

2023-S2 AI6126 Project 2

Blind Face Super-Resolution

Project 2 Specification (Version 1.0. Last update on 22 March 2024)

Important Dates

Issued: 22 March 2024

Release of test set: 19 April 2023 12:00 AM SGT

Due: 26 April 2023 11:59 PM SGT



Group Policy

This is an individual project

Late Submission Policy

Late submissions will be penalized (each day at 5% up to 3 days)

Challenge Description

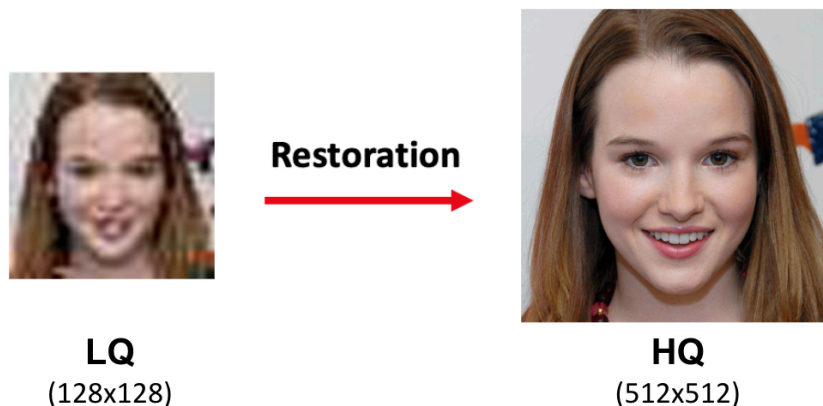


Figure 1. Illustration of blind face restoration

The goal of this mini-challenge is to generate high-quality (HQ) face images from the corrupted low-quality (LQ) ones (see Figure 1) [1]. The data for this task comes from the FFHQ. For this challenge, we provide a mini dataset, which consists of 5000 HQ images for training and 400 LQ-HQ image pairs for validation. Note that we do not provide the LQ images in the training set. During the training, you need to generate

the corresponding LQ images on the fly by corrupting HQ images using the random second-order degradation pipeline [1] (see Figure 2). This pipeline contains 4 types of degradations: Gaussian blur, Downsampling, Noise, and Compression. We will give the code of each degradation function as well as an example of the degradation config for your reference.

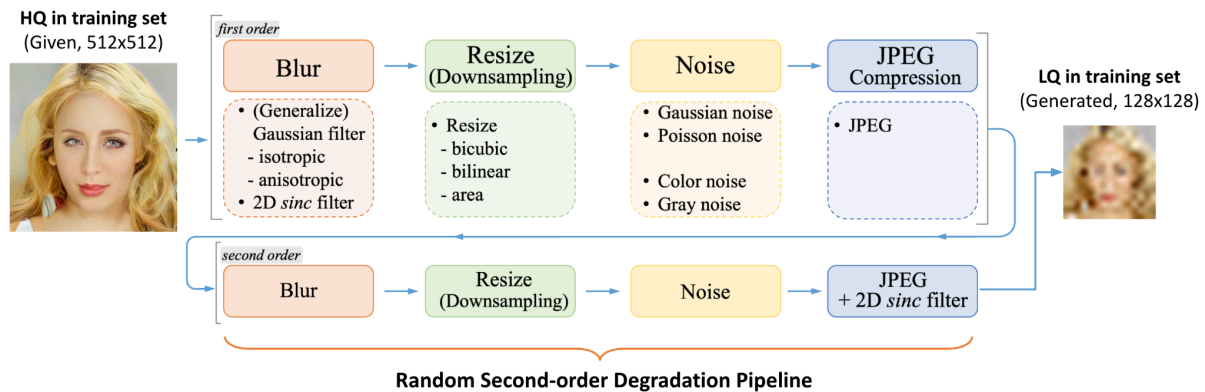


Figure 2. Illustration of second-order degradation pipeline during training

During validation and testing, algorithms will generate an HQ image for each LQ face image. The quality of the output will be evaluated based on the PSNR metric between the output and HQ images (HQ images of the test set will not be released).

