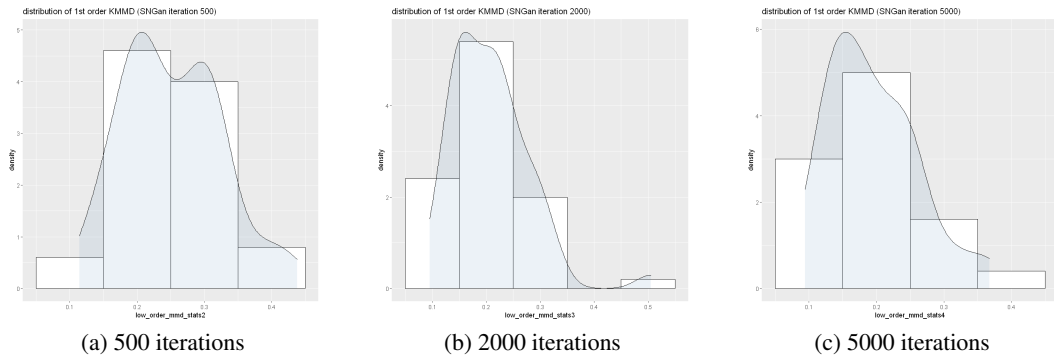
(a) 500 iterations

(b) 2000 iterations

(c) 5000 iterations

Figure 14: Dataset Face

(a) 500 iterations

(b) 2000 iterations

(c) 5000 iterations

Figure 15: Dataset Anime

(a) 500 iterations

(b) 2000 iterations

(c) 5000 iterations
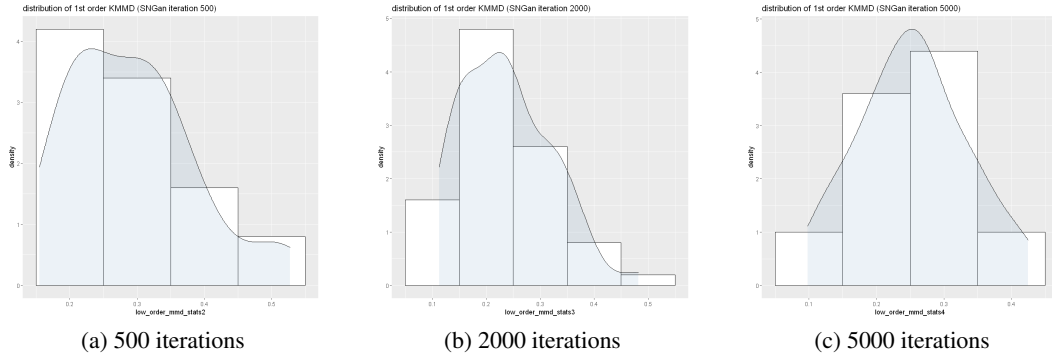
KMMD first order test statistics distribution for BigGAN

Figure 16: Dataset Flower
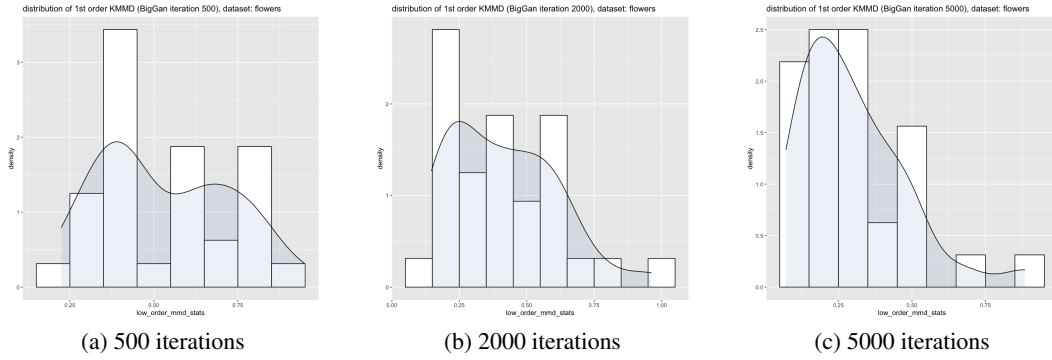
(a) 500 iterations

