Addis Ababa Science and Technology University

College of Electrical and Mechanical Engineering

Department of Electrical and Computer Engineering

Computer Engineering Stream

Computer Vision

Term Project Report

Title: Real-Time Face Mask Detection System with Alarm Using Deep Learning

By

Sileshi Nibret

Submitted to: Dr. Beakal Gizachew

March 2021

Table of Contents

List of Figures

Introduction

The World Health Organization (WHO) has urged numerous countries to ensure that their people wear masks in public places due to the transmission of coronavirus disease (COVID-19) [1]. Research studies show the effectiveness of N95 and surgical masks in preventing virus transmission are 91% and 68% respectively [2]. Deep Learning has proved its effectiveness in image analysis for identification and classification as technology has progressed. Deep learning advancement with the integration of computer vision offers the breakthrough in development in countless fields of technology. Deep learning techniques can be used to discern facial recognition and determine whether or not the individual is wearing a facemask. An initiative was started by the French government to identify passengers who are not wearing masks in the metro station. For this initiative, an AI software was built and integrated with security cameras in Paris metro stations [4].

The main aim of this work is to develop a deep learning model for the detection of whether a person is wearing a face mask or not. The proposed model uses the transfer learning of MobileNet and ResNet50 to identify persons who are not wearing a mask in public places. The system can be integrated with surveillance cameras

Methodology

In doing this project we have followed the following methodology to detect whether the person is wearing facemask or not.

