Name: Sean Goh Ann Ray
Matric No.: G2202190G

Preprocessing of Data:

The FFHQ data provided consists of 5000 high-quality (HQ) face images of size 512 x 512 pixels for training, as well as 400 high-quality and low-quality (LQ) face image pairs for validation, where the LQ images are corrupted, degraded HQ images resized to 128 x 128 pixels. Since only HQ images are provided for the training, preprocessing is required to make the LQ counterparts to train the super-resolution (SR) model.

This is done by corrupting and downsizing the HQ training images to make LQ images with a random degradation pipeline, including gaussian blur, downsampling, adding gaussian noise, JPEG compression, and then resizing to 128 x 128 pixels. With the HQ-LQ image pairs, the SR model will then learn and generate realistic HQ images from the LQ images by minimizing the loss function. No preprocessing was performed on the validation set as they were provided as HQ-LQ pairs.

2 test sets were provided at a later date, the first being 400 LQ images from the FFHQ dataset and the second being 6 LQ real-world images. As the model required a ground-truth (GT) image to run, the fake GT counterparts were made by upscaling the LQ images 4 times (from 128 x 128 to 512 x 512) using a bicubic interpolation. These fake GT images are used to calculate the peak signal-to-noise ratio (PSNR) metric during testing, but the real PSNR test score is only available upon submission to the CodaLab competition.

GPU:

NTU's SCSE GPU cluster was utilized as the platform for the algorithm training and testing, with the specifications of the 2 nodes utilized as shown in Figure 1 below.

Table I. SCSE GPU Specifications

	SCSEGPU_M1	SCSEGPU_M2
No. of CPUs	7	10
No. of GPUs	2	1
Memory	12GB	30GB

Model and Number of Model Parameters:

Using the MMEditing library, the provided srresnest_ffhq_300k.py configuration was used as a baseline. It utilized MSRResNet with 16 residual blocks. Several configuration hyper parameter settings were explored and conducted. The hyperparameter settings of the model with the best validation PSNR score of 29.7899 are shown in Table II, alongside the original parameter settings. The number of parameters is obtained using the tutorial from MMEditing webpage, where the following command was used:

Python tools/analysis_tools/get_flops.py srresnet_ffhq_500k_v8.py –shape 3 128 128

Table II. Model Hyperparameter Setting

Model	Baseline	Best Validation PSNR Score
No. of Residual Blocks	16	24
No. of Parameters (Million)	1.518	2.108
Loss Function	L1	PSNR
No. of Iterations	300k	500k
Cosine Annealing Periods	150k, 150k	100k, 100k, 100k, 100k, 100k

Loss Functions:

The Adam optimizer with the following parameters was used:

Learning Rate = 0.0002
Betas = 0.9, 0.999
Learning Rate Scheduler = Cosine Annealing with 5 periods of 100k iterations

The loss function used was PSNR loss between the pixels, with the best model weights being saved as a checkpoint based on the highest PSNR score, which goes by the following equation:

$$PSNR = 20 * log \frac{255}{\sqrt{MSE}}$$

where MSE is the mean squared error of the pixel values between the output of the model and the HQ validation images.

Training Curves:

The PSNR score on the validation set was recorded during the training of all model iterations, and 3 were chosen to be plotted and is shown in Figure 1. The best performing model was loaded with the checkpoint that achieved the best validation PSNR score of 29.7899 and was used to evaluate on the test dataset which was released at a later date. The generated super-resolution images of the LQ test images are then submitted to CodaLab, where the PSNR score on the test set was 29.49933.

