Security with CCTV Camera and Deep Learning with Python

Durgesh Rao, Department of Computer Science, LNCT University

Kolar Road, Bhopal, Madhya Pradesh M.P

Abstract

The Corona Virus Disease known as COVID-19; An Acute Respiratory Syndrome Coronavirus-2 Disease called SARS-CoV-2. As, this virus is genetically related to the SARS outbreak of 2003. This deadly and unstoppable disease, which all started in Wuhan, China in Dec 2019, was declared as the World Pandemic on 11 March, 2020 by WHO World Health Organization.

Face Mask and Social Distancing are the Best Precautions for staying safe. These Research Papers focuses on implementing Real Time Face Mask Detection using CCTV Closed-Circuit Television Camera and Deep Learning, to Catch the Real Culprit of Virus Spread and Report him or her Immediately.

Keywords

Face Mask Detection, CCTV Closed-Circuit Camera, Machine Learning, Scikit-learn Library, COVID-19, CNN Convolutional Neural Network and OpenCV Opensource Computer Vision Library.

Introduction

With the recent spread of covid-19, individuals with coronavirus lack symptoms known as [Asymptomatic], spreads the disease to other people unknowingly and people who develop symptoms called [Symptomatic]. Below are the ways the virus can spread in the crowded people are as follows: Close contact with people, like sneezing, coughing, handshake, etc.

Especially, the heavy crowded public places like, Offices, Corporate's and Practical's in Colleges, where Manual presence is necessary, increases the risk of virus spread. To deal with such a Problem, Technology like AI Artificial Intelligence plays a major role.

Many international Scientists and Organizations have stated that, The Birth of even more dangerous viruses than covid may takes place with the emerging Biotechnology and Science in the next 20 years. As per the future needs, the technology is Scalable and Reliable for the Big Industries dealing with effective solution.

The Third wave of Covid-19 is still on the move, The people are becoming more and more careless and negligible in wearing mask and taking precautions. As, it is not possible all the time to keep an eye on each and every person. Statistics, says that more than 10,00,000+ lives have lost till now, by just not wearing mask. The new "Mutant" of coronavirus which is even more dangerous, as it cannot be even detected in RT-PCR Test.

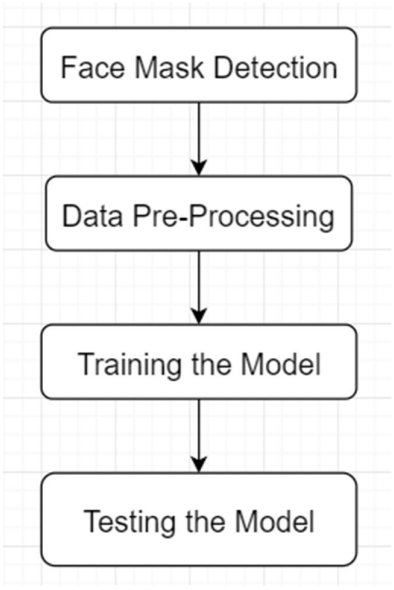

Figure 1. Overview of Complete Model

A Face Mask Detection Deep Learning Model, compatible with CCTV Camera Software detects the people not wearing mask, wearing torn mask and wearing mask under nose. The camera then clicks the photo of the individual and sends it to the Reception, Where the details of the individual will be fetched and will restrict his entry into the organization Campus.

Literature Review

Face Mask Detector: The device is very useful in object detection, especially used in public places like airports, heavy traffic places and in shopping malls to monitor the crowd, whether taking precautions from virus spread or not. The device uses Google Collab for data pre-processing, hence takes a lot of time while preparing the data. It also doesn't allow the webcam or portable camera to connect with, so cannot be tested in real-time basis.

SSDMNV2: The model uses OpenCV DNN Deep Neural Network, along with MobileNetV2 for mobile classifier, which provides light weight and accurate predictions. The device also provides better results with frontal face with mask and frontal face without mask, Keras and Tensorflow are also been used along with SSD & ResNet-10 for classifying images in different angles and orientations with different colors.