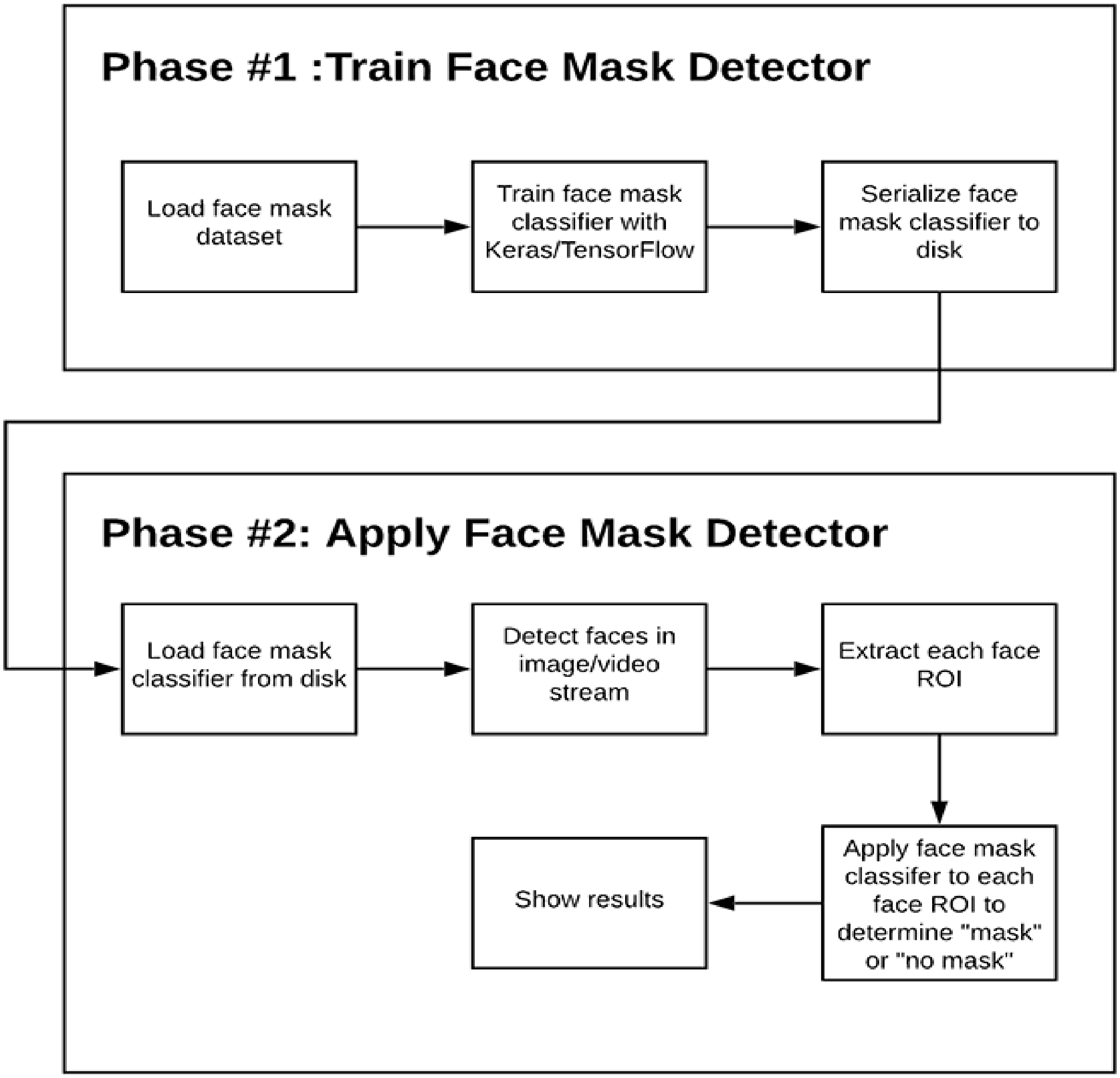

Phase #1 :Train Face Mask Detector
Load face mask dataset
Train face mask classifier with Keras/TensorFlow
Serialize face mask classifier to disk
Phase #2: Apply Face Mask Detector
Load face mask classifier from disk
Detect faces in image/video stream
Extract each face ROI
Apply face mask classifer to each face ROI to determine "mask" or "no mask"
Show results

Figure 1: Working method of proposed system

Data collection

Data is the heart of computer vision and deep learning projects. In this project face data with mask and without mask are used.

Faces with mask is collected from different researchers who make public the data in Kaggle. The images in the dataset are collected on different scenarios like outdoor, luminated grouped and others. These images make the model more general and robust.

Non masked in collecting non masked face image I tried to integrate generative adversarial network (**GAN**) to generate new faces using the existing images to test the flavor of GAN and to have some knowledge about the network. In doing so I tried to generate the faces from class mates face image collected previously for another assignment. But the process is resource consuming and expensive.

How does GANs work?

GANs learn a probability distribution of a dataset by pitting two neural networks against each other. one neural network, called the Generator, generates new data instances, while the other, the Discriminator, evaluates them for authenticity; i.e., the discriminator decides whether each instance of data that it reviews belongs to the actual training dataset or not.

Meanwhile, the generator is creating new fake images that it passes to the discriminator. The goal of the generator is to generate passable images: to lie without being caught. However, the goal of the discriminator is to identify images coming from the generator as fake.

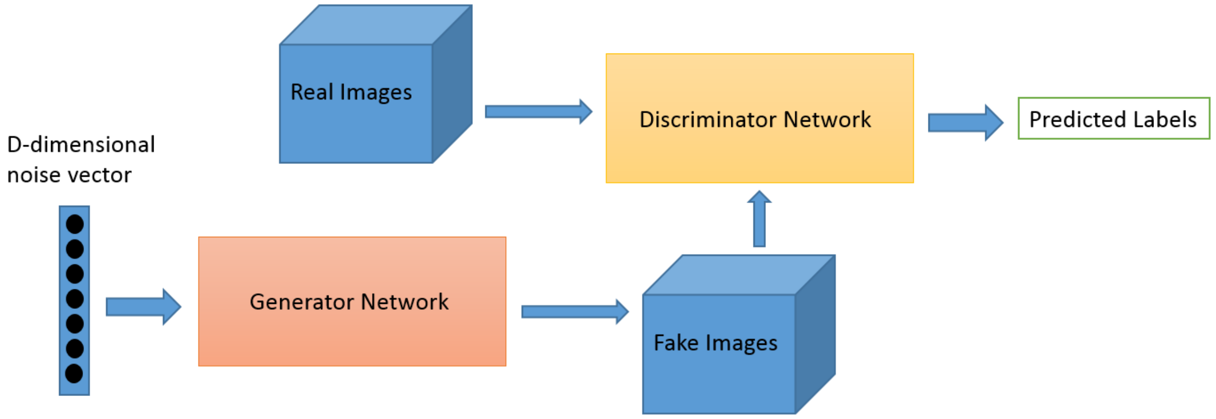


Figure 2: GAN Architecture

Here are the steps a GAN takes:

