REALTIME FACE MASK DETECTION USING DEEP LEARNING

Seminar (IT290) Report

Submitted in partial fulfilment of the requirements for the degree of

BACHELOR OF TECHNOLOGY

In

INFORMATION TECHNOLOGY

by

APRAMEYA DASH (191IT209)

DEPARTMENT OF INFORMATION TECHNOLOGY

NATIONAL INSTITUTE OF TECHNOLOGY KARNATAKA

SURATHKAL, MANGALORE -575025

APRIL, 2021

DECLARATION

I hereby *declare* that the *Seminar (IT290) Report* entitled **REALTIME FACE MASK DETECTION USING DEEP LEARNING** which is being submitted to the National Institute of Technology Karnataka Surathkal, in partial fulfilment of the requirements for the award of the Degree of Bachelor of Technology in the department of Information Technology, is a ***bonafide report of the work carried out by me.*** The material contained in this seminar report has not been submitted to any University or Institution for the award of any degree.

Aprameya Dash (191IT209)

Aprameya Dash

Place	:	NITK, Surathkal
Date	:	30th April, 2021

CERTIFICATE

This is to certify that the Seminar entitled **"Real time Face Mask Detection using Deep Learning"** has been presented by *Aprameya Dash (191IT209)*, a student of IV semester B.Tech. (IT), Department of Information Technology, National Institute of Technology Karnataka, Surathkal, on 30th April, 2021, during the even semester of the academic year 2020 - 2021, in partial fulfilment of the requirements for the award of the degree of Bachelor of Technology in Information Technology.

Guide Name: **Prof. Shrutilipi Bhattacharjee**

(Signature of the Guide with Date)

Place: NITK, Surathkal

Date: 30th April, 2021

ABSTRACT

The COVID 19 pandemic has caused a dramatic loss of human lives across the globe and has presented an unprecedented challenge to public health, small and large businesses and various other sectors. The economic and social disruption caused by this pandemic has been catastrophic as numerous people have lost their livelihoods and many enterprises have gone bankrupt. A big reason behind this pandemic is the highly contagious nature of the disease, due to which millions of people are at risk of developing this disease. The best way for protecting ourselves from such highly contagious diseases and preventing the transmission of such diseases is wearing face masks properly in public places. Face masks provide protection against highly contagious diseases and have many other applications such as protection from fumes, smoke, dangerous environments etc. Thus, ensuring that people are wearing face masks in public places and industries is necessary. Fortunately, we can use deep learning to do such intensive tasks automatically with high accuracy and less response time. This paper proposes a method to perform automatic and real-time face mask detection using various techniques such as deep learning and image processing. A deep learning model, based on MobileNetV2 architecture, has been trained on a dataset containing a large number of relevant images. This trained model can, then, be used to detect the presence or absence of face masks in images, videos as well as in real-time video stream. The trained model achieves very high accuracy with 99.52% training accuracy and 99% validation accuracy. Further, due to the use of MobileNetV2 architecture, the trained model is a very light weight model and can be easily and efficiently integrated even with mobile devices. The model has been created using the Keras framework of TensorFlow library and all image processing works have been done using the OpenCV library. Thus, due to automatic and accurate monitoring using the proposed method, transmission of various contagious diseases can be slowed down and prevented and safety of workers in industries and harmful environments can be enhanced.

TABLE OF CONTENTS

LIST OF FIGURES

Chapter 1 INTRODUCTION

The COVID 19 pandemic has created a huge health crisis in the present scenario by disrupting the social and economic conditions all over the globe and burdening the healthcare system established in every country. A huge number of people have been infected by the disease till now and a large number of deaths have also happened due to the pandemic. Further, the pandemic has also caused various industries, institutions and businesses to shut down, therefore, causing huge losses to various organizations. One of the biggest reasons behind this uncontrollable increase in the transmission rate is that the disease is highly contagious and can easily pass from one person to another. Apart from COVID 19, there are many other diseases like influenza, SARS etc. which are highly contagious and can easily transmit from an infected person to a healthy person.

To combat the transmission and spread of the pathogens causing such highly contagious diseases, it has become a necessity to enforce proper rules and regulations in public places all over the world and one of the most important methods for controlling the transmission of this disease is wearing face masks properly in public places. Many countries, private organizations and communities have made it mandatory to wear face masks in public places. Thus, face mask detection has become crucial task to prevent or slow down the spread of many contagious diseases as well as to ensure that all created rules and regulations are being enforced properly. Apart from healthcare sector, there are many other fields which can benefit from face mask detection systems. Facial masks also provide protection from polluted air, fumes, smoke, chemicals and hazardous environments. So, integrating face mask detection systems in industries that deal with fatal chemicals, mining fields etc. will help to ensure that people who enter such dangerous environments are equipped with proper protective gears. This can further prevent a lot of casualties that may happen due to inhaling of dangerous gases, fumes and chemicals.

