

DeepFake Classification Assignment

Deadline: **05/18/2023**

Inst Email: swoo@g.skku.edu

TA email: kishor25@g.skku.edu

1. Download the High Quality (c23 compression) and Low Quality (c40 compression) images of deepfake from [Here](#). You will see there are two folders named High Quality and Low Quality. Each category contains 2 datasets named Face2Face and NeuralTexture.
2. Try three pre-trained models for each dataset -> 1) [Xception](#), 2) [EfficientNetB0](#) in Pytorch. Alternatively, you can use [timm](#) to load these pretrained models. [Here](#) is a colab example of how to use *timm* to load pretrained model. So there will be 4 experiments in total : HighQuality-Face2Face, HighQuality-NeuralTexture, LowQuality-Face2Face and LowQuality-NeuralTexture

Steps you should follow to train models:-

- a) Make dataloader for train, test and validation set.
 - b) All pre-trained models expect input images normalized in the same way, i.e. mini-batches of 3-channel RGB images of shape (3 x H x W), where H and W are expected to be 224. The images have to be loaded in to a range of [0, 1] and then normalized using mean = [0.485, 0.456, 0.406] and std = [0.229, 0.224, 0.225]. This is a binary classification problem since there is only 2 classes (real and fake).
 - c) Plot the loss curves as well as the accuracies of training and validation sets for 100 epochs.
3. Report the performance (Accuracy, F1 Score, Precision and Recall) for test set.
 4. Write a brief report and submit it as a pdf. Your pdf should contain link of the code (github/colab) , pseudocodes , plots, graphs etc. **Do not upload any zip.**

[Contact the TA if you have any further queries.](#)