**MASK DETECTION USING MACHINE LEARNING**

**A Project Report submitted in partial fulfillment of the requirements for**

**the award of the degree of**

**BACHELOR OF TECHNOLOGY**

**IN**

**COMPUTER SCIENCE AND ENGINEERING**

**Submitted by**

**Vegu Mahesh Babu 121710318057**

**V.S.B. Rao 121710318055**

**Patibandla Dharmesh 121710318039**

**Gondrala Dinesh 121710318013**

**Under the esteemed guidance of**

**Mrs Manasa Vavilapalli**

**Assistant Professor,GITAM (CSE)**



**DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING**

**GITAM**

**(Deemed to be University)**

**VISAKHAPATNAM**

**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING**

**GITAM INSTITUTE OF TECHNOLOGY**

**GITAM**

**(Deemed to be University)**

****

**DECLARATION**

We, hereby declare that the Project review entitled “**MASK DETECTION USING MACHINE LEARNING**” is an original work done in the Department of Computer Science and Engineering, GITAM Institute of Technology, GITAM (Deemed to be University) submitted in partial fulfilment of the requirements for the award of the degree of B.Tech. in Computer Science and Engineering. The work has not been submitted to any other college or University for the award of any degree or diploma.

**Date**: 12 December 2020.

**Name: Registration No:**

Vegu Mahesh Babu 121710318057

V.S.B. Rao 121710318055

Patibandla Dharmesh 121710318039

Gondrala Dinesh 121710318013

**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING**

**GITAM INSTITUTE OF TECHNOLOGY**

**GITAM**

**(Deemed to be University)**

****

**BONAFIDE CERTIFICATE**

This is to certify that the project report entitled “**MASK DETECTION USING MACHINE LEARNING**” is a bonafide record of work carried out by **Vegu Mahesh Babu 121710318057 ,V.S.B. Rao 121710318055, Patibandla Dharmesh 121710318039, Gondrala Dinesh 121710318013**, submitted in partial fulfilment of requirement for the award of degree of Bachelors of Technology in Computer Science and Engineering.

**PROJECT GUIDE** **HEAD OF THE DEPARTMENT**

**Mrs. Manasa Vavilapalli**  **Dr.K.Thammi Reddy**

**(Assistant Professor) (Professor)**

**CSE, GIT CSE, GIT**

**GITAM GITAM**

**ACKNOWLEDGEMENT**

We would like to thank our project guide **Mrs. Manasa Vavilapalli**, Assistant professor, Department of CSE for his stimulating guidance and profuse assistance. We shall always cherish our association with him for his guidance, encouragement and valuable suggestions throughout the progress of this work. We consider it a great privilege to work under his guidance and constant support.

We also express our thanks to the project reviewers **Venkata Ramana M, Assistant Professor, and Appa Rao G, Assistant Professor, Department of CSE, GITAM (Deemed to be University)** for their valuable suggestions and guidance for doing our project.

We consider it a privilege to express our deepest gratitude to **Dr. K. Thammi Reddy**, Head of the Department, Computer Science Engineering for his valuable suggestions and constant motivation that greatly helped us to successfully complete this project.

Our sincere thanks to **Dr. C. Dharma Raj**, Principal, GITAM Institute of Technology, GITAM (Deemed to be University) for inspiring us to learn new technologies and tools.

Finally, we deem it a great pleasure to thank one and all that helped us directly and indirectly throughout this project.

Vegu Mahesh Babu 121710318057

V.S.B. Rao 121710318055

Patibandla Dharmesh 121710318039

Gondrala Dinesh 121710318013

**TABLE OF CONTENTS**

|  |  |  |
| --- | --- | --- |
|  | Declaration | 2 |
|  | Certificate | 3 |
|  | Acknowledgements | 4 |
| 1. | Abstract | 6 |
| 2. | Introduction | 7 |
| 3. | Problem Identification And Objectives  3.1 Existing Problem  3.2 proposed Solution  3.3 Technologies/Tools Used | 8-10 |
| 4. | System Architecture And Block Diagram  4.1 Convolution Neural Networks  4.2 Layers Of Cnn  4.3 Methodology  4.4 Block Diagram For two phase Covid-19 Face Mask Detector | 11-15 |
| 5. | Dataset Specifications | 16 |
| 6. | Software Specifications | 17 |
| 7. | Programming Language | 18 |
|  |  |  |
| 8. | Implementations | 19-22 |
| 9. | Results | 23 |
| 10. | Conclusion | 24 |

11. References 25

**1.ABSTRACT**

Nowadays, Machine learning is developing very fastly. Almost in every sector machine learning is used. And it is an application of artificial intelligence where available information is used through algorithms to process or assist the processing of statistical data. While Machine Learning involves concepts of automation, it requires human guidance. Our system consists of a dual-stage Convolutional Neural Network (CNN) architecture capable of detecting masked and unmasked faces and can be integrated with pre-installed CCTV cameras. This will help track safety violations, promote the use of face masks, and ensure a safe working environment.