Thus, there is no doubt that installation of proper face mask detection systems in various places such as public places, industries, mining fields etc. is necessary to reduce the number of casualties by preventing the transmission of contagious diseases, ensuring that people enter dangerous environments with proper protection and many other ways. But, detecting face masks manually can be a really painful, intensive and expensive task, especially if it is carried out in public places where a large number of people have to be kept under surveillance continuously for extended periods of time. Thus, in order to solve this problem, an automated system is required which can perform face mask detection using minimum number of resources and that too, with high accuracy and high success rate.

Fortunately, due to the advancement of technology, we can create an automated model for face mask detection by integrating Deep learning methods and Computer vision techniques. Deep learning is a subclass of machine learning which is based upon utilization of algorithms that imitate the functioning of human brain in processing data and creating patterns for use in decision making. When integrated with various Computer Vision techniques, deep learning can be used to perform manual tasks like face mask detection with high accuracy and low response time. In Deep learning, a Convolutional Neural Network (CNN), is a subclass of Deep Learning Networks (DNNs) which specialize in image classification and require very little preprocessing as compared to other similar algorithms. Thus, after training a CNN model, it can be used to detect face masks in images which is basically a type of image classification, something in which CNNs specialize.

In this paper, a face mask detection model is proposed, which is able to detect face masks using deep learning and computer vision algorithms. This model can be used to detect face masks in images, videos as well as real time video streams and thus, it can be used for a variety of applications. It basically serves as a classifier which can take a set of images containing faces of people as input and then analyses the images to decide which images belongs to the class 'with mask' and which images belong to the class 'without masks'. For face mask detection from videos or real time video streams, frames of the video are extracted after a fixed interval of time and face mask detection is performed on each of the extracted frames.

The implementation of the deep learning model has been done using Keras framework of TensorFlow library, which is a very popular machine learning and deep learning library. Keras is one of the most popular deep learning frameworks because of its smaller learning curve, focus on user experience and huge deployment capabilities. Built on top of TensorFlow, Keras is a high-level API which enables fast experimentation through a high level, user-friendly, modular and extensible API, capable of running on both CPU and GPU. Further, OpenCV library has been used to deal with all image processing and computer vision challenges involved in the project. OpenCV is a cross platform open-source library consisting of a huge number of programming functions mainly aimed at real-time computer vision. Furthermore, Python programming language has been used for the implementation of the project due to its huge simplicity, readability, flexibility, platform independence and great ecosystem of libraries and packages. Due to so many advantages, python is a great choice for deep learning.

The proposed model is based on MobileNetV2 architecture which ensures that the final trained model gives high accuracy even though the model is a light weight model. MobileNetV2 is a neural network architecture, officially supported by TensorFlow, which is extensively used for various purposes like image classification, object detection etc. The architecture delivers high accuracy results while keeping the parameters and mathematical operations as low as possible to bring deep neural networks to mobile devices. This ensures that the final model is as light weight as possible but at the same time the model gives high accuracy and performance. The proposed face mask detection model exploits the advantages of transfer learning using the pretrained MobileNetV2 architecture after excluding the classification layers at the top. The MobileNetV2 model is then modified by adding a few custom layers to it and then it is trained on a dataset so that the final model becomes suitable to be used for face mask detection.

This trained model can then be used for face mask detection on an image or a set of images (in case of video). But, before this, face detection has to be performed on the input image. This is done by using the preexisting face detection DNN already present in OpenCV which can detect faces in a given image. These detected faces are then fed into the trained face mask detector model which makes and displays the final decision (‘with mask’ or ‘without mask’). Experimental results show that the proposed face detector model achieves excellent results with 99.52% training accuracy and 99% validation accuracy.

This lightweight proposed model can be installed at various places like public places, industries, mining fields etc to detect face masks and prevent transmission of diseases as well as loss of lives. It can be integrated with devices such as CCTV cameras to encourage wearing of masks, track safety violations and ensure proper protection in hazardous environments.

Chapter 2 LITERATURE REVIEW

[1] In this paper, a novel model for medical masked face detection has been created. For image detection, a YOLO v2 based ResNet-50 model has been used to produce high-performance outcomes. Data augmentation has also been used as it is a method that can be used to artificially increase the diversity of datasets for training detectors. The proposed model improves detection by introducing mean IoU to estimate the number of anchor boxes. To train and validate detector in a supervised state, a new dataset has been designed, based on two public masked face datasets. The dataset has been split up into 70% training images, 10% validation images, and 20% testing images. Performance metrics such as AP and log-average miss rates score had been studied for SGDM and Adam optimizer experiments. YOLO-v2 detector is a good model to find a masked face in an input image based on ResNet-50. This method achieved an accuracy of 81% using YOLOv2 with ResNet-50.

