**READ-ME FILE**

Coding instructions for Discriminator Model (.ipynb file):

1. Upload all the files in the given folder to the Google drive.
2. Open the uploaded Model (.ipynb file) in the drive through Google Colab.
3. Change the run time setting for the notebook to access GPU computation power. Go to Runtime –> Change run time type –> Hardware accelerator –> GPU –> Save.
4. Mount the Google drive by running first cell of the notebook.
5. Change the path locations of the .csv files in the cells where we load the train and test data (attribute) to their respective paths in the drive.
6. Change the path locations of the directory in the cells where flow from data frame function is used, to the respective path of the data set in the drive.
7. Run each cell one by one under each section in the given order below

(Discriminator-1 --> Loading Train data – > Image Preprocessing (Train and Validation) –> Loading Test Data –> Convolutional Neural Network Model –> Discriminator-2

Note: The cells which contain (train\_generator.next () & validation\_generator.next ()) takes a bit of time to load 14000 &6000 images from the Google drive directory.

Coding instructions for GAN (.ipynb file):

1. Upload all the files in the given folder to the Google drive.
2. Open the uploaded GAN (.ipynb file) in the drive through Google Colab.
3. Change the run time setting for the notebook to access GPU computation power. Go to Runtime –> Change run time type –> Hardware accelerator –> GPU –> Save.
4. Mount the Google drive by running first cell of the notebook.
5. Change the path locations of the .csv files in the cell where we load the attribute of face images to their respective paths in the drive.
6. Change the path locations of the directory in the cells where flow from data frame function is used, to the respective path of the data set in the drive.
7. Run each cell one by one from the start to end of the notebook.

Note1: The last cell in the notebook is dedicated for training the GAN model hence, it takes a lot of time to train the model and generate images depending upon the number of epochs mentioned in the model.

Note2: Define less number of epochs and then run the last cell multiple times to analyze the changes in the generated images as number of epoch’s increases.

Note3: The generator and discriminator loss can be seen while training. Once the training is complete, only the generated images are shown.

Authors: Sai Preetham Chakka and Karthik Kumar Reddy Gangireddy