Appendix: Generic and Privacy-free Synthetic Data Generation for Pretraining GANs

Anonymous Author(s)
Affiliation
Address
email

1 Additional Experiments and Analysis

Conditional generation task using CIFAR. We conduct conditional generation via transfer learning on CIFAR-10 and 100 as summarized in Table 1. Figure 1 shows the qualitative evaluation result on CIFAR-10 with 10% of samples; our `Primitives-PS` produces the general shape and its structural components better than the baseline and DiffAug. Compared to BigGAN trained from scratch, BigGAN trained from scratch with DiffAug significantly improves the FID score, and the gain is pronounced as the number of training samples decreases. However, we observe that DiffAug suffers from augmentation leakage [1] when the samples are scarce (i.e., the generated samples contain the cutout). Our pretrained model with `Primitives-PS` shows remarkable performances under the data-hungry scenario, better than DiffAug.

However, when the samples are sufficient (100%), pretraining does not always provide gains over DiffAug. This tendency appears in various downstream tasks. Newell et. al. [2] reported that the self-supervised pretraining for semi-supervised classification is not advantageous when the amount of data-label pairs are sufficient. TransferGAN [3] showed that the gain via transfer learning decreases when the amount of samples is sufficient. In the same vein, the advantage of our pretraining with `Primitives-PS` decreases as the number of samples increases.

For the extreme low-shot scenario, we also evaluated the model trained with 1% of the dataset. Only for this evaluation, we compare three models; 1) the model naïvly trained from scratch, 2) the model trained with DiffAug only (DiffAug), and 3) our model pretrained with `Primitives-PS` and then finetuned without DiffAug. The FID score of the baseline, DiffAug, and ours are 112.13, 101.91, and 78.48, respectively. Although DiffAug improved FID, we observe that DiffAug suffers from the augmentation leakage issue. Therefore, the improvement in FID and its generation results are not meaningful. In contrast, our pretrained model can significantly improve the generation performance without any issue.

Diverse filters matter for transferring GANs. From the superior performances of our pretrained model, we conjecture that our achievement was possible by the unbiased nature of our dataset; the pretrained model with FFHQ (FreezeD) has an inductive bias as the face dataset. A previous study analyzing the transferability of CNN [4] also pointed out that the performance of the target dataset degrades when the filters are highly specialized to the source dataset. To analyze the transferability empirically, we measure the similarity between the filters of each layer of the pretrained model. We regard that highly diverse (less similar to each other) filters can indicate that the model is less biased towards a particular domain. That means that the highly transferable model tends to have low filter similarity on average. Specifically, given a weight matrix of each layer, its shape is $[O, I, H, W]$, where O filters have $I \times H \times W$ tensors. Then, we measure the cosine similarity among all possible permutations of O filters and report the average similarity of all layers in Table 3.

In summary, `Primitives-PS` shows the more diverse filter set in 21 out of 26 layers than the FFHQ pretrained model. According to [4], the higher layer (close to the output) tends to specialize in the trained dataset. The same observation holds in our discriminator. The similarity in the last layer of

Submitted to NeurIPS 2022 Workshop on Synthetic Data for Empowering ML Research. Do not distribute.

	CIFAR-10			CIFAR-100		
	10%	20%	100%	10%	20%	100%
BigGAN	44.14	20.80	9.45	66.21	34.78	13.45
+ DiffAug	29.78*	14.04	**8.55**	41.70*	21.14	11.51
+ Pretrained (PS)	**21.33**	**12.79**	8.79	**32.57**	**20.58**	**11.29**

Table 1: The FID of BigGAN, with DiffAug, and with DiffAug initialized by `Primitives-PS` (PS) pretrained model on CIFAR. '*' indicates the best FID before augmentation leakage [1].

Policy	Obama	Grumpy cat	Bridge	Panda
Fix ($1/10$)	48.30	29.74	63.00	17.69
Fix ($1/5$)	46.41	29.22	64.02	14.97
Fix ($1/2$)	48.05	29.37	64.65	15.14
`PinkNoise + PS`	49.13	29.87	66.00	15.12
Rand	44.85	29.84	60.45	14.67
Decay	**41.62**	**26.01**	**54.02**	**12.23**
# of particles	Obama	Grumpy cat	Bridge	Panda
0	49.13	29.87	66.00	15.12
10	44.10	28.00	63.26	13.35
50	42.49	28.40	59.17	**11.79**
100	**41.62**	**26.01**	54.02	12.23
500	42.45	27.92	**52.27**	12.12

Table 2: The average consine similarity between the filters in the same layer. The lower value indicates the more diverse filters.

(a) From scratch (b) + DiffAug (c) + `Primitives-PS` pretraining

Figure 1: Qualitative evaluation on CIFAR-10 dataset with 10% of samples. Each row contains samples in the same class.

the FFHQ pretrained model is approximately four times higher than `Primitives-PS`. This explains that the FFHQ pretrained model specialized in human faces, thus transferring well to Obama but not to others.