[2] In this paper, an SR network has been combined with a classification network (SRC-Net) for facial image classification. To identify face-mask-wearing condition, the proposed methodology involves four major steps: input image pre-processing, facial detection and cropping, image super resolution and facemask-wearing condition identification. SRC-Net got 98.70% accuracy and out-performed traditional end-to-end image classification methods by over 1.5% in kappa. A public dataset, Medical Masks Dataset, containing 3835 images with 671 images of no facemask-wearing, 134 images of incorrect facemask-wearing, and 3030 images of correct facemask-wearing is used to train and evaluate the model. As for the proposed algorithm, the identification for image, where an average of 10 images can be identified in a second, does not meet the video frame rate of 24 (fps). Hence, SR network with a classification network (SRC-Net) for facial image classification has been used along with the proposed algorithm, image pre-processing, datasets, facial detection, cropping and SR networks. This approach achieved an accuracy of 98.70%.

[3] This paper uses an approach which involves training a face mask detector and then, implementing face mask detection using the trained model. The faces are detected in images and

video and then the region of interest (ROI) is extracted. In this approach, the training model uses Facemask net. MATLAB programming is used to build the facemask detector model. The trained model achieved an accuracy of 98.6 %. This classifier is then applied for face mask detection in images and live video streams. The faces are recognized in images and videos and these faces are extracted. The face mask identifier is not complicated in structure and gives results and hence can be used in CCTV footages to identify whether a person is wearing a mask correctly. Mass screening can be done in crowded places like railway stations, bus stops, markets, streets, mall entrances, schools, colleges, etc.

[4] In this paper, a deep learning model has been created that can detects a person who is not wearing a mask. This deep learning model is built using transfer learning with the help of InceptionV3 architecture. In this approach, image augmentation techniques are used to boost the performance of the model as this increases the diversity of the training data. Image augmentation is used to increase the diversity of the dataset by introducing random mutations to the images. Images are augmented with eight distinct operations: zooming, blurring, contrasting, flipping horizontally, rotating and sharing. The model is trained for 80 epochs with each epoch having 42 steps. In this approach, InceptionV3 pre-trained model is used to detect people who are not wearing face masks. For this to work, the last layer of the InceptionV3 pre trained model is removed and is finetuned by adding 5 more layers to the network. The model achieved a training accuracy of 99.92%.

The survey displayed that various methods and approaches can be used to perform face mask detection. Transfer learning methods can be used to make the face mask detection model even better. Many papers actually used transfer learning to create the face mask detection model. Many others used CNNs and other similar ways for the same purpose. But this mostly results into the formation of a model which will not be lightweight and would require high-end hardware support to function properly. Further, such models can take a lot of time to train and predict results. Thus, for efficient face mask detection, a system is required which is light weight, produces good accuracy and can be easily integrated with devices such as CCTV cameras, embedded devices etc.

Chapter 3 TECHNICAL DISCUSSION

3.1 METHODOLOGY

This section contains a detailed explanation about the creation, implementation and functioning of the face mask detection model. This entire process involves various smaller sub processes which have been integrated to create the final system.

3.1.1 Creation and training of face mask detection model

The first major step for creating the proposed face mask detection system is creating and training a deep learning-based model which is capable of taking images as input and classifying the input images into either of the two classes: ‘with mask’ or ‘without mask’. This step involves various smaller steps like creating a dataset from which the deep leaning based model can be trained, preprocessing all images present in dataset, performing one hot encoding of the labels of the images, and finally, defining and training the model which will be responsible for performing face mask detection by classifying images into either of the two mentioned classes.

A. Dataset

Creation of a proper and well-balanced dataset is a very important part of creating any deep learning model. A dataset is a collection of data which will be used by the final deep learning model to train itself and eventually get better at what the model is trained to do. Similarly, for creating a face mask detector model, a dataset containing a large number of images is created. The dataset contains two major sections: training dataset and testing dataset. The training dataset is the one which is used to train the deep learning model whereas the testing dataset is the one which is used to evaluate how well the model has been trained using the training dataset. For creating the face mask detection model, a dataset containing a total of 2,414 images was created. These images were collected from various sources and each of the images belong to one of the two classes: ‘with mask’ or ‘without mask’. Out of the 2414 images, the training dataset contains 1914 images which accounts for about 80% of the total data. Further, inside the training dataset, 958 images belong to the class ‘with mask’ whereas 956 images belong to the class ‘without mask’. The testing dataset contains the remaining 500 images which accounts for about 20% of the total images present in the dataset. Inside the testing dataset, 250 images belong to the class 'with mask' and the remaining 250 images belong to the class 'without mask'.

