**Project Name:** Face Mask Detection

**Submitted by:** Chandrika Deb (GitHub username: chandrikadeb7)

**Description:**

This project intends to develop a Face Mask Detection system using OpenCV, Keras/TensorFlow and Deep Learning. The system can be easily integrated to various embedded devices with limited computational capacity as it uses MobileNetV2 architecture. It will detect face masks in images as well as in real-time videos.

In recent times, where Covid-19 has impacted a domino effect on manufacturing, travel, tourism, hospitality, crippling the global economy. In addition to it, is the growing curve of human deaths across the globe due to the pandemic, this project which relies on computer vision and deep learning, intends to make an impact and solve the real-world problem of safety measures at some significant level.

This project can be used at airports, offices, hospitals and many more public places to ensure that the safety standards are maintained and people are abiding by the rules and regulations to wear protective masks at public places. If the detection system classifies as ‘No Mask’, reminders can be given as well as actions can be taken against such individuals.

This project uses recent techniques in the field of computer vision and deep learning. Custom dataset was made from scratch using Bing Search API, Kaggle datasets and RMFD dataset, and the evaluation of the model on test dataset was found consistent. The system correctly detected the presence of face masks on human faces that it detected in static images as well as real-time video streams.

To create our face mask detector, we trained a two-class model with images of people ***wearing masks*** and ***not wearing masks****.*

We then fine-tuned our model using MobileNetV2 on our ***mask/no mask***dataset and obtained an image classifier that was **93% accurate.**

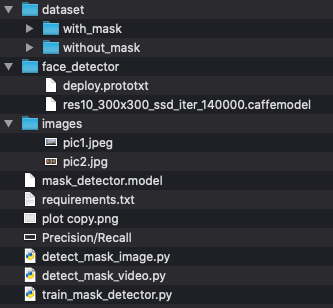
We then took this face mask classifier and applied it to both ***images*** and ***real-time video streams*** by:

1. Detecting faces in the images/video
2. Extracting each individual face ROI
3. Applying our face mask classifier

Our face mask detector is accurate, and since we used the MobileNetV2 architecture, it’s also computationally efficient and thus making it easier to deploy the model to embedded systems (Raspberry Pi, Google Coral, etc.).

This system can therefore be used in real-time applications which require face-mask detection for safety purposes due to the outbreak of Covid-19. This project can be integrated with embedded systems for application in airports, railway stations, offices, schools, and public places to ensure that public safety guidelines are followed.

Contents of Project Directory:



The contents of the project directory are as follows:

**dataset:** It has 2 folders - ‘with\_mask’ and ‘without\_mask’. The number of images are as follows:

with\_mask: 1916 images

without\_mask: 1919 images

**face\_detector:** This folder contains the OpenCV caffe model for face detection.

* The **.prototxt**file defines the model architecture (i.e., the layers)
* The **.caffemodel** file contains the weights for the actual layers

**images:** This folder contains images for detection of face masks on static input images in .jpg, .jpeg, .gif, or .png format.

**mask\_detector.model:** Our face mask classifier model developed using transfer learning on MobileNetV2

**requirements.txt:** The Python libraries dependencies are listed in this file along with their versions

**plot copy.png:** The training accuracy/loss curves are depicted in this image.

**Precision/Recall:** The accuracy scores are depicted in this image.

**detect\_mask\_image.py:** Python script to detect masks in static input images

**detect\_mask\_video.py:** Python script todetect masks in real time video streams.

**train\_mask\_detector.py:** Python script totrain our mask detector model.

**Steps for Installation & Working**

1. Clone the repo

$ git clone https://github.com/chandrikadeb7/Face-Mask-Detection.git

1. Change your directory to the cloned repo and create a Python virtual environment named 'test'

$ mkvirtualenv test

1. Now, run the following command in your Terminal/Command Prompt to install the libraries required

$ pip3 install -r requirements.txt

**Working**

1. Open terminal. Go into the cloned project directory folder and type the following command:

$ python3 train\_mask\_detector.py --dataset dataset

1. Now detect the face masks in images

$ python3 detect\_mask\_image.py --image images/pic1.jpeg

1. Detection in real-time video streams

$ python3 detect\_mask\_video.py