(b) 2000 iterations

(c) 5000 iterations

Figure 17: Dataset Face

(a) 500 iterations

(b) 2000 iterations

(c) 5000 iterations

Figure 18: Dataset Anime

(a) 500 iterations (b) 2000 iterations (c) 5000 iterations
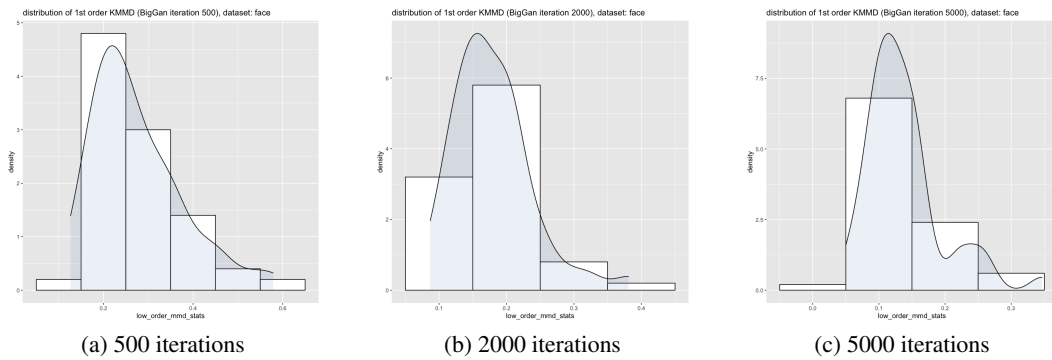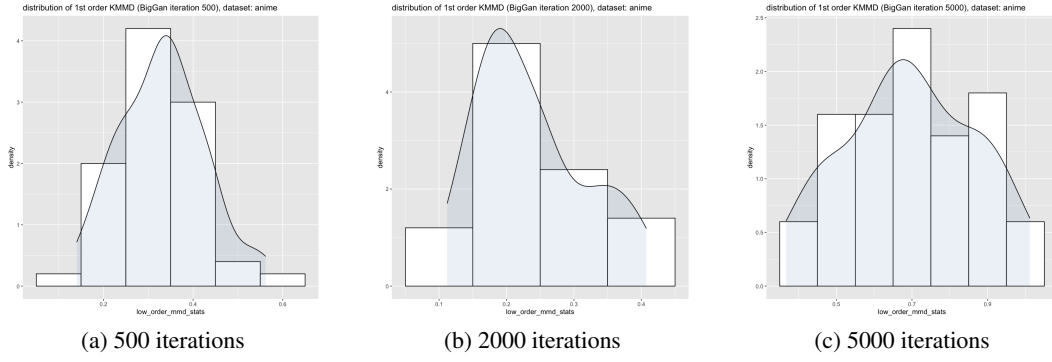
KMMD third order test statistics distribution for SNGAN

Figure 19: Dataset Flower
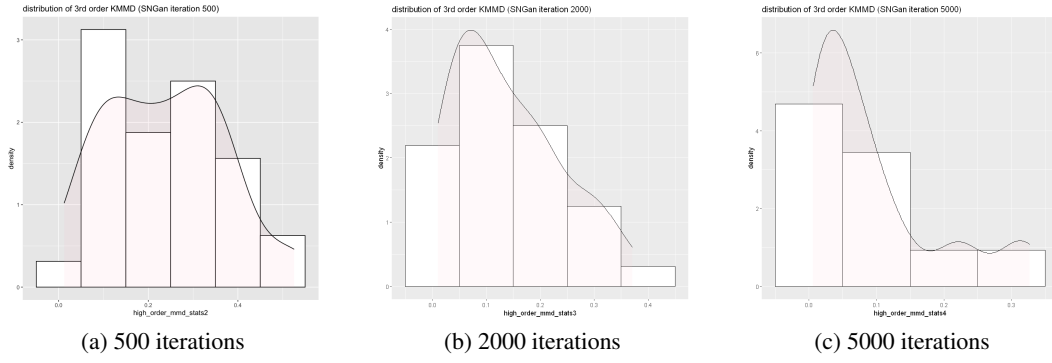
(a) 500 iterations (b) 2000 iterations (c) 5000 iterations