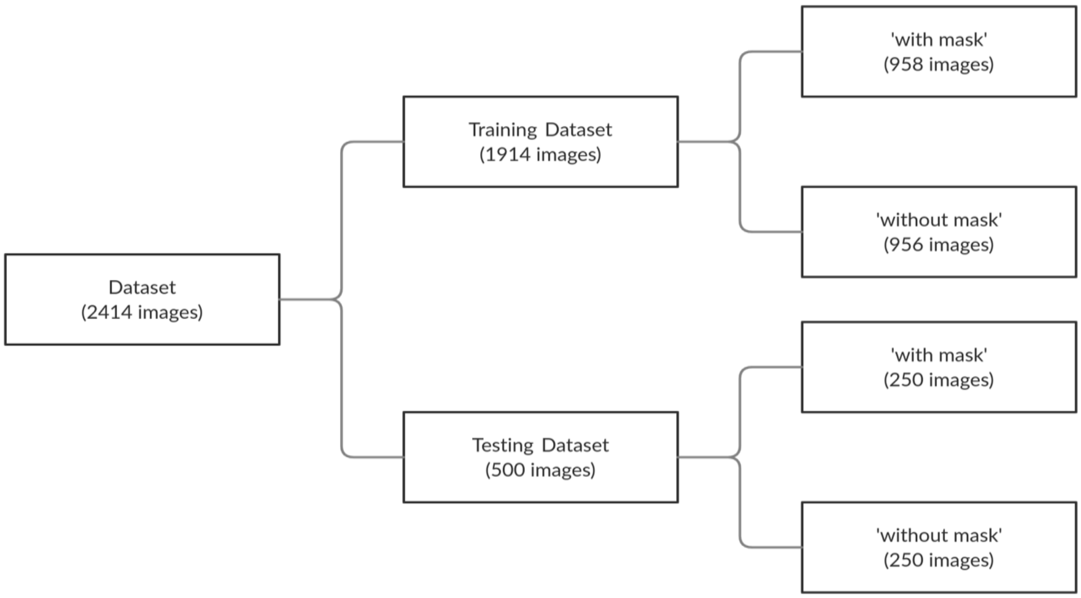

Figure 3.1: Structure of dataset

'With Mask' 'Without Mask'

Figure 3.2: Samples from dataset

Thus, the entire dataset has been divided into two parts: training containing about 80% of the total images and testing containing remaining 20% of the total images which is good enough. Further, each section of dataset contains almost equal number of images belonging to each of the two classes (‘with mask’ and ‘without mask’). This ensures that neither of the two classes is dominating over the other class. Thus, the created dataset is a well-balanced collection of data and it can be now used for the further processes.

B. Preprocessing of images

The next step after creation of a dataset is performing preprocessing of the images present in the dataset. In this step, each of the images has to be converted into a form which can be used to train the final model. For this, first, the paths of all images present in training dataset are collected. Then, for each image path in the list of paths, the image is loaded and resized to the dimensions (224 x 224 x 3). Next the image is converted to a numpy array and then the image is preprocessed so that it can be converted into a form which is suitable for MobileNetV2 architecture. Finally, a list of preprocessed images from training dataset is obtained. The same process is applied for the images present in the testing dataset which gives another list of preprocessed images from testing dataset. Simultaneously, two separate lists containing the labels or categories of the images from training dataset and the testing dataset respectively is also maintained. The label for an image can be ‘with_mask’ or ‘without_mask’.

C. Performing one hot encoding on labels

The next step involves converting the labels of the images into a vector which is more suitable for deep learning algorithms. One hot encoding is a process by which categorical variables are converted into a form that could be provided to ML algorithms to do a better job in prediction. Till now, the label for an image is either ‘with_mask’ or ‘without_mask’. To make prediction easier, these labels are converted into vectors. Considering ‘with_mask’ as class 0 and ‘without_mask’ as class 1, the one hot encoded vector for the class ‘with_mask’ will be [1, 0] whereas the one hot encoded vector for the class ‘without_mask’ is [0,1]. This way, the label for each of the images is converted into a vector to make class prediction easier.

D. Defining face mask detection model

The face mask detection model proposed in this paper uses a MobileNetV2 based architecture. MobileNetV2 is a neural network architecture which is widely used for image classification tasks and it is officially supported by TensorFlow. The face mask detection model consists of the following layers:

- **MobileNetV2:** MobileNetV2 architecture forms the base of the face detection model. MobileNetV2 architecture is loaded along with pretrained weights of ImageNet from TensorFlow after excluding the topmost fully connected layer at the top. The pretrained weights make the MobileNetV2 model suitable to perform image classification through transfer learning. The topmost layer of the MobileNetV2 has to be removed as we want to add our own custom layers at the top of the model to make it suitable for classifying images according to our requirement.
- **GlobalAveragePooling2D layer:** This layer is the next layer on top of the MobileNetV2 model. This layer performs global average pooling on the spatial data received from the previous layers. This layer helps in reducing the number of parameters involved while passing the data to the next levels and thus, it helps in preventing overfitting of the model.
- **Dense layer:** The next layer is a dense layer, that is, a fully connected network consisting of 64 neurons. This layer uses 'ReLU' or rectified linear activation function as the activation function.
- **Dropout layer:** The dense layer is followed by a dropout layer which helps in preventing overfitting by randomly dropping out (ignoring) 20% of the inputs.
- **Dense layer:** The final layer of the model is a dense layer consisting of two neurons. 'Softmax' activation function is used for this final layer. This ensures that for every prediction, the model will assign a probability to each of the two classes involved in the classification.

```
Model: "sequential"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
mobilenetv2_1.00_224 (Model) (None, 7, 7, 1280)        2257984
_________________________________________________________________
global_average_pooling2d (Gl (None, 1280)              0
_________________________________________________________________
dense (Dense)                (None, 64)                81984
_________________________________________________________________
dropout (Dropout)            (None, 64)                0
_________________________________________________________________
dense_1 (Dense)              (None, 2)                 130
=================================================================
Total params: 2,340,098
Trainable params: 82,114
Non-trainable params: 2,257,984
_________________________________________________________________
```