The proposed project is on Mask detection. Here we used Convolutional neural networks(CNN) from deep learning and used MobileNetv2 architecture. And we used some modules like Tensorflow, Keras, Openc

**2.INTRODUCTION**

Due to COVID-19 coronavirus epidemic, wearing face masks in public has increased all over the world. Before COVID-19 people used to wear masks to protect themselves from air pollution. Scientists have also proved that wearing face masks works on impeding COVID-19 transmission. COVID19 is the latest epidemic virus that hit human health in the last century. The rapid increase/spreading of CoronaVirus in 2020 has forced the World Health Organization (WHO) to declare COVID-19 as a global pandemic.

People are concerned about their health, and public health is considered as the top priority for governments. The World Health Organization (WHO) also recommends that people should wear face masks if they have respiratory symptoms, or they are taking care of people with symptoms.

With the rapid development of machine learning and deep learning methods, the problem of face detection seems to be well addressed yet. Artificial Intelligence (AI) based on Machine learning and Deep Learning can help to fight Covid-19 in many ways. Technologies in fields like machine learning and deep learning have made our lives easier and provide many solutions to several complex problems in various fields/areas. Face mask detection has become a crucial computer vision task to help the global society.

Face mask detection refers to detecting whether a person is wearing a mask or not. The provision of healthcare needs funding for emerging technology such as artificial intelligence, IoT, big data and machine learning to tackle and predict new diseases. In order to better understand infection rates and to trace and quickly detect infections.

Now let us understand the meaning of each word in detail in the term “FACE MASK DETECTION”

Firstly, here we discuss the meaning of face mask, different types of face masks present in the market and their uses.

“FACE MASK” is a covering made of polypropylene fiber or cotton which is used to cover the mouth and nose. It is generally used by the people to prevent the spread of various infectious viruses and bacteria from one-person to the other.

**2.LITERATURE REVIEW**

**2.1. Introduction**

For identifying whether a person is wearing a face mask or not, we develop an efficient system here. The mechanism of identifying masked and unmasked faces leads to input data collection from various resources. In this section of the paper, we discuss certain research papers, which presented different studies on face recognition and face mask recognition and techniques used by the researchers in achieving their goal.

**2.2. Related work**

**1. Md. Sabbir Ejaz and Md. Rabiul Islam (2019)** proposes Multi-Task Cascaded Convolutional Neural Network (MTCNN) for Face Mask Detection. But due to the presence of various face masks present, there is a problem in detecting the masked person. In this model he first detects facial recognitions. The occluded face detection problem has been approached using Multi-Task Cascaded Convolutional Neural Network (MTCNN). Then facial features extraction is performed using the Google Face Net embedding model. And finally, the classification task has been performed by Support Vector Machine (SVM). The primary concern to this work is about facial masks, and especially to enhance the recognition accuracy of different masked faces.

**2. Chengbin Peng (2017)** proposes a new CNN-based cascade framework, which consists of three carefully designed convolutional neural networks to detect masked faces. It is a CNN based cascade structure face detector. The proposed masked face detection algorithm achieves satisfactory performance.

**3. Gayatri Deore, Ramakrishna Bodhula, Dr. Vishwas Udpikar, Prof. Vidya More (2016)** proposesthe method of Histogram of Oriented Gradients. Histogram of Gradients (HOG) is a feature set based on evaluating well-normalized local histograms of image gradient orientations in a dense grid. It gives good results for person detection, reducing false positive rates relative to the best Haar wavelet-based detector. The aim is to detect if a person is wearing a mask or not.

**3.PROBLEM IDENTIFICATION AND OBJECTIVES**

**3.1 EXISTING PROBLEM:**

Considering the current situation, there is a huge breakout of the COVID-19 virus all around the globe. This virus has affected millions of lives around the world. Many precautionary measures have been taken to control the spread of virus from one person to the other. Some of them include wearing face masks while travelling, maintaining social distancing, washing hands at regular intervals of time, sanitizing. One of the important measures is to wear a face mask while travelling to near-by places like grocery stores, vegetable markets, shopping malls.

