APPENDIX
TOWARDS FASTER AND STABILIZED GAN TRAINING FOR HIGH-FIDELITY FEW-SHOT IMAGE SYNTHESIS

Anonymous authors
Paper under double-blind review

A PERFORMANCE BOOST FROM SKIP-LAYER EXCITATION

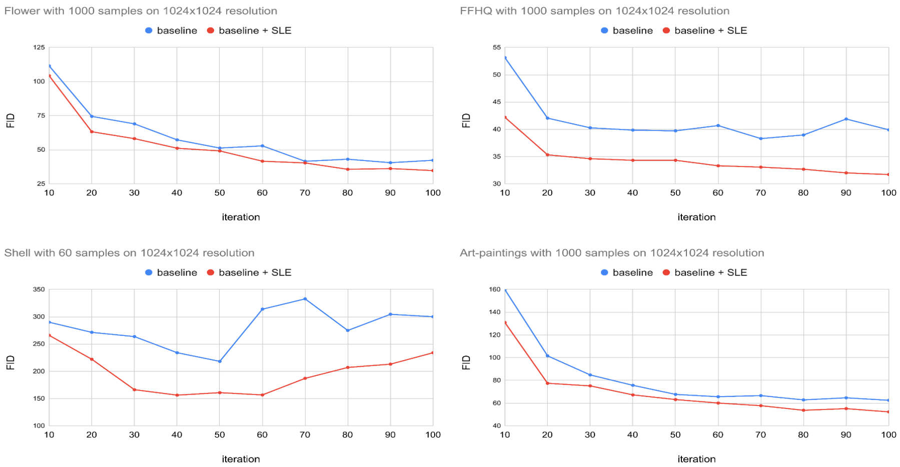


Figure 1: **Ablation study for SLE module** on 1024×1024 resolution datasets. Each unit on the x-axis represents 1000 training iterations, and y-axis represents the FID score.

Here we present a more detailed ablation study for the skip-layer excitation (SLE) module. We compare between the baseline model and the baseline equipped with SLE. On four 1024×1024 resolutions datasets: Flower, FFHQ, Shell and Art-paintings, we record the FID performance every 10000 iterations for every model. As shown in Fig. 1, SLE brings a constant performance boost on the baseline model over all iterations.

Our key observation is, SLE speeds up the convergence of GAN, where the most noticeable effect happens at the beginning of the training. In the first 20000 iterations, the generator G is able to converge faster and reach to a good point where the baseline model needs much more training iterations to reach. On the other hand, although SLE provides a faster convergence on G, the overall model behavior with SLE seems follow the baseline model quite well, with a slightly better overall performance.

In other words, the lines for the two models are parallel in each sub-plot in Fig. 1. Specifically, on Shell, the model with SLE also collapsed after 60000 iterations training, just like the baseline model. And on the rest three datasets, the FID improves much slower and almost stop changing in the later half training iterations. We think such model behavior makes sense, because the SLE module neither increases the model capacity (have very few parameter increase) nor exert any explicit regularization or guidance on the training of GAN. Therefore, SLE is unlikely to make a big difference after the model reaches a good converged state.

On the other hand, SLE does a good job speeding up the convergence for G, and improves the performance of G. More importantly, it is SLE that enables the unsupervised style-content disentanglement for our model, in a simpler and more cost-efficient way than StyleGAN and StyleGAN2.

B Feature-extraction performance of Discriminator

Table 1: LPIPS on D's feature-extracting performance

	Grumpy Cat	Obama	FFHQ			Art	
Image number	100	100	1k	70k	0	1k	0
StyleGAN2	0.914	0.652	3.177	2.43	2.289	3.051	2.761
Baseline	1.632	0.733	2.421	N/A	1.943	2.677	2.421
Baseline + Contrastive	1.251	0.647	1.821	N/A	1.943	2.124	2.421
Baseline + AE	**0.725**	**0.405**	**1.075**	N/A	1.943	**1.806**	2.421
Baseline + AE + Contrastive	1.156	0.578	1.345	N/A	1.943	1.927	2.421