Figure 3.3: Architecture of face mask detection model

E. Training the face mask detection model

After defining the model, the next step is training the model. Before training the model, data augmentation is performed on the list of processed images from the training dataset. This performs random mutations to the images, thus, helping in increasing the variance of the dataset. Then, the model is trained for 20 epochs. In each epoch, first the model is trained using the training dataset and then, the model is evaluated using the testing dataset. The model is trained using Adam optimizer with a learning rate of 0.0001 and binary cross-entropy is used as the loss function. The final model is saved in local storage for further use.

3.1.2 Deploying the trained face mask detection model

After the creation of deep leaning based face mask detection model, the next step is deploying the model for practical use. This involves performing face mask detection on images, videos and real time video streams.

A. Loading neural network models

Before performing face mask detection on any data, all neural network models necessary for this process require to be loaded. The following two neural networks have to be loaded before deploying the system:

- **MobileNetV2 based model for Face mask detection:**

  This is the same deep learning model that has been previously trained to perform face mask detection and then stored in local storage.

- **OpenCV DNN for face detection:**

  Before performing face mask detection on a image, all images of faces present in the input image have to be detected. Fortunately, OpenCV image processing library has its own face detection neural network which can be directly loaded to perform face detection on any input image with high accuracy. This face detection neural network consists of two files which are available in the source repository of OpenCV. The first file is the model's 'prototxt' file which contains information about the architecture of the face detection neural network. The second file is the model's 'caffe-model' file which contains all the trained weights of the neural network. These two files can be loaded and then integrated using OpenCV to recreate the actual DNN which can then be used for detecting faces in input images.

B. Face mask detection in images

This section shows the way in which face mask detection is performed on images. It is assumed that all required neural network models have been loaded before starting this process. The basic steps involved in this process are:

- **Taking input image through GUI**

  For performing face mask detection, first the image on which this has to be performed must be taken as input. This is done by creating a GUI and allowing the user to browse through his/her machine to find the image. The GUI has been created using 'Tkinter' which is a GUI library present in the default python package.

- **Performing face detection**

  After getting the input image, first, face detection has to be performed on it. For this, first the input image has to be preprocessed in such a manner that it becomes suitable for the face detection model. So, the input image is resized to a size of 300 x 300 pixels and them it is converted into a BLOB image. BLOB stands for Binary Large OBject and refers to a group of connected pixels in a binary image. The preprocessed BLOB image is then forwarded to the OpenCV DNN for face detection which performs face detection on the input image and returns the locations and dimensions of all detected faces present in the image along with the confidence level for each detected face.

- **Performing face mask detection on detected faces**

  Finally, face mask detection has to be performed on the detected faces. In the previous step, the face detection DNN of OpenCV performs face detection on input image and returns a list of detected faces along with details such as location, boundaries and confidence level of each detected face. Next for each detected face, first the confidence level associated with it is checked. A detected face is considered to be valid only if the confidence associated with it is greater than a certain threshold value (0.5). If the detected face is valid, then, the image of the detected face is cropped out from the original image, preprocessing is performed on the extracted image and then, it is forwarded to the previously trained face mask detection model. This model predicts a probability for each of the two classes ('with_mask' and ‘without_mask’). The detected face belongs to the class which has higher probability. This final output is displayed properly using a GUI.

C. Face mask detection in videos and real time stream

The proposed face mask detection model can also be used to detect face masks in videos and real time streams. For this, first all necessary neural network models have to be loaded. Next, the real time video stream (or offline video) is played and a frame (or snapshot) of the video is captured after a fixed interval. Now, face mask detection is performed on this captured frame by following the same procedure which is used to perform face mask detection on an image. Thus, in case of real time video stream or offline video, the system just reads the video frame by frame, performs face mask detection on each of the frames and then displays an output for each frame. Currently, the implementation of the face mask detection system allows browsing video from the local machine/ computer as well as real time video stream through an integrated webcam.

3.2 EXPERIMENTAL RESULT

The proposed MobileNetV2 architecture based deep learning model for performing face mask detection, created using Keras framework of TensorFlow in python programming language, is trained using a dataset containing a total of 2414 images for 20 epochs using 4GB of GTX Geforce 1650ti as GPU for faster training. The following metrics have been used to evaluate the performance of the final trained model:

- **Accuracy:**
  Accuracy is ratio of correctly predicted observations to the total observations and it is given by the formula:

$$Accuracy = \frac{True\ Postitives + True\ Negatives}{Total\ number\ of\ predictions}$$

- **Precision:**
  Precision is the ratio of correctly predicted positive observations to the total predicted positive observations.

$$Precision = \frac{True\ Postitives}{True\ Postitives + False\ Postitives}$$

- **Recall:**
  Recall is the ratio of correctly predicted positive observations to the total number of observations which are actually positive.

$$Recall = \frac{True\ Postitives}{True\ Postitives + False\ Negatives}$$

- **F1 score:**
  F1 Score is the weighted average of Precision and Recall.

$$F1\ Score = \frac{2 * (Recall * Precision)}{Recall + Precision}$$

Figure 3.4 shows the classification report generated after analyzing the performance of the trained model. As is evident from the classification report, the model gives a training accuracy of 99% which is excellent. To be exact, the model achieves a training accuracy of 99.52% and a validation accuracy of 99% which shows that the model has trained well.