Right now, there are no efficient face mask detection applications which are very much needed for densely populated areas, residential districts, large scale manufacturers to ensure safety. Also, due to the absence of huge datasets, this made this task more cumbersome and challenging. So, one of the best ways in this pandemic situation is to implement an efficient face mask detection system to ensure safety and reduce the anxiety and fear among people.

**3.2 PROPOSED SOLUTION:**

Deep Learning is a subset of Artificial Intelligence that mimics the working of the human brain in processing data. It is used for detecting objects, recognizing speech, translating languages and making decisions.

In this project we use Convolution Neural Networks (CNN), which is a popular algorithm used for image classification and Object Detection. Our system consists of a Dual Stage Convolution Neural Networks which is used for detecting the differentiation between masked and unmasked faces. This system is then integrated with pre-installed CCTV cameras. This system therefore can be installed at public places like schools, colleges, Airports, Railway Stations, Shopping malls. This ensures the safety of the people and thus stops the transmission of virus from one person to the other to a great extent.

**3.3 TECHNOLOGIES / TOOLS USED:**

**1.SPYDER IDE:** Spyder is an open-source cross platform Integrated Development Environment for scientific programming in Python language.

**2.TENSORFLOW:** TensorFlow is an open-source library for numerical computation and large-scale Machine Learning

**3.KERAS:** Keras is an open-source library that provides a Python interface for artificial neural networks. Keras acts as an interface for the TensorFlow library.

**4.FLASK:** Flask is a web application framework written in Python. It is developed by Armin Ronacher.

**5.ANACONDA NAVIGATOR:** It is used for Application Launching and it manages all the packages and provides a Graphical user Interface (GUI).

**4.SYSTEM ARCHITECTURE AND BLOCK DIAGRAM**

In this project we have used Convolution Neural Networks Algorithm to detect whether a person is wearing a mask or not. Convolution Neural Networks is one of the popular algorithms for Image Classification and Object Detection. As our project “FACE MASK DETECTION” deals with image classification and Object Detection CNN is the best suited algorithm which accurately predicts whether a person is wearing a mask or not. Here we have also imported various python libraries like TensorFlow, sklearn, Keras for implementing this model.

**4.1 CONVOLUTION NEURAL NETWORKS:**

A **Convolutional Neural Network is** a Deep Learning algorithm which can take in an input image, assign importance (learnable weights and biases) to various aspects/objects in the image and be able to differentiate one from the other.

**4.2 LAYERS OF CNN:(Convolution Neural Networks)**

CNN contains 4 layers

1.Convolution Layer

2.Max Pooling Layer

3.Flatten Layer

4.Fully Connected Layer

**1.Convolution Layer:**

Convolution Layer is the first layer to extract features from an input image. It preserves the relationship between pixels by learning image features by using small squares of input data.

**2.Max Pooling Layer:**

This layer is used to perform an input sample by decreasing its dimensionality. It generally reduces the number of parameters when the images are too large. Therefore, the most extrusive features of the preceding feature of the map will be the output after the max pooling layer. This layer generally takes a 2\*2 matrix and selects the block with the largest value and gives it to the output.

**3.Flatten Layer:**

The output of the pooled layer is given as an input to the flatten layer. This layer is mainly used to convert an n-dimensional array/matrix into 1-D Array/matrix

**4.Fully Connected Layers:**

It is a very important layer and it is the last layer of CNN. It generally takes the inputs from the feature analysis and applies weights to predict the correct label. It gives the final probabilities for each label.