Figure 20: Dataset Face

(a) 500 iterations (b) 2000 iterations (c) 5000 iterations

Figure 21: Dataset Anime

(a) 500 iterations
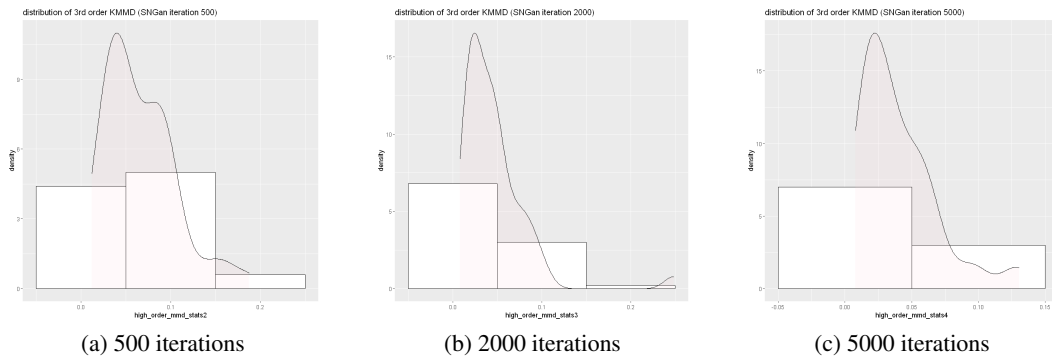
(b) 2000 iterations
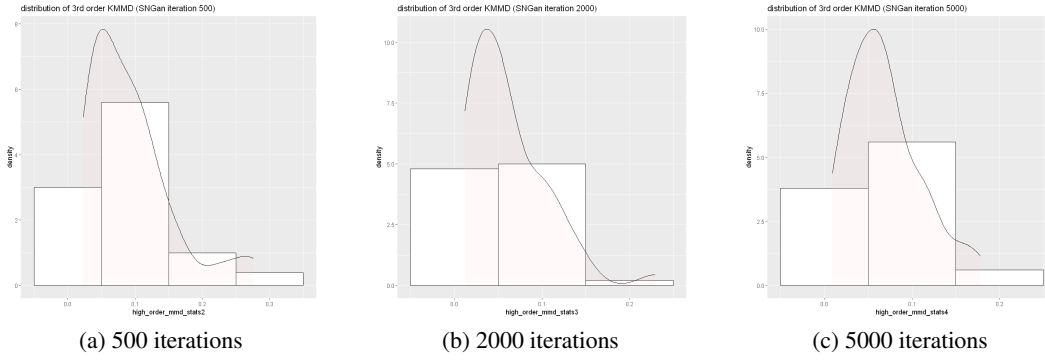
(c) 5000 iterations

KMMD third order test statistics distribution for BigGAN

Figure 22: Dataset Flower

(a) 500 iterations

(b) 2000 iterations

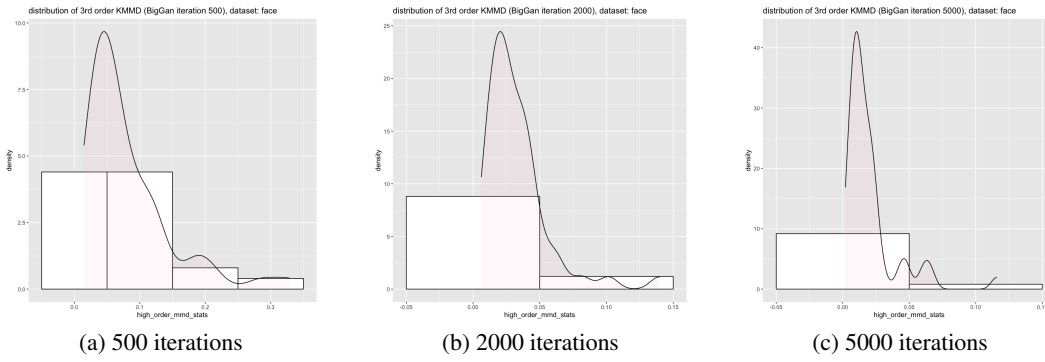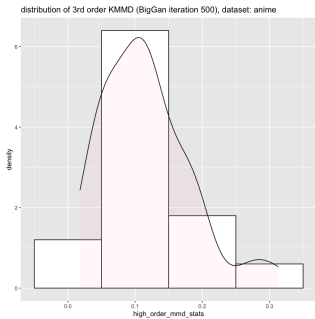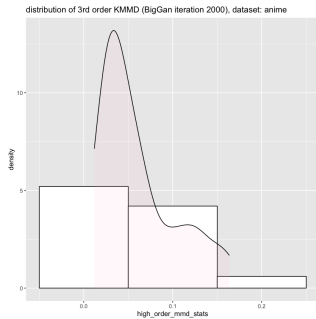(c) 5000 iterations

Figure 23: Dataset Face

(a) 500 iterations

(b) 2000 iterations

(c) 5000 iterations

Figure 24: Dataset Anime

distribution of 3rd order KMMD (BigGan iteration 500), dataset: anime

distribution of 3rd order KMMD (BigGan iteration 2000), dataset: anime

distribution of 3rd order KMMD (BigGan iteration 5000), dataset: anime
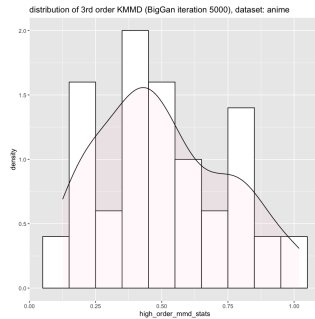
(a) 500 iterations　　　　　(b) 2000 iterations　　　　　(c) 5000 iterations