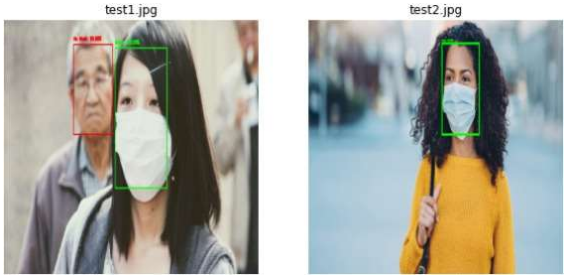

Figure 2. SSDMNV2 working Model

Research Methods

The following are the Tools and Technologies Used in this research thesis explained with their use and importance in the future:

Python3: Python is a popular programming language, created by Guido Van Rossum and released in 1991. Python is mostly preferred over many other programming languages like C++, Java, etc. especially for Machine Learning and Artificial Intelligence, because of its easy and simple syntax, makes it beginner friendly. It works on different platforms like Raspberry Pi, Mac, Linux, etc. It is mainly used for System Scripting. The latest version used in this thesis is python3.

Jupyter Notebook: It is an open-source web application that allows machine learning engineers and data scientists to write and edit python scripts, perform scientific computing and create visualizations effectively.

The Main Idea of implementing this technology is broadly divided into three main categories are: Data Pre-Processing, Training the Model and Real-Time Face Detecting.

Data Pre-Processing: This is one of the most important steps before building any model form basic to advance. Data collected from various resources contains one or more faults like missing values, outliers, unnecessary data, etc. which needs to be removed and filter out before further process. As, we perform three important steps inside data Pre-Processing, like: Data Cleansing, Data Transformation and Data Reduction. But, in this case we are using images, so we will be using Image Classification for this process.

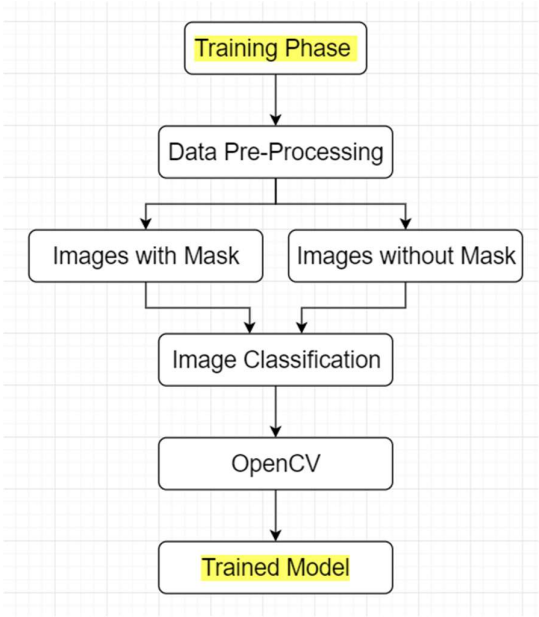

Figure 3. Overview of Training Phase

Firstly, we will store Images of people with mask in one folder and images of people without mask in another folder. Then using Dictionary concept of keys and values, we will be labelling the images numerically. Hence, approximately 670+ images of each kind with different angle and orientation have been taken to train the model. Finally, we convert the images size into one fixed 100 x 100 px for similarity, also convert the color of all images from BGR [Blue Green Red] format to Greyscale [Black & White] Method for the same similarity.

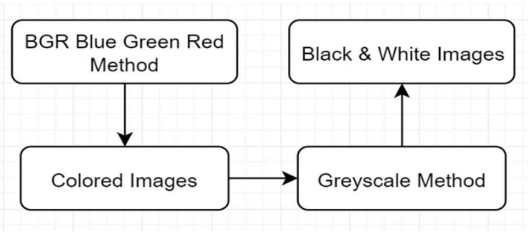

Figure 4. BGR to Greyscale with 100px resize

Training the Model: Now, we train the model using CNN Convolutional Neural Network as follows: Firstly, we upload the prepared dataset, then define the CNN, which extracts information from pixels [smallest unit of images], Then comes defining the Architecture of CNN, which includes Convolutional Layer: which applies number of filters to the feature map, after this we need a relay activation function to add non-linearity to the network.