**4.3 METHODOLOGY:**

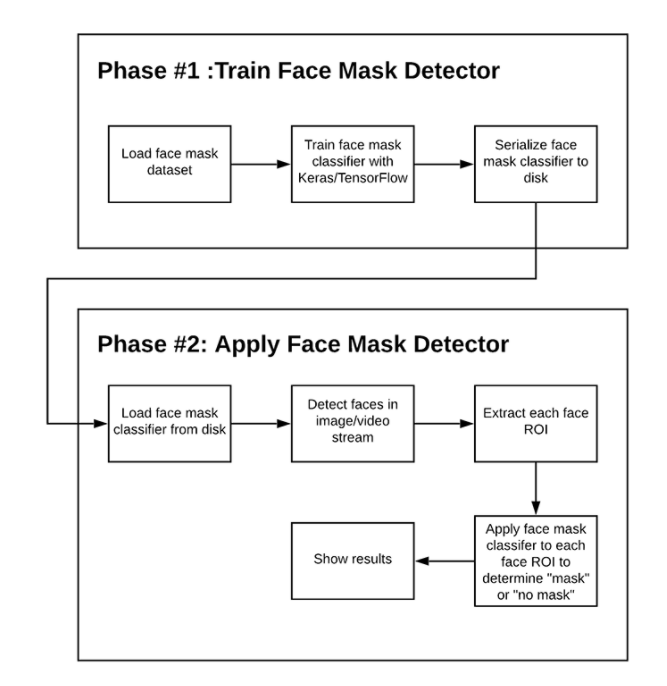
Convolution neural network algorithm is a multilayer perceptron that is the special design for identification of two-dimensional image information. In CNN we can give the input images of different kinds like Binary Images (Black/White Images), Gray Scale Images (It ranges between black and white images), Colored Images(Red,Green,Blue).

Firstly, the datasets are downloaded from Kaggle. The dataset consists of test and train images. Each test and train images have two categories which are “with mask” and “without mask”. In total the dataset has about 500-700 images.

Secondly, we used a Dual Stage Convolution Neural Network Algorithm. The picture below depicts the flow chart of Dual stage CNN.

**4.4 BLOCK DIAGRAM FOR TWO-PHASE COVID-19 FACE MASK DETECTOR:**

**4.4 BLOCK DIAGRAM FOR TWO-PHASE COVID-19 FACE MASK DETECTOR:**



**PHASE-1: Train Face Mask Detector:**

This implementation is present in train\_mask\_detector.py file. Here first we have to explore datasets from different websites and choose and download the required dataset. Import all the required libraries and packages. Next, we have to train the face mask classifier using Keras and Tensorflow packages. Now we serialize all the images into the required format and load it into our disk memory.

**PHASE-2: Apply Face Mask Detector:**

This implementation is present in detect\_mask\_video.py file. First, we have to load face mask classifier from the disk. It contains a method detect\_and\_predict\_mask which predicts whether the face is masked or unmasked. Next, we implement a real-time video streaming where a person’s face can be detected using camera or images. Finally, it will detect the image and predict the results as the person with mask and without mask accurately.

**5. DATASET SPECIFICATIONS**

The dataset is downloaded from Kaggle

https://www.kaggle.com/ashishjangra27/face-mask-12k-images-dataset

**Type of Dataset:**

It is an image dataset consisting of various masked and unmasked faces.

**Attributes:**

There are 2 attributes namely masked and unmasked images.

**Size of Dataset:**

It consists of about 3770 images divided into 2 categories.

With mask – 1884 images

Without mask – 1886 images

**6.SOFTWARE SPECIFICATIONS**

**Software:**

* TensorFlow
* Flask
* Spyder IDE
* Anaconda Navigator

**Anaconda Navigator:**

Generally, Anaconda Navigator contains many packages. It is used for starting applications. It has a command line interface. Navigators can look for packages on Anaconda cloud. It can be used in any kind of Operating System like Windows, Linux and Mac.There are various versions of the Navigator. Fixed Versions of Navigator depend upon the scientific ones. To differentiate between these, scientists use different versions of packages. It is a project manager as well as an environment planner. It helps in understanding various packages and their dependencies.

**TensorFlow:**

TensorFlow is an open-source library for numerical computation and large-scale Machine Learning.It is a package used widely for building Neural Networks. It was designed for both research and development purposes.

**Flask:**

Flask is a web application framework written in Python. It is developed by Armin Ronacher.

The flask package is installed from Python Package Index (PPI).

**Spyder Ide**:

Spyder is an open-source cross platform Integrated Development Environment for scientific programming in Python language. It integrates a number of packages like NumPy, SciPy, Matplotlib.

**7. PROGRAMMING LANGUAGE**

As a part of developing the project we need a Computer Programming language which is very efficient and easy to understand and write the code. We used **Python Programming language for developing Face Mask Detection Project**.

