COMP343-Machine learning
Jiaxuan Chen
Final project Proposal
Topic: Mask detection classifier via TensorFlow

Pandemics put everyone on masks. In many public space entrance, it will be crucial to have some live detection software/ algorithm to check if people actually wear masks in order to stop the pandemics spread. Therefore, I would like to use what I've learnt in class (TensorFlow, neural networks) to write a python script that trains a classifier to achieve mask detection ,which can be used further for mask object detection in those scenarios (via Python module OpenCV).

The very first steps of project will be focused on preprocessing data. There are tons of photos with people wearing masks vs. not wearing masks. It is important to separate them out into two folders and bind them with correct label. Original TensorFlow workstations data is .xml, which is a type of text delimited file with (ROI, area of interest, in this case, people's face). We need to manually draw the binding boxes for people's face, and assign a label to each image in TensorFlow work station. I planned to write a xml_parsing python script to deal with mixed label images, and separate them out into two folders with their correct labels. After this initial steps, I will get two label image folder, one with_mask, one without_mask. Then, I will write the main project script, by importing those two data folders in and trained them with TensorFlow.

In regards to data preparation, I secured an image dataset that has over 7000 images with correct with_mask, without_mask label folder. I also found a raw TensorFlow workstation data (image folders and xml file folders) and I used xml_parsing.py and image_load.py to crop the images into same size and convert the NumPy array of images into csv.

After building up the basic framework of TensorFlow training using the tutorial (https://www.tensorflow.org/tutorials/load_data/images) provided for image training, I will try to use cross-validation, splitting k-folds, build different neural networks model, change the number of images in each layer to check performance and avoid overfitting. My current progress is setting up the basic framework and trying to do cross validation over that basic framework. Ideally, I can find a very high test accuracy that keeps up with the train accuracy.

Hopefully, I can finish these steps successfully and move on some extracurricular part of the project, tweaking OpenCV to achieve object detection using laptop built-in camera. This is the optimal if time permits. **At the bare minimum, I think my project should achieve:**

1. Usable image classifier for with_mask / without_mask
2. Compare performance of classifier using two different model buildup (the default in the tutorial is CNN)
3. Besides, try to tweak the hyperparameters as well, like epoch of algorithm, number of images in each layer, try different split etc. and discuss performance