The Convd2d(), constructs a two-dimensional convolutional layer with various other parameters. Max_pooling2d(), Constructs a two-dimensional pooling layer using max-pooling algorithm. Dense(), Constructs a dense layer with hidden layers and units.

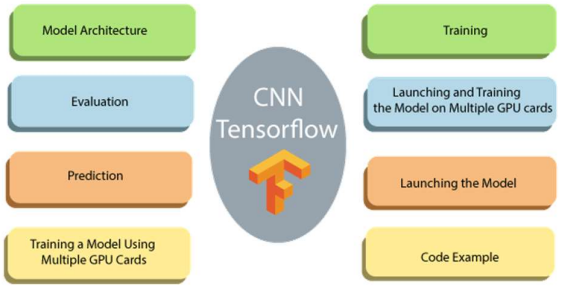

Figure 5. CNN Architecture

Then Pooling Layer, where the mobility of feature map and overfitting of the model is reduced and prevented. Max Pooling splits feature map into subfields and only holds maximum values. Fully Connected Layers, where All neurons from the past layers are associated with other next layers. CNN has classified from convolutional layers and reduced with any pooling layer. The first and second layer in CNN is pooling layer followed by relu. Flatten Layer is used to stack the output convolutions from second convolution layer. Then we added a dense layer of about 64 neurons. The final layer is thus ended with two outputs for two categories.

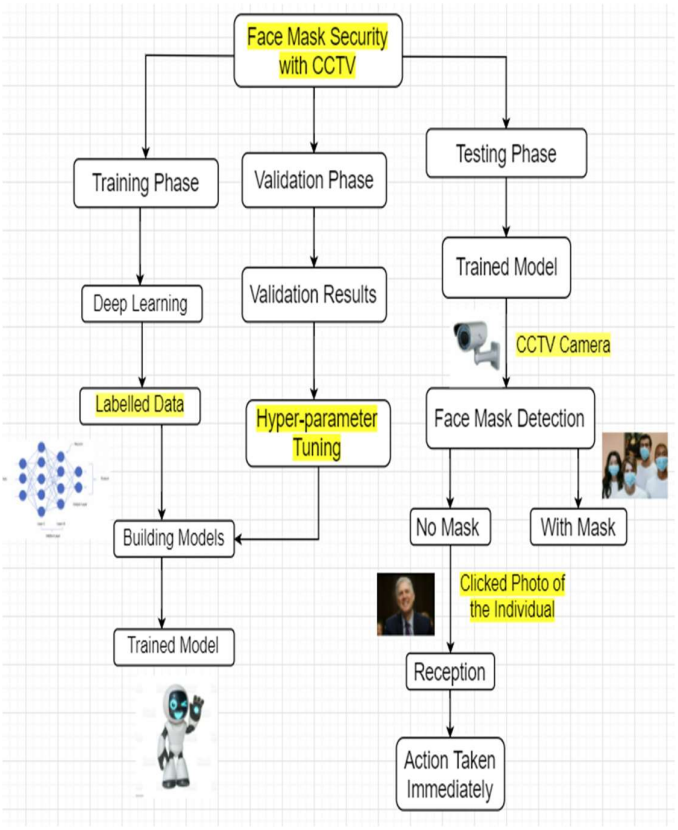

Figure 6. Complete Overview of Model

Testing the Model: While training we used two types of images one with mask [positive images] and other without mask [negative images]. Hence, a cascade function is trained from a lot of positive and negative images. We have used Haarcarcade Algorithm, which is an object detection algorithm used to identify images in real time webcam or video. After this, we use webcam as the scanner for real time face mask detection. This Model is then made compatible with CCTV Cameras, for monitoring in heavy crowded public places and catch people, who are neglecting the importance of mask.

Figure 7. Real Time Working Model

Result Discussions

The model was trained with 20 epochs, and the results are accurate and good.

The below is the graph of training loss and validation loss verses epochs. We can see that that the training and validation loss decreases with the increase in number of epochs, and training loss reduced from 0.7 to almost null after the number of epochs crosses 17.5 and validation loss reduced from 0.7 to 0.2.