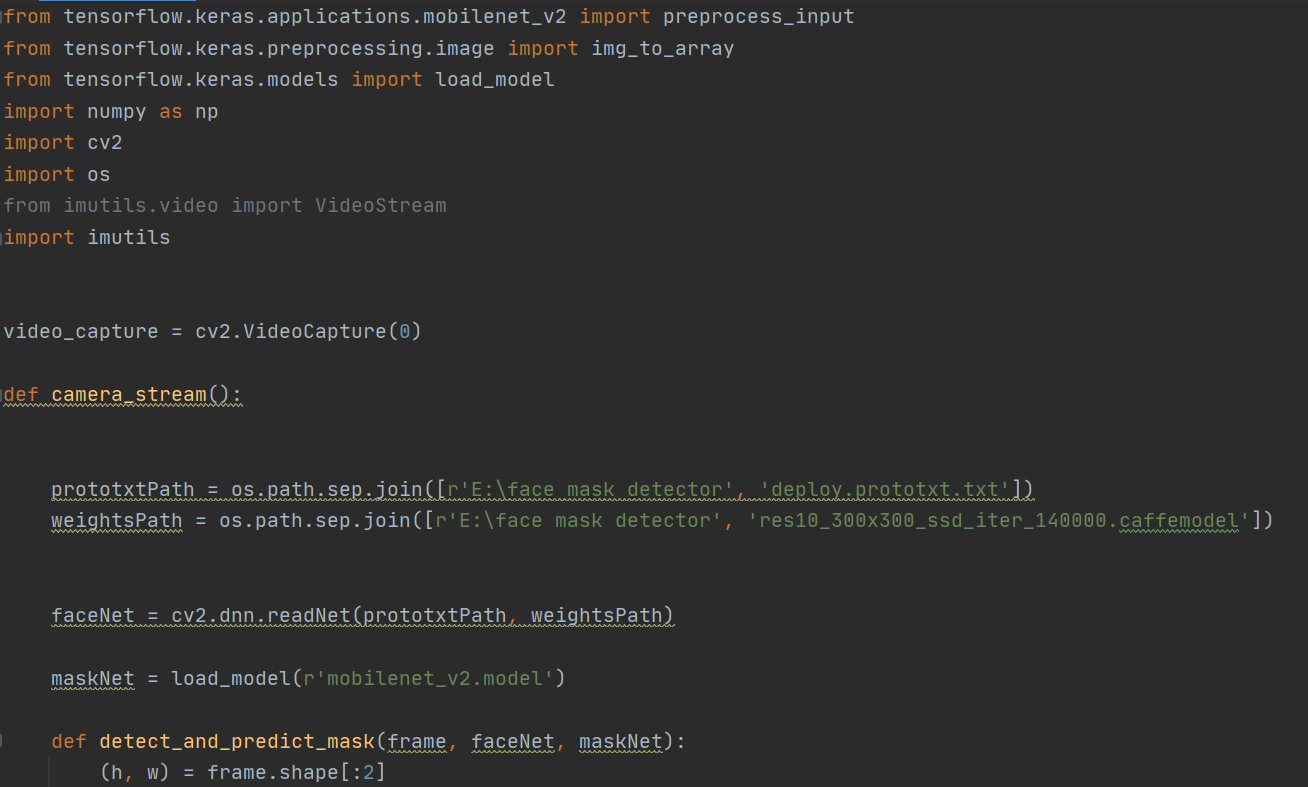
Python is an interpreted, object-oriented, high-level programming language with dynamic semantics. Its high-level built-in data structures, combined with dynamic typing and dynamic binding, make it very attractive for Rapid Application Development, as well as for use as a scripting or glue language to connect existing components together. Python supports modules and packages, which encourages program modularity and code reuse. The Python interpreter and the extensive standard library are available in source or binary form without charge for all major platforms, and can be freely distributed.