Ablation study. When developing `Primitives-PS`, we introduce two hyperparameters; 1) the total number of shapes and 2) the policy to determine the size of each component. For determining the size, we consider three policies; **Fix**, **Rand** and **Decay**. **Fix** indicates that all particles have the same size. To examine the effect of various scale, we set this size as $H \cdot [1/10, 1/5, 1/2]$, where H is the image resolution. **Rand** randomly samples the size from the uniform distribution. Both policies can induce the occlusion of the previously injected shapes by the later shape. **Decay** can bypass the occlusion issue effectively. **Decay** arbitrarily samples the size from the uniform distribution, where the maximum size is limited to $(H \cdot 1/5 \cdot (N-n)/N)$, and N and n are the total number of shapes and the number of previously injected particles. In this way, we can ensure that the shapes inserted in the early stage are still visible in the final data. The upper-side of Table 2 summarizes the FID score for each policy on four datasets. The differences in FID among **Fix** policies are trivial in that their ratios are not highly correlated with their ranks. Also, we observe that the shapes at the final stage overwrite the previous shapes. Then, the overall appearance with **Fix** are similar to `PinkNoise` with a salient object. We investigate the synthesizer that combines `PinkNoise` with `PS` by injecting a saliency and then applying `PinkNoise` on it. Interestingly, we observe that it shows the similar FID scores to **Fix**. For **Rand**, it improves the FID score on Obama and bridge, however, the overall performance is much worse than **Decay**. Therefore, we choose a **Decay** policy as default for choosing the size.

Besides, the total number of shapes is important because it affects the transferability and the time complexity of the synthesizer. The lower-side of Table 2 demonstrates the performance trends upon the total number of shapes. A zero particle case implies that only one background and one salient object, thus equivalent to `PinkNoise + PS`. As the number of shapes (N) grows upon roughly 100, the performance tends to improve. However, over $N = 100$, we do not observe the consistent gain. From the ablation study, we decide $N = 100$ in each image to enjoy the reasonable performance gain and to reduce the time complexity.

	Discriminator		Generator	
	Primitives-PS	FFHQ	Primitives-PS	FFHQ
conv0	**0.00660**	0.01245	**0.00315**	0.00685
conv1	0.02104	**0.00932**	**0.00273**	0.00843
conv2	0.01012	**0.00779**	**0.00291**	0.00956
conv3	**0.00839**	0.01216	**0.00348**	0.01080
conv4	**0.00607**	0.00713	**0.00539**	0.01059
conv5	**0.00596**	0.00668	**0.00329**	0.01406
conv6	**0.00507**	0.00563	**0.00363**	0.01199
conv7	**0.00632**	0.00714	**0.00433**	0.01465
conv8	0.00380	**0.00365**	**0.00652**	0.01317
conv9	**0.00521**	0.00703	**0.00933**	0.01626
conv10	0.00503	**0.00420**	**0.01133**	0.01778
conv11	**0.00462**	0.00760	0.01981	**0.01977**
conv12	**0.01844**	0.08438	**0.03176**	0.03250
Mean	**0.00820**	0.01348	**0.00828**	0.01434

Table 3: Ablation study on the policy to determine the size of each particle (upper) and the number of particles (lower).

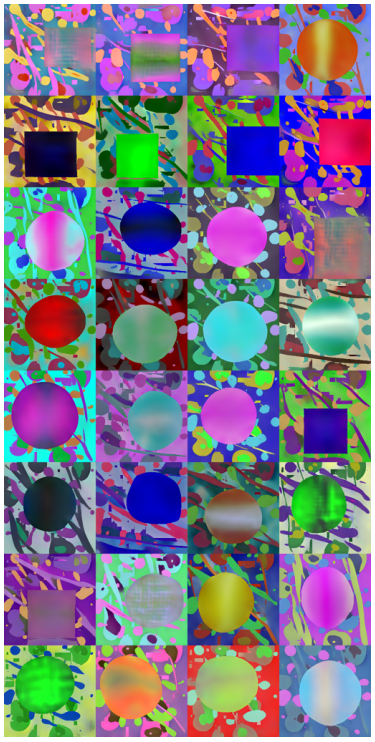

Figure 2: The outputs of the model pretrained with Primitives-PS. The generated outputs are similar to the synthetic samples.

2 Pretraining Results and Details

We provide the outputs of the generator pretrained with Primitives-PS. For pretraining, we train the model during 800K images with batch size = 16, therefore, the total number of iterations is 50K. For finetuing all the models, we train the model during 400K images. The generated (fake) synthetic images are similar to the real synthetic samples as shown in Figure 1 of the main text.