1. The generator takes in random numbers and returns an image.
2. This generated image is fed into the discriminator alongside a stream of images taken from the actual, ground-truth dataset.
3. The discriminator takes in both real and fake images and returns probabilities, a number between 0 and 1, with 1 representing a prediction of authenticity and 0 representing fake.

4. Discriminator’s job is to perform Binary Classification to detect between Real and Fake so its loss function is Binary Cross Entropy.

Generator

The Generator is a neural network with fully connected input layer and seven hidden layers. It takes in input and produces a 128 ×128 × 3 image. In the hidden layer, I used the Leaky ReLU activation function. Unlike a regular ReLU function, which maps any negative input to 0, ***Leaky ReLU*** allows a small positive gradient that prevents gradients from dying out during training, which tends to yield better training outcomes.

At the output layer, I employ the ***tanh*** activation function, which scales the output values to the range [–1, 1]. The reason for using tanh (as opposed to, say, sigmoid, which would output values in the more typical 0 to 1 range) is that tanh tends to produce crisper images.

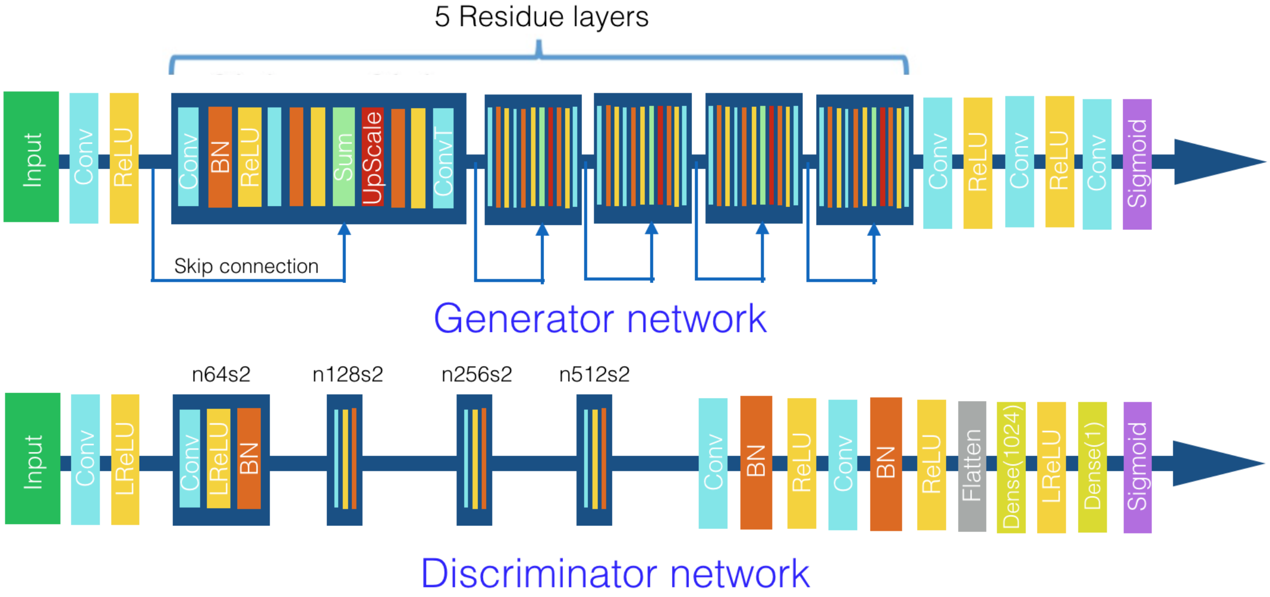


Figure 3:DCGAN image processing architecture [5]

Discriminator:

The discriminator network consists of convolutional layers the same as the generator. For every layer of the network, we are going to perform a convolution, then we are going to perform batch normalization to make the network faster and more accurate and finally, a Leaky ReLu were performed.