Here we continue the discussion on the effectiveness of the self-supervised auto-encoding training for the discriminator D. Specifically, we explore the relationship between the *feature-extracting behavior on the discriminator* D and the *synthesis performance of GAN* . By feature-extracting performance, we mean how comprehensive the feature-maps extracted by D cover the information from the input images. This feature-extracting performance can be easily checked via an auto-encoding training. In detail, we take D trained in GAN and fix it, then train a decoder for D which tries to reconstruct the images from the feature-maps encoded by D. The intuition is, if D pays attention to all the regions of an input image, and encode the image with a minimum information lost, then the decoder is easier to reconstruct the images encoded by D. In contrast, if D is overfitting and only focus on limited local patterns of the images, the it outputs feature-maps with lost information, thus a decoder is unable to reconstruct the images from D's output feature-map.

We extract the second-last layer's activation for the decoder, which is the one for D to determine the real/fake of an image. Table 1 shows the result, where we train all the decoders for the same 100000 iterations (all the decoders are converged). Note that such feature-extracting performance on D does not necessarily imply a better synthesis performance for G. Moreover, the D from StyleGAN2 is not comparable to the D from baseline, since they have totally different model structure and complexity.

Instead, according to Table 1, we can get some interesting information. Firstly, the GAN training is actually making D performs worse as a feature-encoder. According to row. 3 (StyleGAN2) and row. 4 (baseline), we find that the D after a GAN training extracts less meaningful features compared to a randomly initialized D (col. 6 and col. 8). It means that while the GAN training leads D to find the discriminative features between the real and fake samples, it also effective let D to ignore quite amount of information from the input images.

Secondly, we compare the baseline model to the ones with self-supervised learning guidance (row. 4,5,6,7). It shows that the self-supervisions on D indeed lead to a more descriptive feature-extraction compared to the randomly initialization on D. Moreover, contrastive learning may also result in overfitting, since only a partial image (some local patterns) may be enough for the classification task. In comparison, the reconstruction task is more likely to let D cover more information from the input images. To our surprise, combining auto-encoding training and the contrastive learning result in a worse performance on D. It shows that the classification objective affects the auto-encoding objective and changes the behavior of D, in a negative way.

Last but not least, we do find that a better *feature-extracting performance on D* result in a better *synthesis performance of GAN* . And it seems true for both StyleGAN2 and the baseline model. For StyleGAN2 trained on FFHQ, D trained with more data indeed preserves more information from the input images than D trained on only 1000 images. For our baseline model, the feature-extracting performance on D aligns well with the respective FID scores. Besides, the self-supervision methods all effectively letting D extracts more information from the images, compared to the randomly initialization and the vanilla GAN training.

Apart from the observations, we would like to emphasize that the experiments are mostly conducted on few-shot datasets. The results does not give a full picture of the relationship between the feature-extraction performance on D and the synthesis performance of GAN, further study on larger-scale datasets are required. However, the experiments do validate the effectiveness of the self-supervision strategies on D for an enhanced performance of GAN, on few-shot datasets.