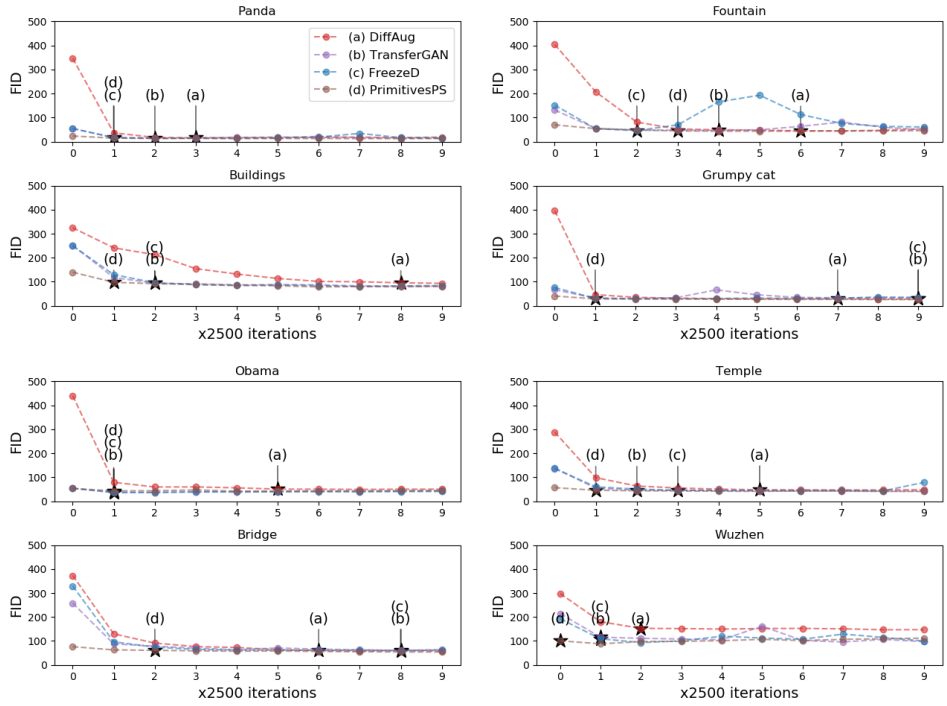

Figure 3: The additional results of Figure 6 in the main text. FID per training iterations. The star marker (★) indicates the point where the model reaches 95% of the best FID score of the from scratch model with DiffAug (baseline). The legend is the same for all graphs.

3 Convergence Speed of Transfer Learning Methods

Figure 3 shows the evolution of the FID scores during the training of the transfer learning methods. The model pretrained with our synthetic dataset exhibits comparable or faster convergence than the competitors that are pretrained on FFHQ. Herein, we observe the convergence speed in terms of the number of iterations to reach 95% of the best FID score of the baseline (from scratch model with DiffAug).

(a) `PinkNoise`

(b) `Primitives`

Figure 4: Low-shot image generation results of the models transferred from `PinkNoise` and `Primitives`.

4 Qualitative Comparison Among Data Synthesizers

In addition to the quantitative comparison of our data synthesizers, we also qualitatively compare our four variants of the data synthesizer used for quantitative evaluation. From the first to the last row, Bridge of sighs, Obama, Grumpy cat, and Panda. `PinkNoise` generates the images with unstructured samples (e.g. Obama and Grumpy cat) and the outputs of `Primitives` on Panda have lower fidelity (e.g. the last three samples). Compared to `PinkNoise` and `Primitives`, `Primitives-S` and `Primitives-PS` provide plausible samples. Between the last two synthetic datasets, `Primitives-S` sometimes drops the important factor, for example, the eyes of the cat (6-th column). While `Primitives-PS` generates more diverse and plausible samples than the other synthetic datasets.

(a) `Primitives-S`

(b) `Primitives-PS`

Figure 5: Low-shot image generation results of the models transferred from `Primitives-S` and `Primitives-PS`.

5 Qualitative Comparisons With Competing Transfer Learning Methods

In addition to the quantitative comparison, we also provide the qualitative comparisons on eight datasets that are used for quantitative evaluation in the main text. From the first to the last row, Buildings, Bridge of sighs, Obama, Medici fountain, Grumpy cat, Temple of heaven, Panda, and Wuzhen. In terms of fidelity of the generated images, our `Primitives-PS` outperforms the competitors. Especially, Grumpy cat images generated by the competitors often do not contain eyes or have only part of the face. Because of the size, we show one figure per page.

Figure 6: The additional generated samples of Figure 5 in the main text. The images are generated with the model trained from scratch.

Figure 7: The additional generated samples of Figure 5 in the main text. The images are generated with the model pretrained with FFHQ and transferred by using TransferGAN.

Figure 8: The additional generated samples of Figure 5 in the main text. The images are generated with the model pretrained with FFHQ and transferred by using FreezeD.

Figure 9: The additional generated samples of Figure 5 in the main text. The images are generated with the model pretrained with our `Primitives-PS`.

References

[1] Tero Karras, Miika Aittala, Janne Hellsten, Samuli Laine, Jaakko Lehtinen, and Timo Aila. Training generative adversarial networks with limited data. In H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 12104–12114. Curran Associates, Inc., 2020.

[2] Alejandro Newell and Jia Deng. How useful is self-supervised pretraining for visual tasks? In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7345–7354, 2020.

[3] Yaxing Wang, Chenshen Wu, Luis Herranz, Joost van de Weijer, Abel Gonzalez-Garcia, and Bogdan Raducanu. Transferring gans: generating images from limited data. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 218–234, 2018.

[4] Jason Yosinski, Jeff Clune, Yoshua Bengio, and Hod Lipson. How transferable are features in deep neural networks? *Advances in Neural Information Processing Systems*, 27:3320–3328, 2014.