Batch normalization is a method that reduces the covariate shift (variations in input value distributions between layers during training) in neural networks by normalizing the output of each layer before it is passed as input to the next layer.

The following image is generated using DCGAN after 10000 epochs which take 12 hours of execution using Google Colab GPU. As we can see it seems working but can't be taken as input to our model.

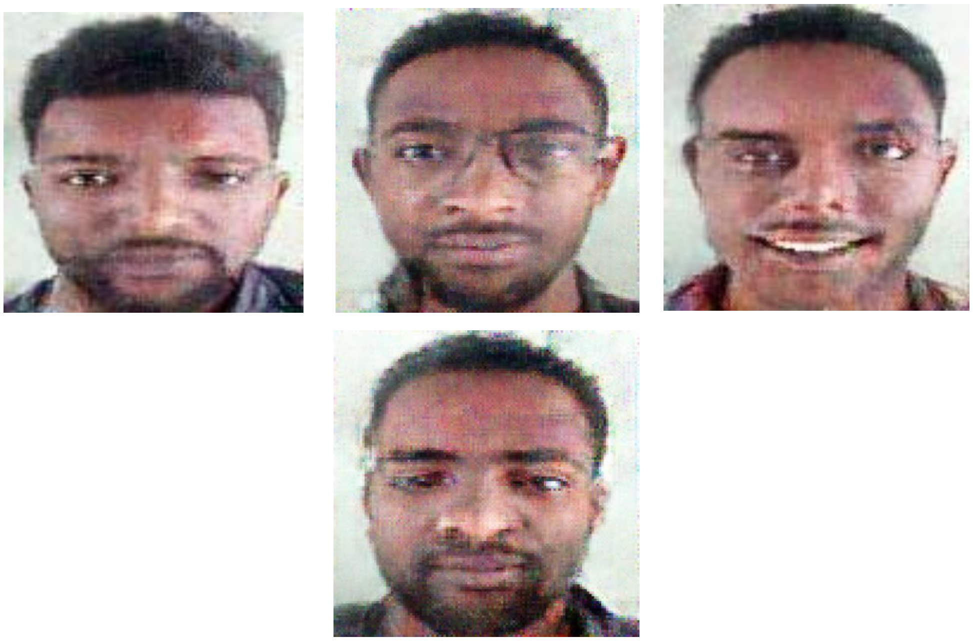


Figure 4:GAN generated fake faces from classmate's face image

Generally,3717 images with mask and 1075 non-masked images collected from different researchers are used. Since the face detection model is a pretrained model using small number of non-masked image is considered.

Image processing

Before training the model to detect whether the person is wearing mask or not the training image is resized to detect the ROI (region of interest) which is the face of the person. This process is done by the model automatically. The processes like image augmentation are also preprocessing in the augmentation technique images are rotates zoomed and shifted. Image augmentation techniques are used to increase the diversity of the training data for enhancing the performance of the proposed model.

Face Detection

Face detection is usually the first step towards many face-related projects, such as face recognition or verification. In this new COVID era face mask detection uses this technique, to detect face mask detecting the face is the primary task of the system. Face detection is among the focuses of the automatic surveillance system. This focus is based on premise that face of a person can be applied into several application to extract useful information.

In this project I used Caffe (Convolutional architecture for fast feature embedding) is a deep neural network module and Caffe models. The model is pretrained model using single shot detector. Training an SSD model from scratch will require a lot of data, so here I have imported pretrained weights **(Caffe Face Detector Model)** using OpenCV.The two components of the detection model are prototxt and caffemodel file.