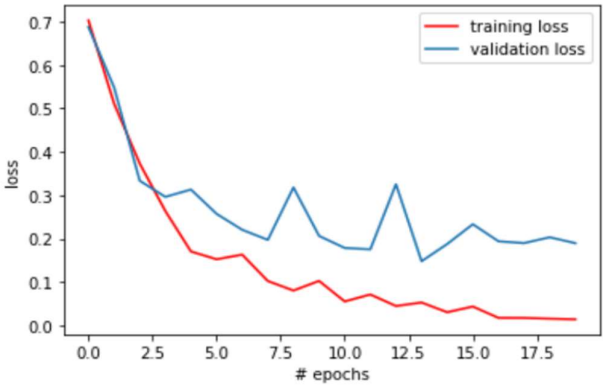

Figure 8. Training & validation loss verses number of epochs

Another graph is of training accuracy and validation accuracy verses epochs. And we clearly see that training and validation accuracy increases with the increase in number of epochs, as a result of which training accuracy gives 1.0 followed by validation accuracy of 0.9.

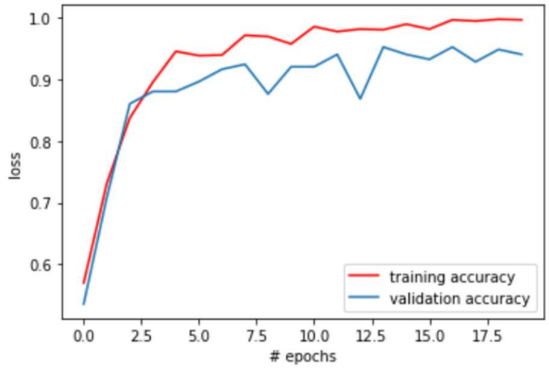

Figure 9. Training & validation accuracy verses number of epochs

Therefore, The Final Results of the model are:

Training Accuracy → 0.99

Validation Accuracy → 0.94

Training Loss → 0.01

Validation Loss → 0.18

Algorithm	Accuracy (%)	Loss (%)
SSDMNV2	0.90	0.19
Haarcascade	0.99	0.01

Figure 10. Result Summary and Comparison

Result Implications

The model has faced a problem, here we have to train the model of same image with different angles and orientations, which makes it bit time consuming. Otherwise, it is the most effective solution using Deep Learning and AI.

Figure 11. Complications of Face Mask

We have to collect data from different sources of different types, like there are various 100 of colors of face masks, apart from that the most important is the angle of wearing mask, nose and mouth should be completely covered. And the different styles of face masks, makes the data collection complicated, otherwise Fine.

Conclusion

Due to the increasing outbreak of COVID-19, this technology of automated face mask detection has been on a trend. Many researchers have chosen this topic for their studies and many solutions has proved efficiently, including the one of CNN Convolutional Neural Networks, in which Data set was collected in the form of images from Kaggle, Google and other Platforms,

after that Conversion of images into proper format took place by resizing them and changing color to black and white format. Then using Cascade Algorithm to classify images and train the model using image classification and OpenCV. The trained model is then used with CCTV Camera to detect crowd and filter out people not wearing mask. This model can be hence used in various public places like shopping malls, airports, etc.

References

[1] Wang, B.; Zhao, Y.; Chen, C.P. Hybrid Transfer Learning and Broad Learning System for Wearing Mask Detection in the COVID-19 Era. IEEE Trans. Instrum. Meas. 2021.

[2] Rudraraju, S.R.; Suryadevara, N.K.; Negi, A. Face Mask Detection at the Fog Computing Gateway. In Proceedings of the 2020 IEEE 15th Conference on Computer Science and Information Systems (FedCSIS), Sofia, Bulgaria, 6–9 September 2020.

[3] I. B. Venkateswarlu, J. Kakarla and S. Prakash, "Face mask detection using Mobile Net and Global Pooling Block," 2020 IEEE 4th Conference on Information & Communication Technology (CICT).

[4] M. S. Ejaz and M. R. Islam, "Masked Face Recognition Using Convolutional Neural Network," 2019 International Conference on Sustainable Technologies for Industry 4.0 (STI)

[5] A. G. Howard, M. Zhu, B. Chen et al., “Mobile-nets: efficient convolutional neural networks for mobile vision applications,” 2017