Assessment Criteria

In this challenge, we will evaluate your results quantitatively for scoring.

Quantitative evaluation:

We will evaluate and rank the performance of your network model on our given **400 synthetic** testing LQ face images based on the PSNR.

The higher the rank of your solution, the higher the score you will receive. In general, scores will be awarded based on the Table below.

Percentile in ranking	$\leq 5\%$	$\leq 15\%$	$\leq 30\%$	$\leq 50\%$	$\leq 75\%$	$\leq 100\%$	*
Scores	20	18	16	14	12	10	0

Notes:

- We will award bonus marks (up to 2 marks) if the solution is interesting or novel.
- To obtain more natural HQ face images, we also encourage students to attempt to use a discriminator loss with a GAN during the training. Note that

discriminator loss will lower the PSNR score but make the results look more natural. Thus, you need to carefully adjust the GAN weight to find a tradeoff between PSNR and perceptual quality. You may earn bonus marks (up to 2 marks) if you achieve outstanding results on the 6 **real-world** LQ images, consisting of two slightly blurry, two moderately blurry, and two extremely blurry test images. (The real-world test images will be released with the 400 test set) [optional]

- Marks will be deducted if the submitted files are not complete, e.g., important parts of your core codes are missing or you do not submit a short report.
- TAs will answer questions about project specifications or ambiguities. For questions related to code installation, implementation, and program bugs, TAs will only provide simple hints and pointers for you.

Requirements

- Download the dataset, baseline configuration file, and evaluation script: [here](#)
- Train your network using our provided training set.
- Tune the hyper-parameters using our provided validation set.
- Your model should contain fewer than 2,276,356 trainable parameters, which is 150% of the trainable parameters in SRResNet [4] (your baseline network). You can use
 - `sum(p.numel() for p in model.parameters())`to compute the number of parameters in your network. The number of parameters is only applicable to the generator if you use a GAN.
- The test set will be available one week before the deadline (this is a common practice of major computer vision challenges).
- **No external data and pre-trained models are allowed in this mini challenge.** You are only allowed to train your models from scratch using the 5000 image pairs in our given training set.

Submission Guidelines

Submitting Results on CodaLab

We will host the challenge on CodaLab. You need to submit your results to CodaLab. Please follow the following guidelines to ensure your results are successfully recorded.

- The CodaLab competition link:
https://codalab.lisn.upsaclay.fr/competitions/18233?secret_key=6b842a59-9e76-47b1-8f56-283c5cb4c82b
- Register a CodaLab account with your NTU email.
- **[Important]** After your registration, please fill in the username in the Google Form: <https://forms.gle/ut764if5zoaT753H7>
- Submit output face images from your model on the 400 test images as a zip file. Put the results in a subfolder and use the same file name as the original test images. (e.g., if the input image is named as 00001.png, your result should also be named as 00001.png)
- You can submit your results multiple times but no more than 10 times per day. You should report your best score (based on the test set) in the final report.
- Please refer to Appendix A for the hands-on instructions for the submission procedures on CodaLab if needed.

Submitting Report on NTULearn

Submit the following files (all in a single **zip** file named with your matric number, e.g., **A12345678B.zip**) to NTULearn before the deadline:

- A short report in pdf format of not more than five A4 pages (single-column, single-line spacing, Arial 12 font, the page limit excludes the cover page and references) to describe your final solution. The report must include the following information:
 - the model you use
 - the loss functions
 - training curves (i.e., loss)
 - predicted HQ images on 6 real-world LQ images (if you attempted the adversarial loss during training)
 - PSNR of your model on the validation set
 - the number of parameters of your model
 - Specs of your training machine, e.g., number of GPUs, GPU modelYou may also include other information, e.g., any data processing or operations that you have used to obtain your results in the report.
- The best results (i.e., the predicted HQ images) from your model on the 400 test images. And the screenshot on Codalab of the score achieved.