P**rototxt** file define the *model architecture* which is the layers of the deep neural network themselves. However, **caffemodel** file contains the *weights* for the actual layers. The **caffemodel** is pretrained model of *ResNet* with 300x300 dimension for iteration based on single shot detection learning. According to [2] using a **base architecture of VGG-16 Architecture**, SSD is able to outperform other object detectors like YOLO and Faster R-CNN in both speed and accuracy. Therefore, pretrained model for face detection is the preferable than other models.

How does face detection work?

- ✓ blobFromImage creates 4-dimensional blob from image. Optionally resizes and crops image from center, subtract mean values, scales values by scale factor, swap Blue and Red channels.
- ✓ The blob is passed through the network and detections are made with some confidence score.
- ✓ Define a threshold confidence score, above this threshold the detection will be considered as a candidate of being a face. (In this case threshold confidence= 0.5)
- ✓ All the detections that qualify the confidence score are then passed to the architecture for boundary mark of the face or non-face image are not boxed to mark the face.

Testing Face Detector model

The face detector model described above is implemented in the project the face detector detects faces in an image correctly.

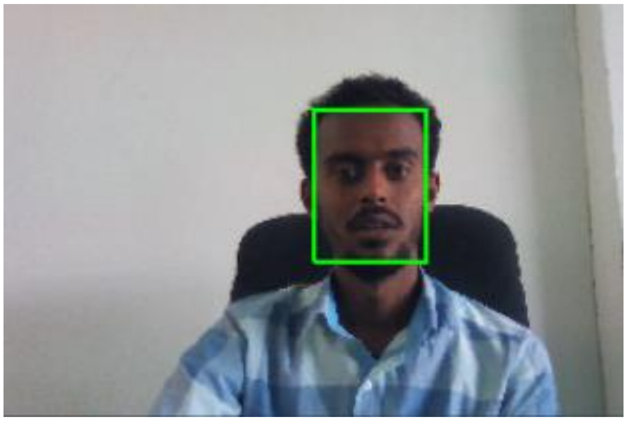


Figure 5:Face detection for single image

The following image is the result of the detector in a group of people.

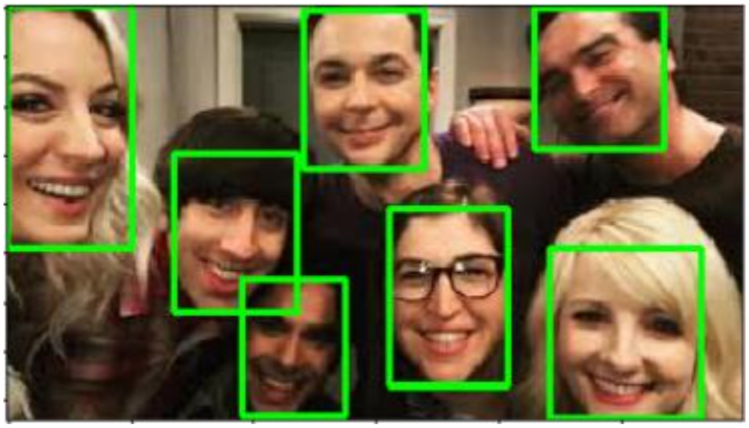


Figure 6:Face detection in group image (favorite TV show BBT)

Face Mask Detection

To detect facemask in real time using deep learning models the identifying the learning model is the first task. In this project I have used pretrained model that is **MobileNetV2** And **ResNet50** networks architecture. The network is hyper tuned to classify the image by implementing the output layer of the network. To increase the generalizability of the model I used image augmentation techniques i.e. I have done rotation, zooming, width shift, height shift, shear and horizontal flip randomly on the given training data.

MobileNetV2 is a convolutional neural network architecture that seeks to perform well on mobile devices. The intermediate expansion layer uses lightweight depth wise convolutions to filter features as a source of non-linearity. As a whole, the architecture of MobileNetV2 contains the initial fully convolution layer with 32 filters, followed by 19 residual bottleneck layers.

In my project I tuned the output layer using softmax activation function, dropout of 0.5 and for binary classification.