C Style-mixing on different resolutions

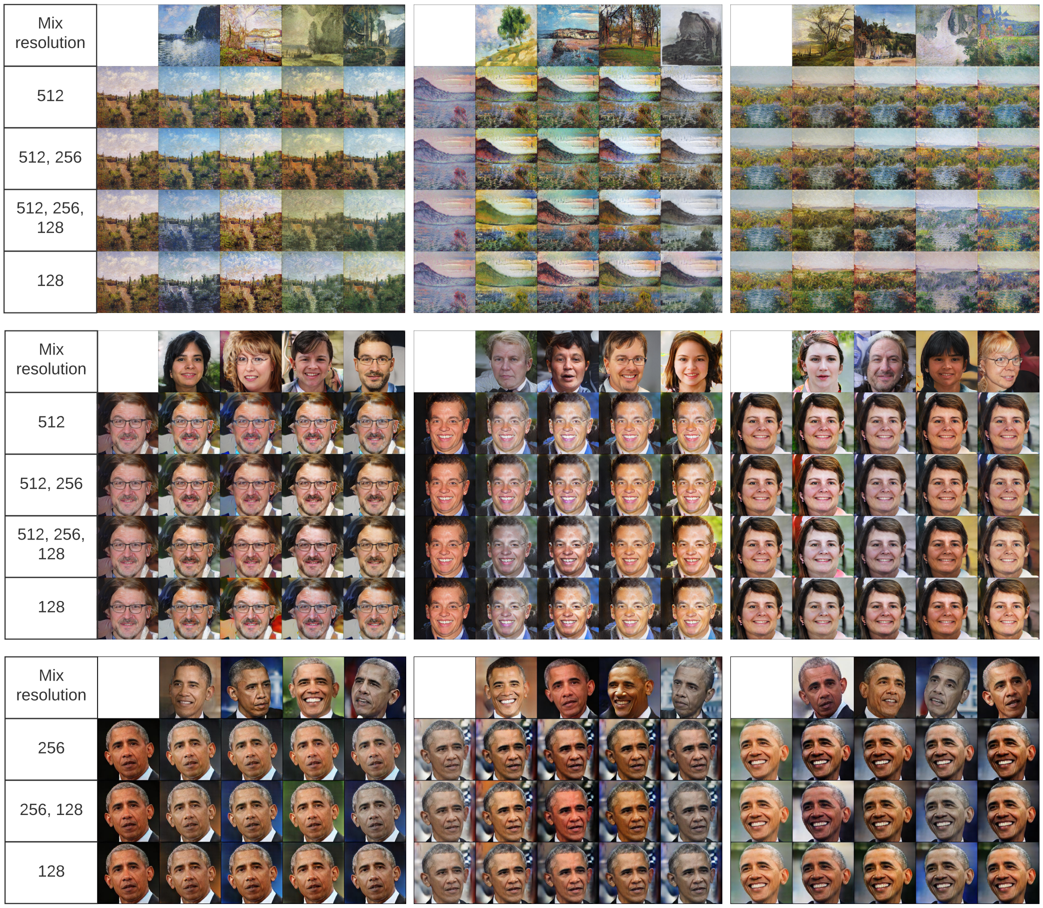


Figure 2: **Style-mixing results** by swapping the features for SLE on different resolutions.

Here we present more qualitative results on the style-mix performance of our model. For the model trained on 1024×1024 resolution, there are three SLE layers that we can swap the feature-maps between generated samples, and there are two SLE layers for model on 256×256 resolution.

Fig. 4 shows the results from the 1000 samples training on Art paintings and FFHQ at 1024×1024 resolution, and the 100 samples Obama at 256×256 resolution. In each row, we swap the $\mathbf{x}_{low}$ in the SLE layer from the image in col. 1 to the one from each image on row. 1. The best style-mixing results is achieved when the feature-map swapping is done on all resolutions. And the most effective layer that causes the most style changes is the layer on 128 resolution. On 256×256 resolution, the model behaviors the same, where the SLE on lower resolution makes the most style difference.

On the Art-paintings data, the model performs well on style-mixing, where not only the coloring but also the texture can be controlled. The models transfers the style of flat or pointy brush stroke among the style-mixed synthetic images. However, the model does not perform as well on the FFHQ data. There are some cases where even the hair color can not be properly transferred. We speculate that the worse performance on FFHQ is due to the limited training sample and the dramatically varied background. The is no clear relationship between the front-end face and the background contents given the limited training samples, which confuses the model to disentangle more detailed style attributes. In contrast, Art-paintings have consistent style cues within each image and obvious connections between each object inside a scene, making it arguably easier than the FFHQ data. On the other hand, the model performs great on Obama given a even less 100 training images. It successfully transfers the style for both the face attributes and the background. Learning on 256^2 resolution is a simpler task, and the model capacity is more sufficient on only 100 samples.