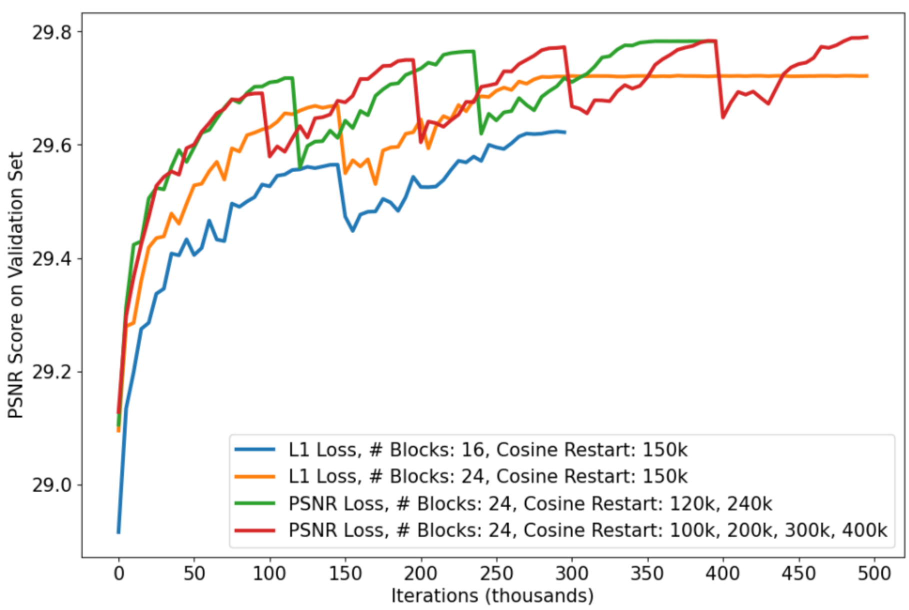


Figure 1. PSNR Plot of Model Training

Observations

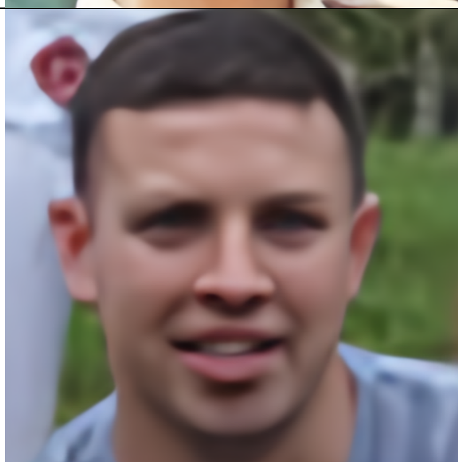
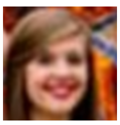
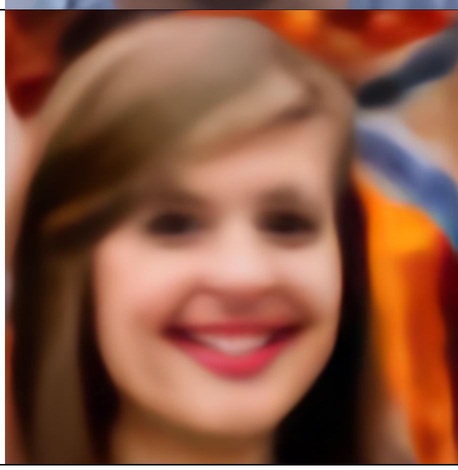
As seen from the blue and orange plots in Figure 1, increasing the number of residual blocks in MSRResNet improves the PSNR score, but also increases training duration. From the orange and green plots, changing the loss function from L1 to PSNR improves the PSNR score on the validation set. From the green and red plots, having more cosine restarts and iterations allowed the model to achieve slightly higher PSNR scores, likely due to 'escaping' from a local minimum, but the periods of each cosine annealing should not be too small as it hindered the improvement, thus a period of 100k was selected for the final model. Increasing the number of iterations is only effective when increasing the number of cosine restarts, but due to limitation in computational time and several variations of model hyperparameter settings explored, the number of iterations was limited to 500k.

Predicted HQ Images on LQ Real-World Images:

6 LQ real world images, with different degrees of blurring, were also provided. These were evaluated on by using the same model and checkpoints used on the test dataset above in order to generate the HQ images. The results of the image super resolution are shown in Table III. The generated images on the slightly and moderately blurry images are decent, but those of the extremely blurry images are severely unrealistic.

Table III. Super-Resolution on Real World Images

Image	LQ Image	Generated HQ Image	Analysis
001			Decent super-resolution image
002			Decent super-resolution image, great details in the eyes but not so much in the hair

003			Decent super-resolution image, but artifacts observed at bottom-left corner of image
004			Decent super-resolution image
005			Poor super-resolution image, facial features recognizable but still blur
006			Very poor super-resolution image, facial features almost unrecognizable, distortion observed at top right areas of face