This model is lightweight and good choice for real-time application like this one (face mask detection) that can be implemented on simple devices like CCTV camera.

Running the model for 20 epochs with 32 batch size and 0.0001 learning rate, it takes approximately ***56.25 Minute.***

ResNet50 is a convolutional neural network that is 50 layers deep. It has a greater number of parameters to be used so it is obvious that it will show better performance as compared to the MobileNetV2. However, learning takes much time than MobileNetV2.

I run the model for 20 epochs with 64 batch size and 0.0001 learning rate. The time taken to train is about 3.9 hour

Results and Conclusion

After training both models I have tested the system in real-time using my laptop camera for both image input and real-time video stream.

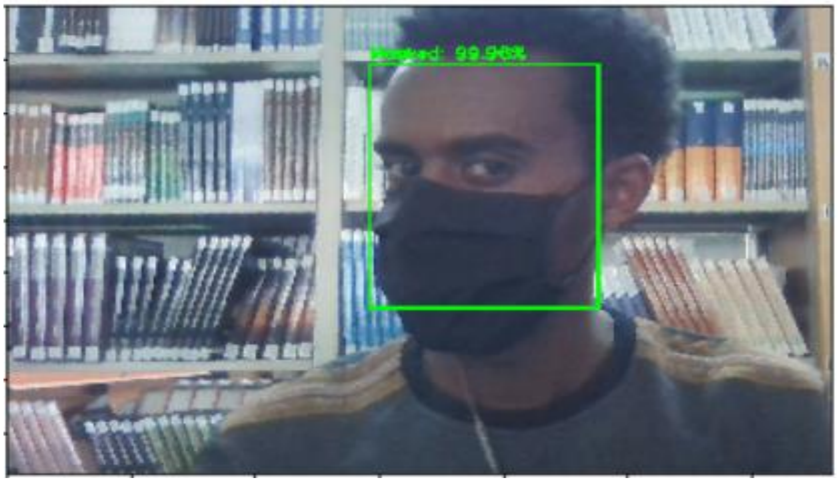


Figure 7:Face mask detected (Green Box)

The above result shows the model detected me I am wearing mask with 99.9 confidence.

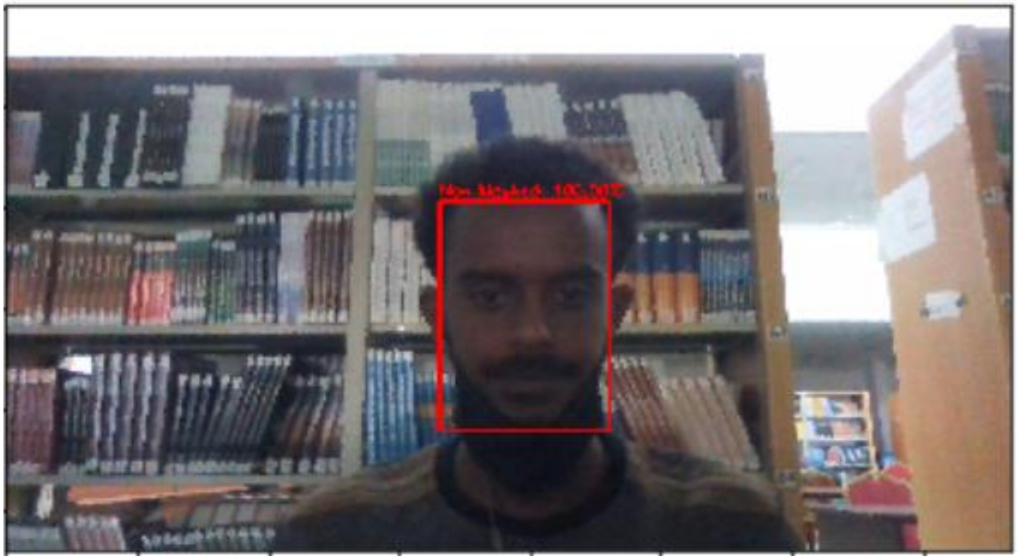


Figure 8:Face mask not detected (Red Box)

The above figure showed that I am not wearing mask that is why it gives red box.