- All necessary codes, training log files, and model checkpoint (weights) of your submitted model. We will use the results to check plagiarism.
- A **Readme.txt** containing the following info:
 - Your matriculation number and your CodaLab username.
 - Description of the files you have submitted.
 - References to the third-party libraries you are using in your solution (leave blank if you are not using any of them).
 - Any details you want the person who tests your solution to know when they test your solution, e.g., which **script to run**, so that we can check your results, if necessary.

Tips

1. For this project, you can use the [Real-ESRGAN](#) [1] codebase, which is based on [BasicSR](#) toolbox that implements many popular image restoration methods with modular design and provides detailed documentation.
2. We included a sample [Real-ESRGAN](#) configuration file (a simple network, i.e., SRResNet [4]) as an example in the [shared folder](#). **[Important]** You need to:
 - a. Put “**train_SRResNet_x4_FFHQ_300k.yml**” under the “**options**” folder.
 - b. Put “**ffhqsub_dataset.py**” under the “**realesrgan/data**” folder.
 The PSNR of this baseline on the validation set is around **26.33 dB**.
3. For the calculation of PSNR, you can refer to ‘evaluate.py’ in the [shared folder](#). You should replace the corresponding path ‘xxx’ with your own path.
4. The training data is important in this task. If you do not plan to use MMEEditing for this project, please make sure your pipeline to generate the LQ data is identical to the one in the configuration file.
5. The training configuration of GAN models is also available in [Real-ESRGAN](#) and [BasicSR](#). You can freely explore the repository.
6. The following techniques may help you to boost the performance:
 - a. Data augmentation, e.g. random horizontal flip (but do not use vertical flip, otherwise, it will break the alignment of the face images)
 - b. More powerful models and backbones (within the complexity constraint), please refer to some works in reference.
 - c. Hyper-parameters fine-tuning, e.g., choice of the optimizer, learning rate, number of iterations
 - d. Discriminative GAN loss will help generate more natural results (but it lowers PSNR, please find a trade-off by adjusting loss weights).
 - e. Think about what is unique to this dataset and propose novel modules.

References

- [1] Wang et al., Real-ESRGAN: Training Real-World Blind Super-Resolution with Pure Synthetic Data, ICCVW 2021
- [2] Wang et al., GFP-GAN: Towards Real-World Blind Face Restoration with Generative Facial Prior, CVPR 2021
- [3] Zhou et al., Towards Robust Blind Face Restoration with Codebook Lookup Transformer, NeurIPS 2022
- [4] C. Ledig et al., Photo-realistic Single Image Super-Resolution using a Generative Adversarial Network, CVPR 2017
- [5] Wang et al., A General U-Shaped Transformer for Image Restoration, CVPR 2022
- [6] Zamir et al., Restormer: Efficient Transformer for High-Resolution Image Restoration, CVPR 2022

Appendix A Hands-on Instructions for Submission on CodaLab

After your participation to the competition is approved, you can submit your results here:

Learn the Details Phases **Participate** Results

Get Data

Files

Submit / View Results

Final test

Step 1

Phase description

The online evaluation results must be submitted through this CodaLab competition site for AI6126 Project 1 CelebAMask Face Parsing.

Max submissions per day: 10

Max submissions total: 100

Max Submission Size: 30 megabyte(s)

Click the Submit button to upload a new submission.

Optionally add more information about this submission

Submit

Step 3

Here are your submissions to date (✓ indicates submission on leaderboard):

#	SCORE	FILENAME	SUBMISSION DATE	SIZE (BYTES)	STATUS	✓
No data available in table						

Then upload the zip file containing your results.

If the 'STATUS' turns to 'Finished', it means that you have successfully uploaded your result. Please note that this may take a **few minutes**.