D More Qualitative Comparison

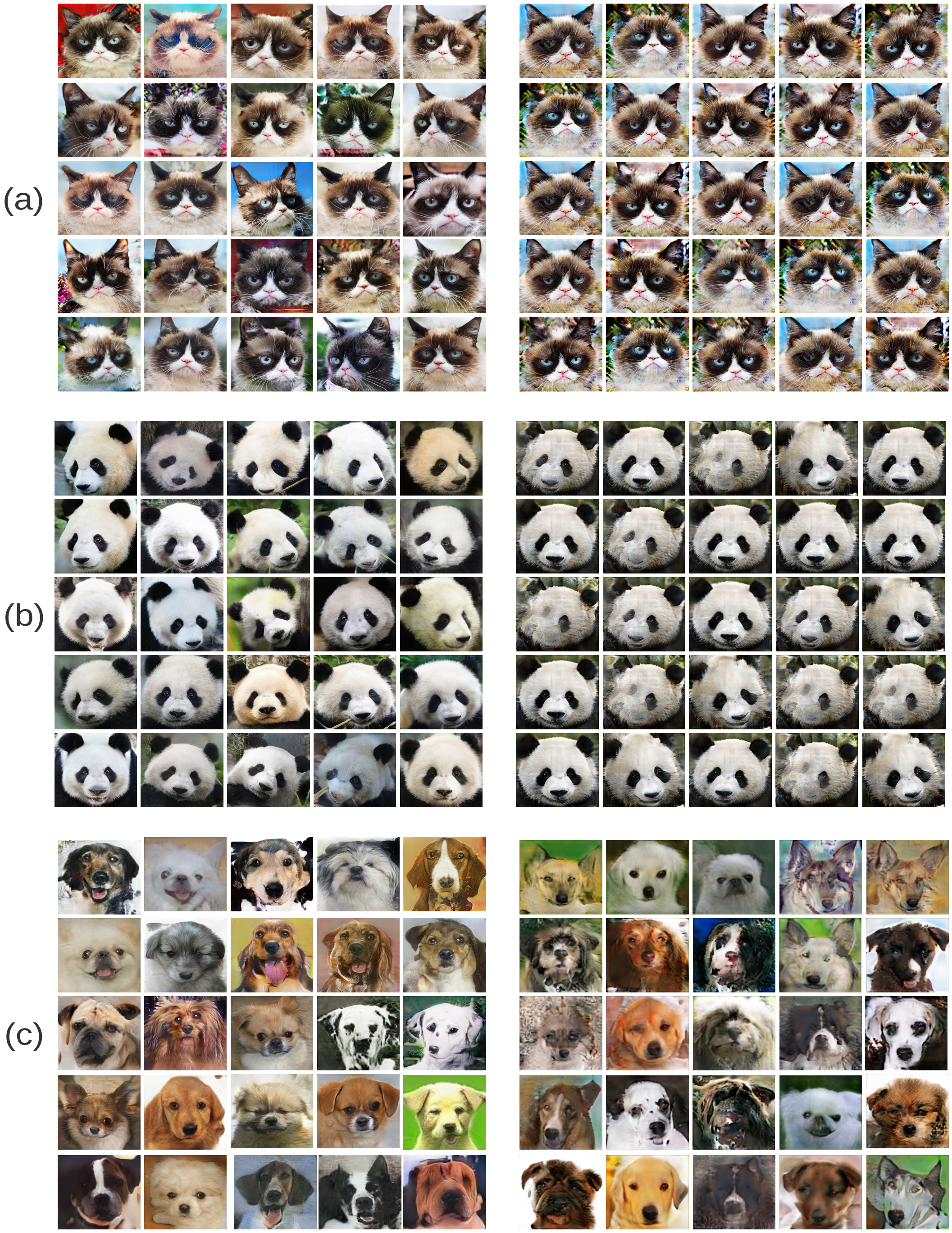


Figure 3: **Comparison between our model and the baseline** For each dataset, the images are generated by the same set of randomly sampled noises. Images from our model is shown on the left, and the baseline results are on the right. All the model are trained for 50000 iterations with batch size of 8, which is more than enough for both models to converge. On (a) Grumpy-cat and (b) Panda, baseline model shows a clear mode collapse, while our model is generating diverse images; on (c) Animalface-dog, although not mode collapse, the baseline model shows a clear quality disadvantage compared to our model.