MODEL EVALUATION

	precision	recall	f1-score	support
with_mask	1.00	0.98	0.99	250
without_mask	0.98	1.00	0.99	250
accuracy			0.99	500
macro avg	0.99	0.99	0.99	500
weighted avg	0.99	0.99	0.99	500

Figure 3.4: Classification Report of trained model

Further, all other metrics such as precision, recall, F1 score etc. also have excellent values. Figure 3.5 shows the trend in the training and validation losses as the number of epochs increase. It can be observed that initially the training and validation loss are both quite high. But as the model is trained for more and more epochs, the training and validation losses reduce drastically and finally both the training and validation losses converge after a point. Similarly, Figure 3.6 shows the change in training and validation accuracies. Although initially both the accuracies are quite low, but, as the model is trained for more epochs, the accuracy improves drastically and finally converges after a certain point. This shows the model has been trained properly on the created dataset and can now be deployed for practical uses.

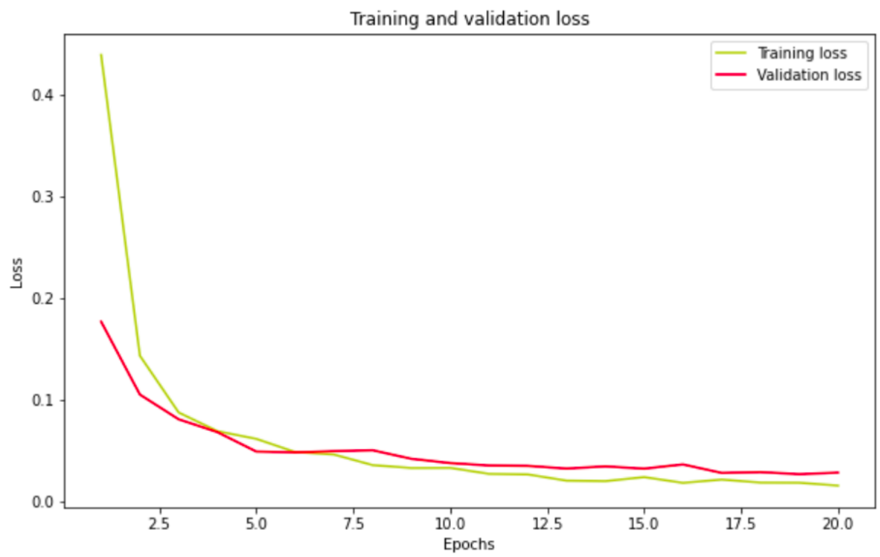

Figure 3.5: Trend in Training and Validation Loss

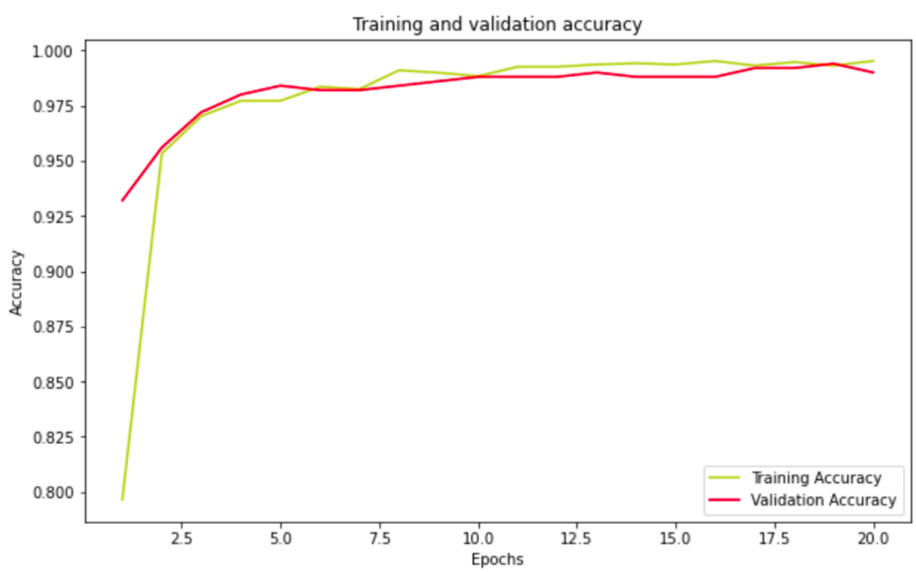

Figure 3.6: Trend in Training and Validation Accuracy

After evaluation of the trained model, the model is deployed for practical use. Face mask detection is performed on images, videos and real time video streams using the trained model to evaluate how the model performs on real data. Figure 3.7, 3.8 & 3.9 show the output after the face mask detection is performed on images, real-time video stream and local videos respectively. The results show that the model is performing well on real life data and it is predicting most results correctly.

