**CSC8637: Deep Learning - Coursework**

**Name: Parth Umeshkumar Shah**

**Student Number: 210468180**

**Implementation of CycleGAN model**

We trained the CycleGAN model to turn human faces into Cats & Dogs faces in this assignment.The human faces dataset had subfolders containing 256x256 photographs of celebrities, whereas the Cats & dogs dataset included images of various sizes.

**Preparing Data:** Firstly, by using the*tf.keras.preprocessing.image\_dataset\_from\_directory()* function, the data from each dataset was split into a training and a testing set with a validation split of 0.25, with the batch size setto be1. Next, a function was created that resized the images to size 64x64 and normalized further. For shuffling, this function was applied to each training and testing set, with a batch size of 1 and a buffer size of 100. Following that, pictures from the training data were visualized:

A collage of a person and a cat

Description automatically generated with low confidence

***Figure 1: Some images ofhuman faces and animal faces from the training data***

**Training the model:**The CycleGAN model was trained once the datasets had been prepared and pre-processed. The model was trained for 7 epochs, where the 8th epoch was stopped as parametrical losses were increasing and the outcome seemed to be less satisfactory in comparison to result of epochs too, the ModelCheckpoint() method was used to store the trained model's weights.

**Results:**The human faces test set was used to test the results. Also, we produced the results for some photos of the demonstrators and meas well:

Results of 7th run epoch are shown here:

A collage of a person

Description automatically generated with medium confidenceA collage of a person

Description automatically generated with medium confidence

A collage of two people

Description automatically generated with medium confidence

***Figure 2: The CycleGAN model tested on faces test set and some photos of the demonstrators and myself***

The outcomes were not typically satisfactory, since the created pictures were not clear, and there were no visible alterations for the demonstrators' photos. As a result, to achieve better outcomes, the model must be trained more thoroughly, mostly by increasing the number of epochs, if the losses don’t seem to increase continuously. Increasing the number of filters and undertaking data augmentation may also result in improved outcomes. Increasing the number of epochs and filters, on the other hand, would result in additional training time which again somehow contributes in less satisfactory image outcomes, hence the technique needs more research for better results and may we look to train some GIF or small videos in future sometime.

**Bibliography**

Team, K., n.d. Keras documentation: CycleGAN. [online] Keras.io. Available at: <https://keras.io/examples/generative/cyclegan/#create-a-callback-that-periodically-saves-generated-images>.

.TensorFlow. n.d. tf.keras.callbacks.ModelCheckpoint  |  TensorFlow Core v2.4.1. [online] Available at: <https://www.tensorflow.org/api_docs/python/tf/keras/callbacks/ModelCheckpoint>.

Chollet, F., 2017. Deep Learning with Python. [online] Manning Publications. Available at: <https://www.manning.com/books/deep-learning-with-python#toc>.