

MMI712: Machine Learning Systems Design and Deployment: Term Project

December 17, 2022

1 Introduction

In this project, you are going to containerize the [Real-ESRGAN](#) repository to apply super-resolution to images using a containerized environment. You only need to make inference on images using `inference_realesrgan.py`. Video inference is not required.

2 Requirements

You need to use Docker and `nvidia-container-runtime` to containerize the repository. You need to use basic Dockerfile commands such as `FROM`, `RUN`, `COPY`, `WORKDIR`, `ENV` etc. You do not need to install PyTorch or CUDA in your Docker images as you can use the images provided by PyTorch on [Docker Hub](#) as the base image. You need to set up the required libraries and tools stated in the repository in your Docker image. You need to upload your Docker image to Docker Hub and use the handle in your inference script. Your images should be self-contained and should be able to run seamlessly on a machine with Docker and Nvidia drivers installed.

If you do not have access to a computer with an NVIDIA GPU, please [mail](#) me as soon as possible for us to allocate a device for you.

3 Deliverables

- An executable bash script named `infer.sh` for making inference using your container on Docker Hub along with a sample input image in png format for us to test your script. All the code files should be located inside the container at the final submission. (20 pts)
- Dockerfile of your Docker image. (20 pts)
- A project report (max. 3 pages), explaining your Dockerfile and all the challenges you encountered through the project in detail. (40 pts)
- We should be able to run your script with the following command: `./infer.sh -i path_to_input_img`. The script should save the output image named `out.png` at the directory of input image. The naming is important since an automated test script is going to be used for grading the implementation. (20 pts)

Good luck, have fun!