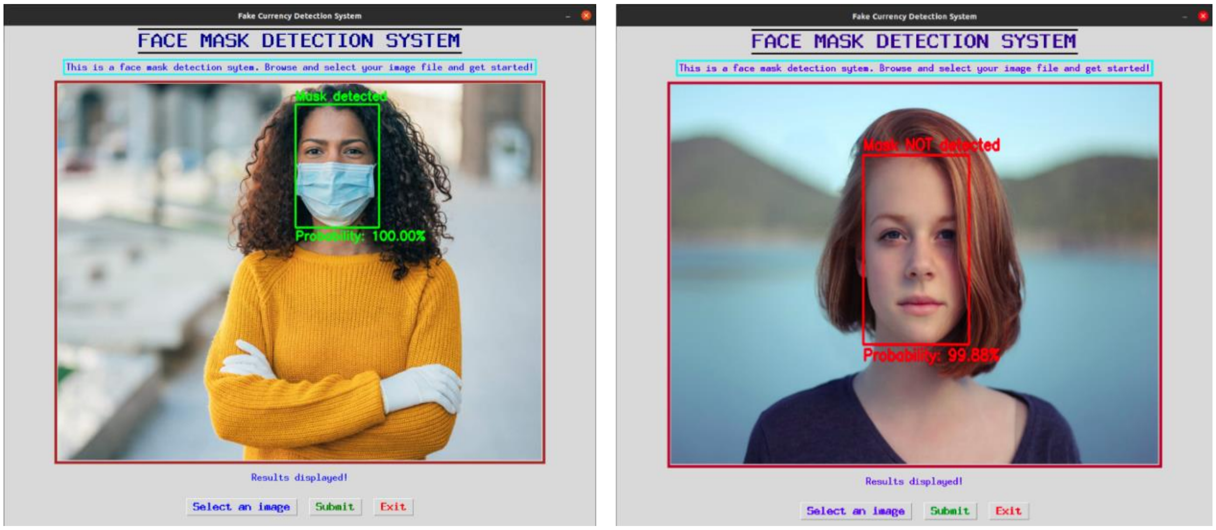

Figure 3.7: Output after face mask detection is performed on images

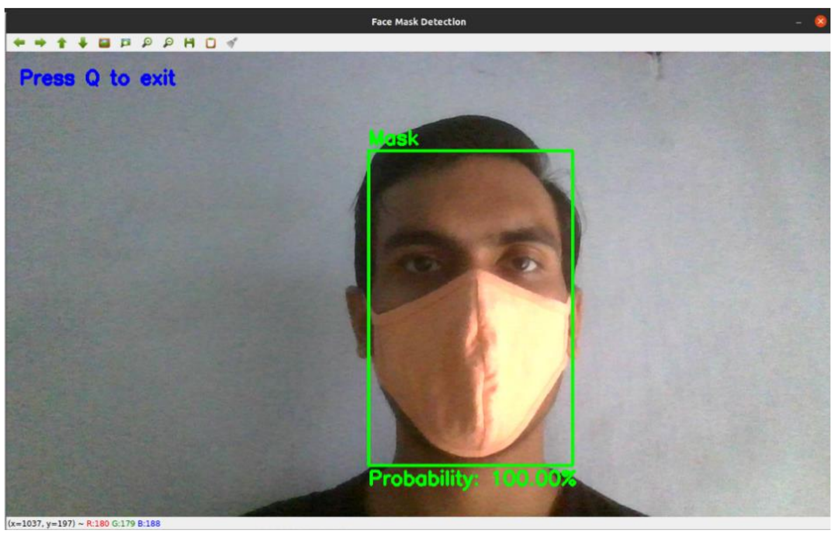

Figure 3.8: Output after face mask detection is performed on real-time video stream