Classification Results of Models

Performance metrics used to compare the models

Precision: is the ratio of *True Positives* to all the positives predicted by the model.

$$\boldsymbol{Precision} = \frac{\boldsymbol{True\ Positive}}{\boldsymbol{True\ Positive + False\ Positive}}$$

The more False positives the model predicts, the lower the precision is.

Recall: is the ratio of *True Positives* to all the positives in your Dataset.

$$\textbf{Recall} = \frac{\textit{True Positive}}{\textit{True Positive} + \textit{False Negative}}$$

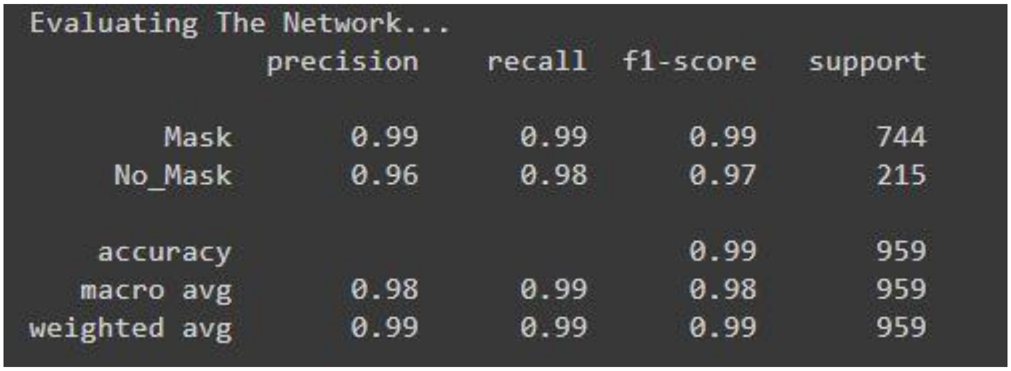
The more false Negatives the model predicts, the lower the recall.

F-Measure provides a single score that balances both the concerns of precision and recall in one number.

$$\text{F1} = 2 * \frac{\textit{Precision}}{\textit{Precision}+\textit{Recall}}$$

A good F1 score means that you have low false positives and low false negatives

Using the above performance metrics MobileNetV2 takes shorter time to train but the accuracy achieved is as follows



```
Evaluating The Network...
              precision    recall  f1-score   support

        Mask       0.99      0.99      0.99       744
     No_Mask       0.96      0.98      0.97       215

    accuracy                           0.99       959
   macro avg       0.98      0.99      0.98       959
weighted avg       0.99      0.99      0.99       959
```