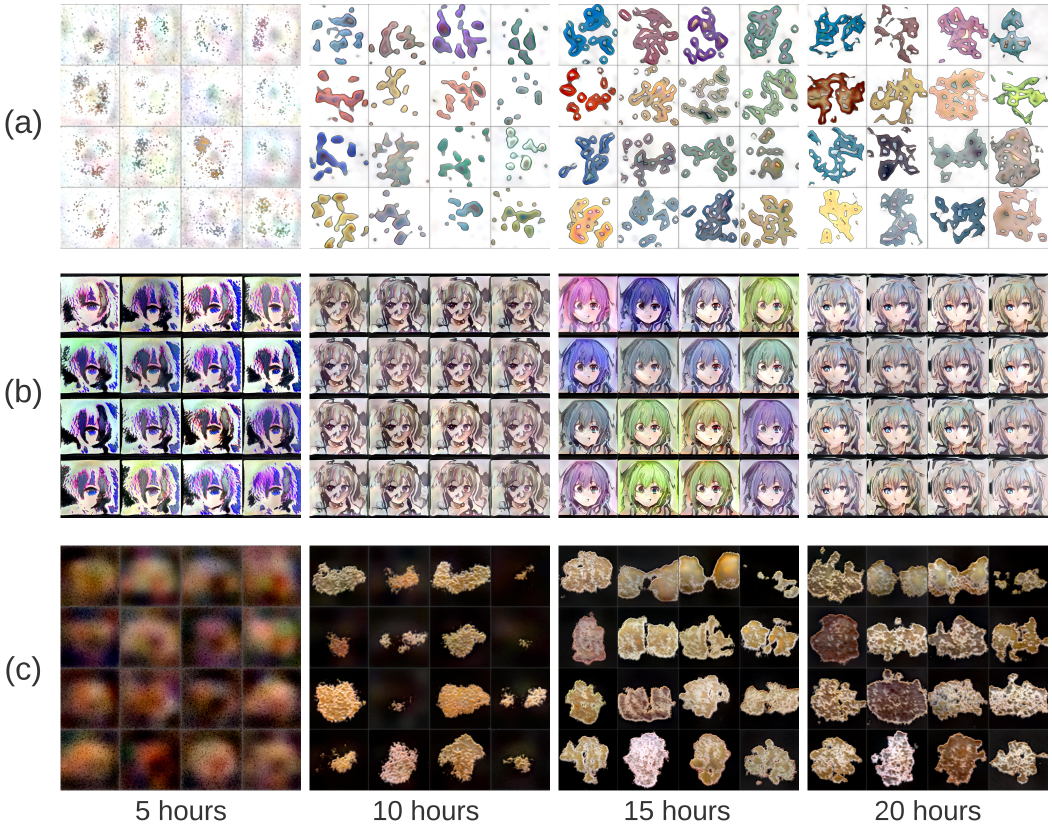
