AI6126 Project 2
Blind Face Super-Resolution

Project 2 Specification (Version 1.0. Last update on 16 March 2022)

Important Dates

Issued: 18 March 2022

Release of test set: 15 April 2022 12:00 AM SGT

Due: 22 April 2022 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)

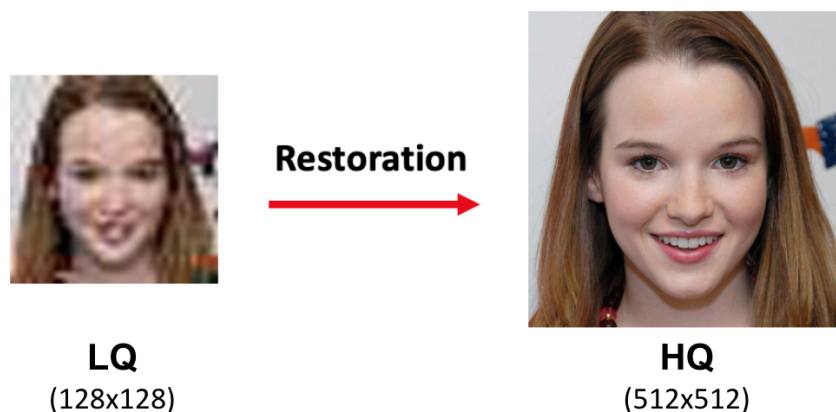## Challenge Description

LQ
(128x128)

HQ
(512x512)

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 4000 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

degradation pipeline (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.
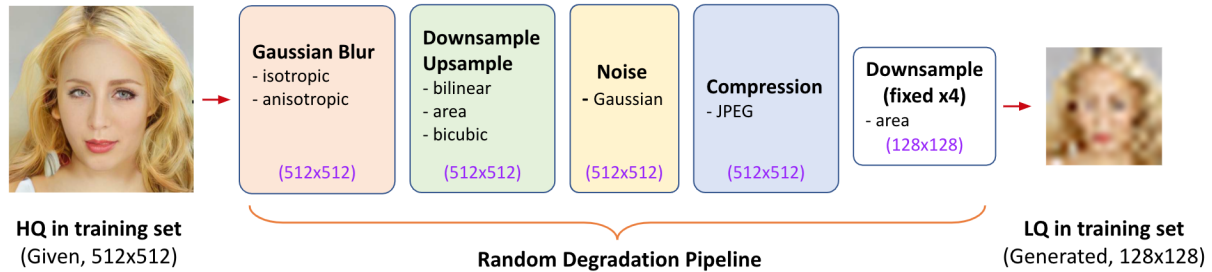

Figure 2. Illustration of degradation pipeline during training

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	≤ 5%	≤ 15%	≤ 30%	≤ 50%	≤ 75%	≤ 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 take an 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. We will award additional bonus marks (up to 2 marks) if your results on the given 5 real-world LQ

images look very good. (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, program bugs, TAs will only provide simple hints and pointers for you.

Requirements

- Download dataset: here
- Train your network using our provided training set.
- Tune the hyper-parameters using our provided validation set.
- To maintain fairness, your model should contain fewer than 1,821,085 trainable parameters, which is 120% of the trainable parameters in SRResNet [2] (your baseline network). You can use
  - `sum(p.numel() for p in model.parameters())`
  to compute the number of parameters in your network.
- 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 4000 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/2848?secret_key= fd9d4a61-a308-4d6a-b78d-28c45cdb3908
- Register a CodaLab account with your NTU email.
- After your registration, please fill in the username in the Google Form: https://forms.gle/t4SdDX86eBVpXA8dA
- 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 5 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 model

    You 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

- For this project, you can use the [MMEditing](#) codebase, which implements many popular image restoration methods with modular design and provides detailed documentation.
- 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.
- We included a sample MMEditing configuration file as an example in the shared folder. The PSNR of this baseline on the validation set is **28.89 dB**.
- The training data is important in this task. If you do not plan to use MMEditing for this project, please make sure your pipeline to generate the LQ data is identical to the one in the configuration file.
- The training configuration of GAN models is also available in MMEditing. You can freely explore the repository.
- The following techniques may help you to boost the performance:
  - Data augmentation, e.g. random horizontal flip (but do not use vertical flip, otherwise, it will break the alignment of the face images)
  - Deeper model (within the complexity constraint)
  - Hyper-parameters fine-tuning, e.g., choice of the optimizer, learning rate, number of iterations
  - Discriminative GAN loss will help generate more natural results (but it lowers PSNR, please find a trade-off by adjusting loss weights).
  - Think about what is unique to this dataset and propose novel modules.

Computational Resource

You can use the computational resources assigned by the MSAI course. Alternatively, you can use Amazon's EC2 or Google CoLab for computation. As a student, you can sign up to receive free $100 credit through the AWS Educate program. We encourage students to use *g2.2xlarge* instances running Ubuntu for maximal ease of installing. Note that $100 of Amazon credit allows you to run a *g2.2xlarge* GPU instance for approximately 6 days without interruption (you should keep it on only while using it).

References

[1] Wang et al., GFP-GAN: Towards Real-World Blind Face Restoration with Generative Facial Prior, CVPR 2021
[2] C. Ledig et al., Photo-realistic Single Image Super-Resolution using a Generative Adversarial Network, CVPR 2017

Appendix A Hands-on Instructions for Submission on CodaLab

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

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.