Figure 9:Metrics evaluation for masked and non-masked image

Accuracy/Loss plot

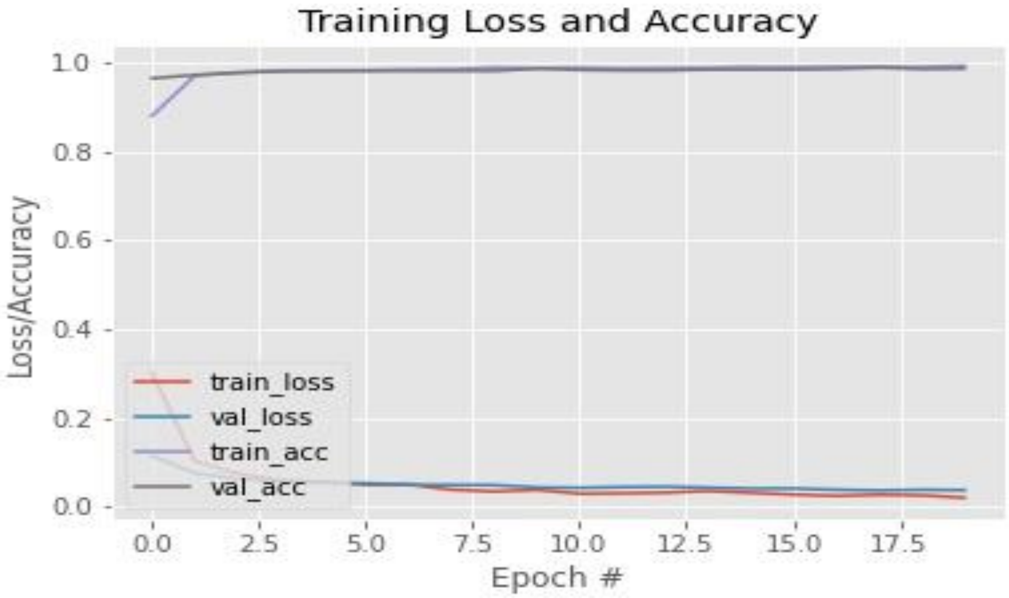
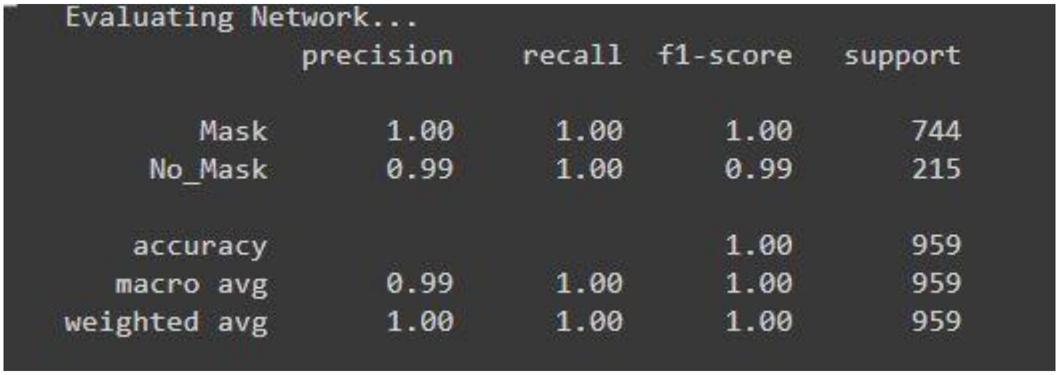


Figure 10:Plot of accuracy vs loss in training and testing

ResNet50 take longer time to train and classify but achieved accuracy of 100 % in mask and 99 % in non-masked images.



```
 Evaluating Network...
                precision    recall  f1-score   support

         Mask       1.00      1.00      1.00       744
      No_Mask       0.99      1.00      0.99       215

     accuracy                           1.00       959
    macro avg       0.99      1.00      1.00       959
 weighted avg       1.00      1.00      1.00       959
```

Figure 11:Metrics evaluation for masked and non-masked image

Conclusion

In the era of COVID-19, wearing facemask decreases the transmision of the virus from one person to another. In public areas like train stations, hospitals and other public areas detecting whether a person is wearing face mask or not is tedious task. Therefore, an effective face mask detection system may be the key for many application. In this project, I implemented various deep learning method for real-time detection with alarming system that is able to determine if someone is wearing a mask or not. From our experiments, transfer learning using two models are experimented and accuracy of 100% detection is atchived when using ResNet50 network and 98% detection accuracy is recorded for MobileNetV2 network.

References

[1] Chowdary and G. Jignesh, "Face mask detection using transfer learning of inception v3," in *International Conference on Big Data Analytics*, Cham, 2020 .

[2] T. Sik-Ho, "towardsdatascience," 3 Nov 2018. [Online]. Available: https://towardsdatascience.com/review-ssd-single-shot-detector-object-detection-851a94607d11.

[3] Roy, Biparnak, et al. "MOXA: A Deep Learning-Based Unmanned Approach for Real-Time Monitoring of People Wearing Medical Masks." Transactions of the Indian National Academy of Engineering 5.3 (2020): 509-518.

[4] Paris tests face-mask recognition software on metro riders, http://bloomberg.com/, online accessed

March 20, 2021

[5] Qiaojing Yan,Wei Wang, "DCGANs for image super-resolution, denoising and deblurring", Stanford University