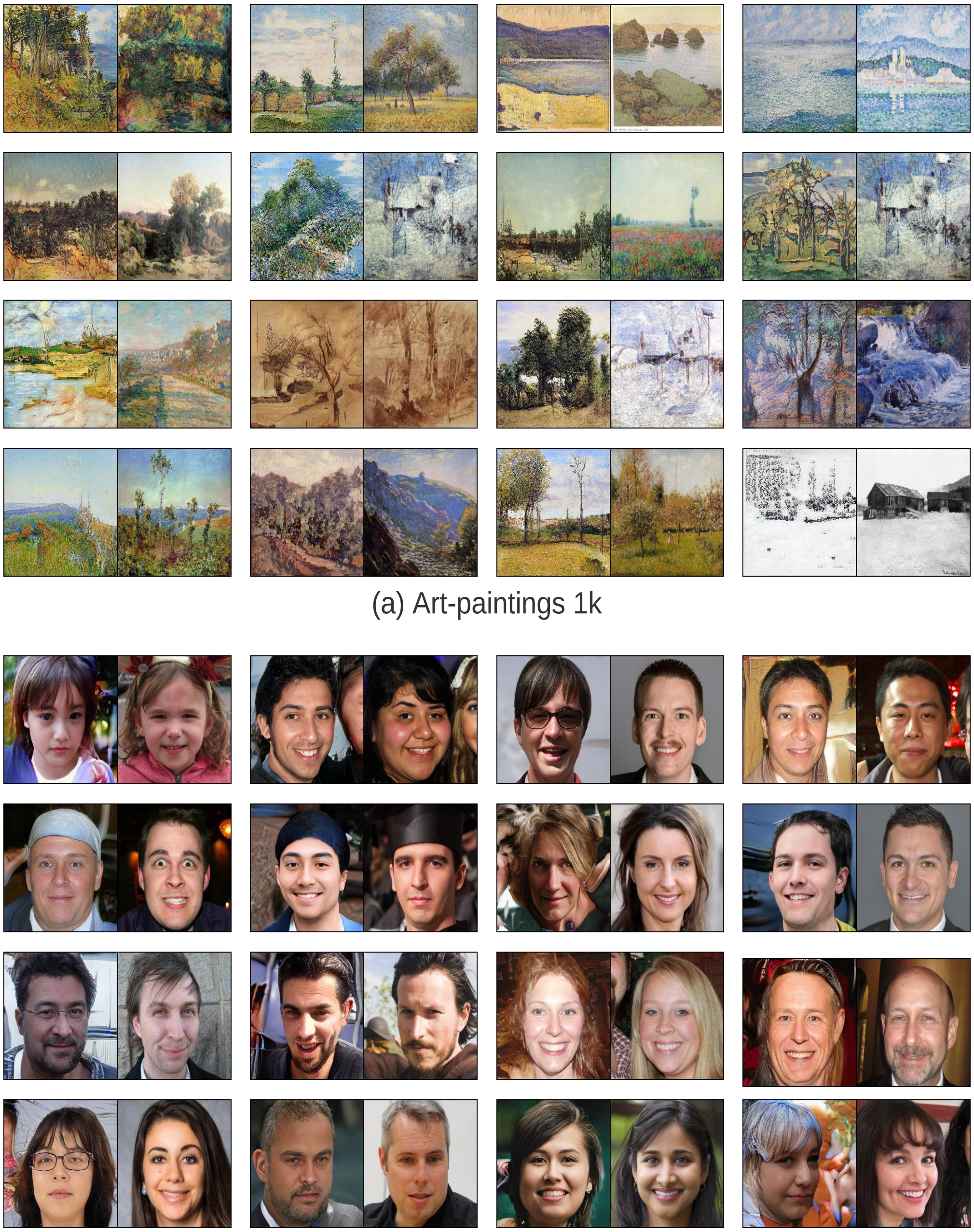