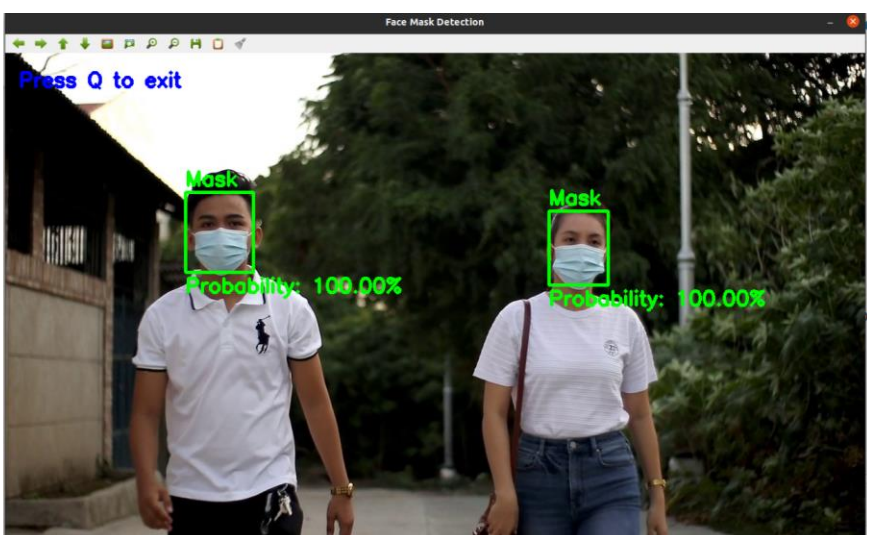

Figure 3.9: Output after face mask detection is performed on local video

From the above experiments, we can conclude that the final model has trained excellently and is producing great results. Not only does this face mask detection model predict results correctly, but also this model performs better and faster than many other face mask detection systems that is currently being used. The proposed model is created using the MobileNetV2 architecture and the biggest advantage of using this architecture is that the final model not only produces great performance and accuracy, but also, the final model is extremely lightweight. Due to this, this deep learning model can be deployed in even less powerful machines such as CCTV cameras and embedded devices. There are many other architectures which can be used to create similar systems, but none of the models can provide both good accuracy and high ease of deployment. MobileNetV2 architecture ensures that the model is both lightweight and gives good accuracy. Due to the proposed model being extremely light as compared to other similar models, this proposed model is also capable of doing real time face mask detection which is a big advantage over other similar models. Good performance, good accuracy, high ease of deployment, lesser training time and ability to perform real time face mask detection makes this proposed model a very good option as a face mask detection system.

Chapter 4 CONCLUSIONS AND FUTURE TRENDS

In this paper, a face mask detection system based on Deep learning and MobileNetV2 architecture has been proposed. The face mask detection model has been trained on a dataset containing 2114 images for 20 epochs and it has achieved a training accuracy of 99.52% and a validation accuracy of 99% which proves that the trained model is capable of performing face mask detection extremely well. Further all other metrics of the trained model such as accuracy, precision, recall and F1 score is also quite good. Upon deploying the model for practical use, it is observed that the system performs quite well and predicts correctly in most cases.

Further, the proposed face mask detection model is built using MobileNetV2 architecture and it also uses the inbuilt face detection DNN present in OpenCV to perform face detection. The DNN implementation for face detection by OpenCV ensures that the model is able to detect faces accurately and the MobileNetV2 architecture ensures that the model predicts the class of the detected faces correctly. The model trains faster as a result of the MobileNetV2 architecture and exploits all the benefits which are associated with Transfer learning. Due to this, the final model is lightweight and easy to deploy, but still, it displays good accuracy. Further, due to its lightweight nature, the model is able to perform real time face mask detection which is a huge advantage.

The face mask detection system can be further improved by improving the weights learned by the model by using even bigger dataset. The model can be trained on a bigger dataset containing images in various conditions such as blurry images, images under various orientations and lighting conditions etc. to improve the model. This face mask detection system can be used in many situations in future. Use of face mask detection systems in public places, hospitals, businesses, shopping complexes etc. can help to reduce the transmission of a variety of highly contagious diseases such as COVID 19. Use of such systems in hazardous environments such as chemical industries, fertilizer and pesticide industries, mining fields can also help to reduce the accidental loss of lives due to absence of proper protective gear. Integrating the model with CCTV cameras can also help various departments to enforce stricter rules and regulations regarding wearing masks in public places during a pandemic. Thus, the proposed face mask detection system can help a lot by reducing loss of lives and other important resources.

REFERENCES

[1] Mohamed Loey, Gunasekaran Manogaran, Mohamed Hamed N. Taha, Nour Eldeen M. Khalifa (2021). "Fighting against COVID-19: A novel deep learning model based on YOLO-v2 with ResNet-50 for medical face mask detection". Sustainable Cities and Society, Volume 65, 102600, ISSN 2210-6707,

[2] Qin, Bosheng & Li, Dongxiao. (2020). "Identifying Facemask-Wearing Condition Using Image Super-Resolution with Classification Network to Prevent COVID-19". Sensors (Basel, Switzerland). 20. 10.3390/s20185236.

[3] Inamdar, Madhura & Mehendale, Ninad. (2020). Real-Time Face Mask Identification Using Facemasknet Deep Learning Network. SSRN Electronic Journal. 10.2139/ssrn.3663305.

[4] Jignesh Chowdary G., Punn N.S., Sonbhadra S.K., Agarwal S. (2020) Face Mask Detection Using Transfer Learning of InceptionV3. In: Bellatreche L., Goyal V., Fujita H., Mondal A., Reddy P.K. (eds) Big Data Analytics. BDA 2020. Lecture Notes in Computer Science, vol 12581. Springer, Cham. https://doi.org/10.1007/978-3-030-66665-1_6