Figure 4: **StyleGAN2 results during training** We show the results of the slimed StyleGAN2 at half channel numbers. StyleGAN2 converges much slower than our model on dataset (a) Pokemon and (c) Shell, and mode collapsed on (b) Anime-Face.

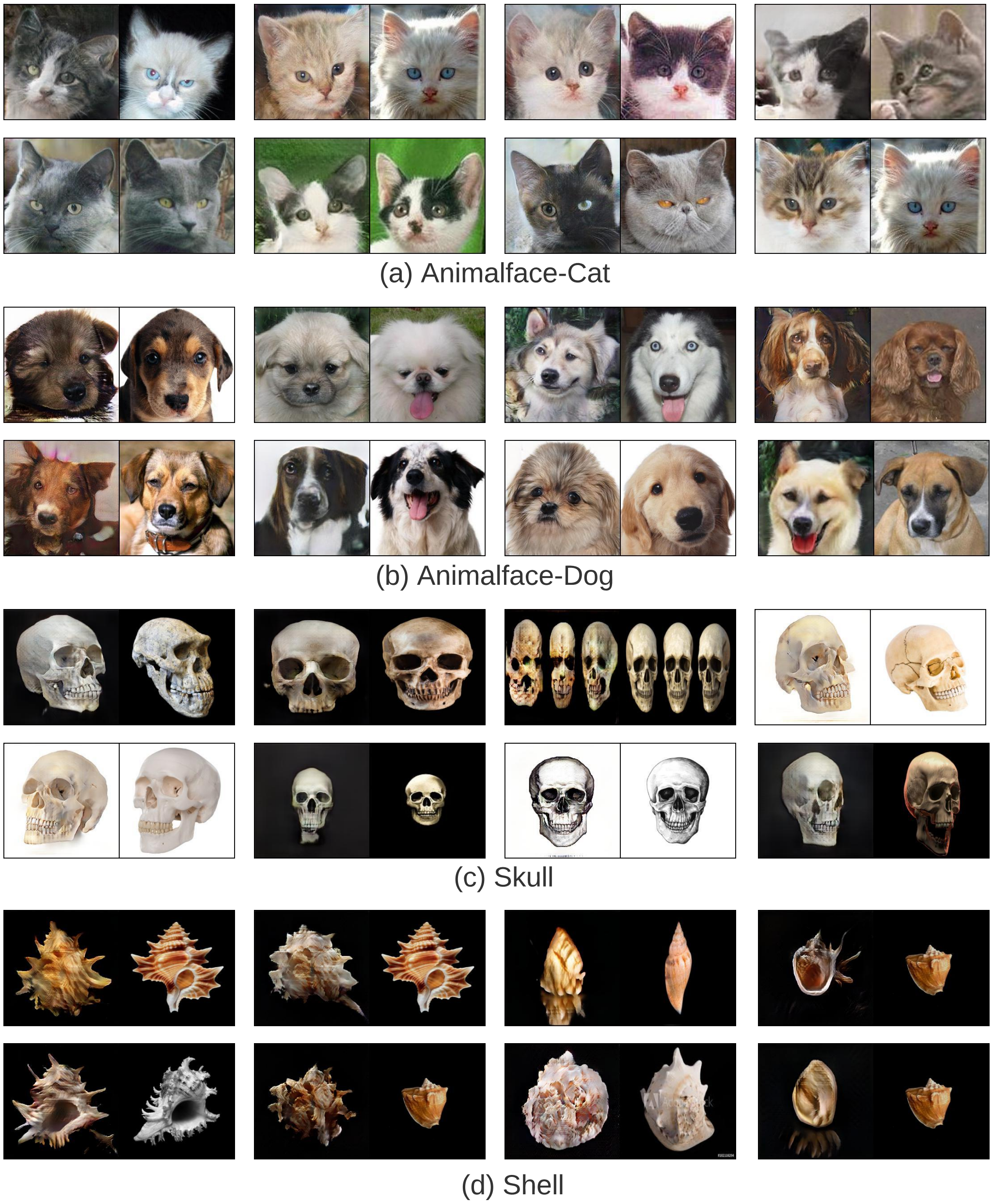
E Nearest images from training sets



(a) Art-paintings 1k

(b) FFHQ 1k

Figure 5: **Nearest real images to the synthesized ones trained on 1000 images** For each pair of images, the left is the synthesized image from our model, and the right image is the closest image found from the real training data ranked by LPIPS score. The samples are uncurated, and our model is able to create new contents that well fitted to the training domain.



(a) Animalface-Cat

(b) Animalface-Dog

(c) Skull

(d) Shell

Figure 6: **Nearest real images to the synthesized ones trained on 100 images** For each image pair, the left is the synthesized image from our model, and the right is the closest image found from the real training data ranked by LPIPS score. Even with only 100 training samples, these uncurated samples show our model is still able to combine the features learned from the real samples and synthesize new compositions.

Table 2: LPIPS between synthetic images and their closest real images.

	Art paintings 1k	FFHQ 1k	Skull	Cat	Dog	Shell
augmented	0.5499	0.5279	0.389	0.3898	0.3847	0.3853
Our G	0.637	0.5859	0.3168	0.5486	0.5647	0.4275

In Table 2, we report the average LPIPS score between the generated samples from our model to their closest real samples ranked by LPIPS score. In comparison, we show the baseline as the LPIPS between real images and their randomly augmented variants (randomly horizontal flipping and random cropping with 0.8 spatial portion). We run each experiment 3 times with 100 randomly synthesized samples or real images, and report the lowest one. The std among the trials are usually lower than 0.005. This experiment shows that, instead of memorizing the real images in the training set, our model is able to perceive the features from the real images, and generate images that are different and novel, in terms of compositions, shapes, and color patterns.

F Decoder result

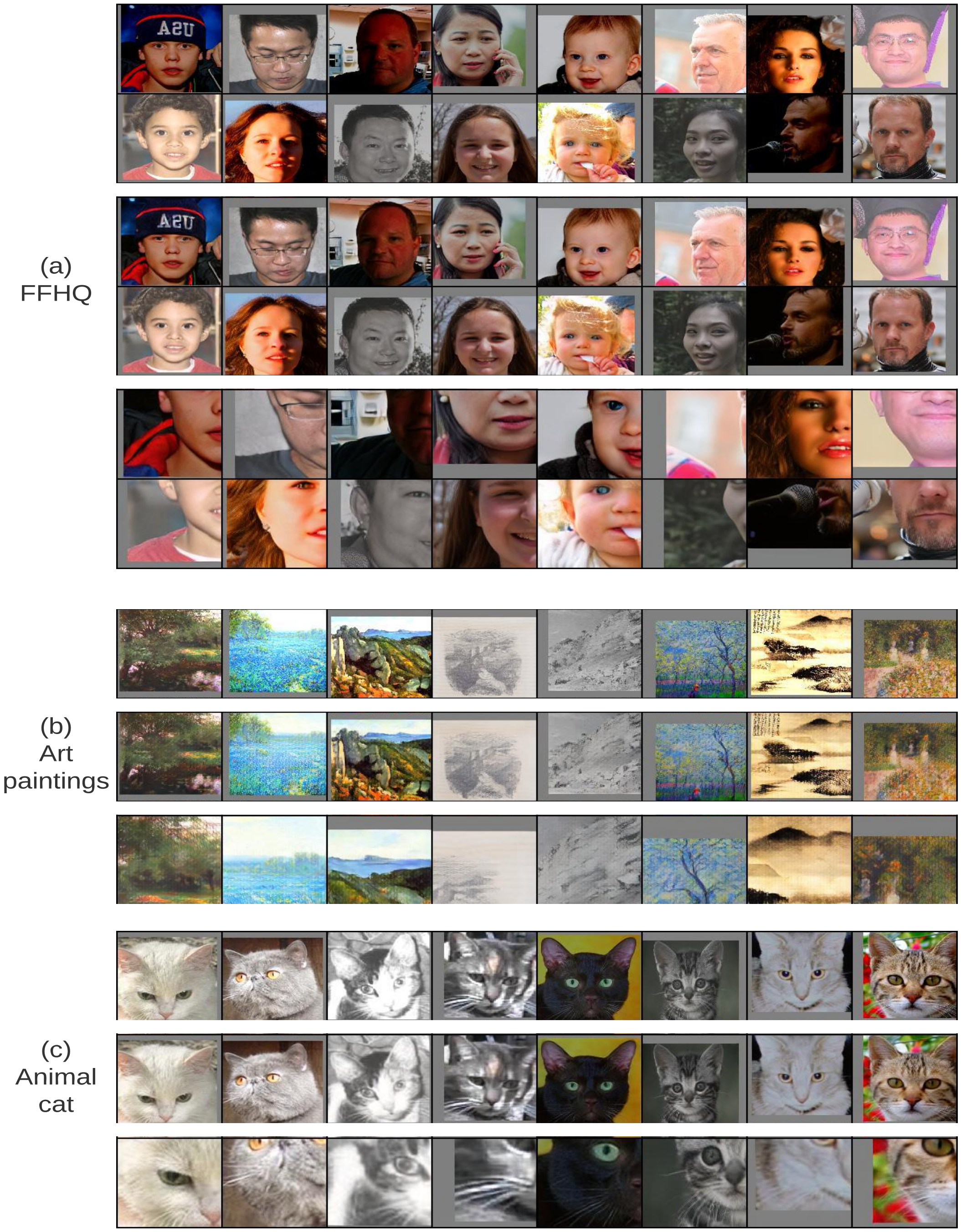


Figure 7: **Reconstruction results from the decoder for training the auto-encoding discriminator**. For each dataset, the first panel shows the augmented real images during training, the second panel shows the reconstruction on full image, and the last panel shows the reconstruction on random cropped portions of the full image. Add the reconstructions are done on 128 × 128 resolution.