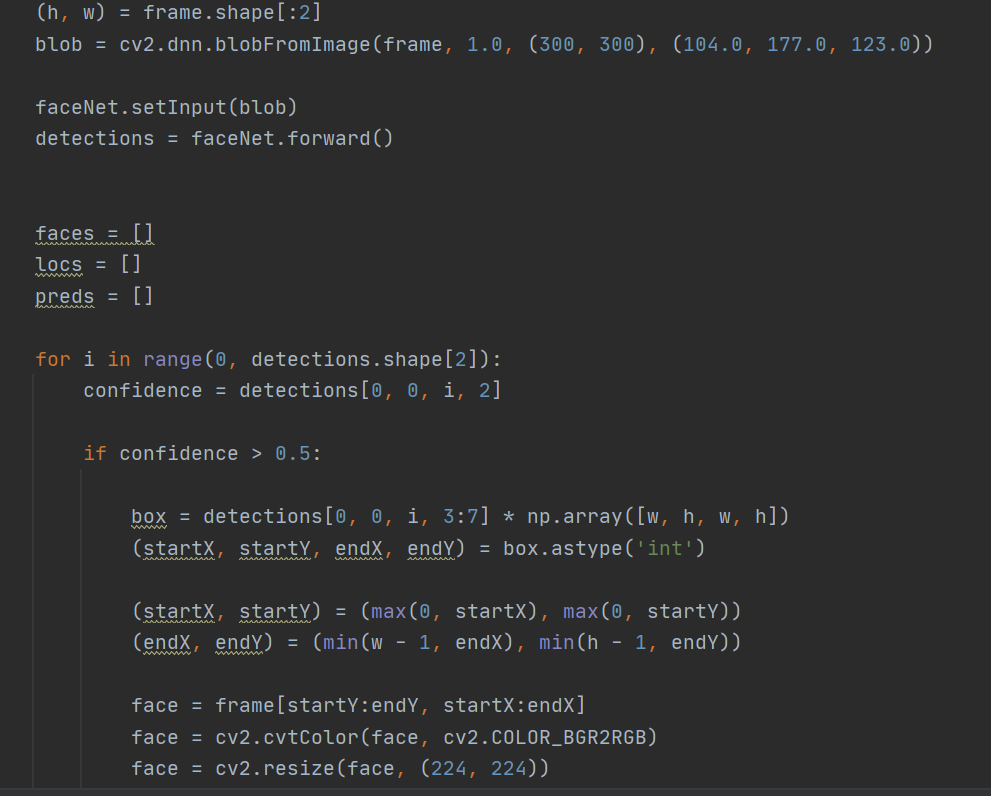
Often, programmers fall in love with Python because of the increased productivity it provides. Since there is no compilation step, the edit-test-debug cycle is incredibly fast. Debugging Python programs is easy, a bug or bad input will never cause a segmentation fault. Instead, when the interpreter discovers an error, it raises an exception. When the program doesn't catch the exception, the interpreter prints a stack trace. A source level debugger allows inspection of local and global variables, evaluation of arbitrary expressions, setting breakpoints, stepping through the code a line at a time, and so on. The debugger is also written in Python itself. On the other hand, often the quickest way to debug a program is to add a few print statements to the source, the fast edit-test-debug cycle makes this simple approach very effective.

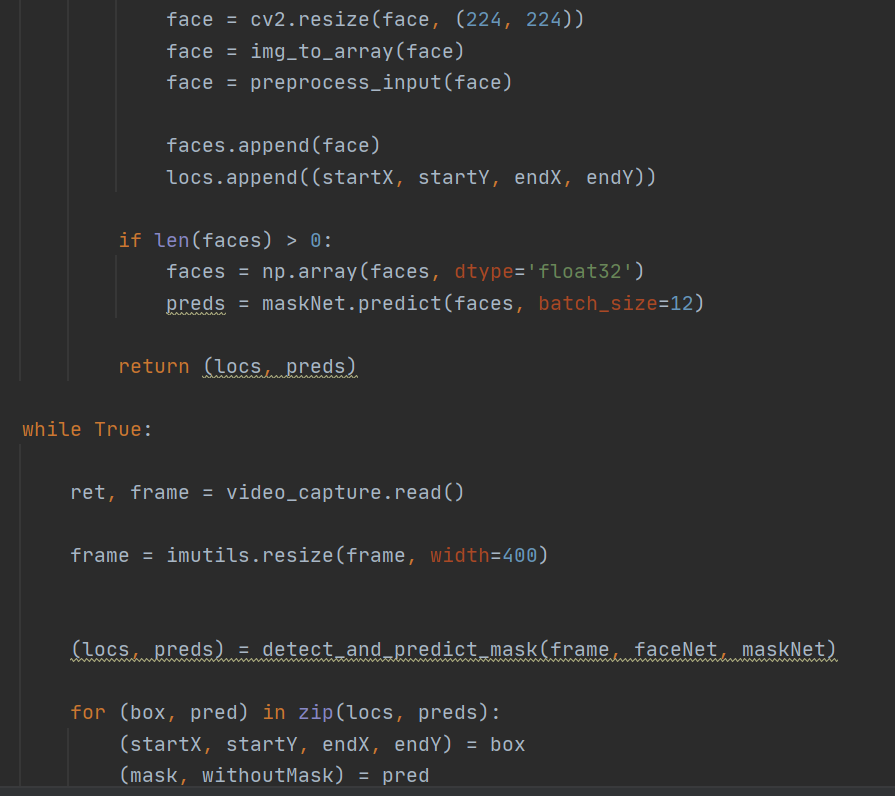
**8. IMPLEMENTATION**

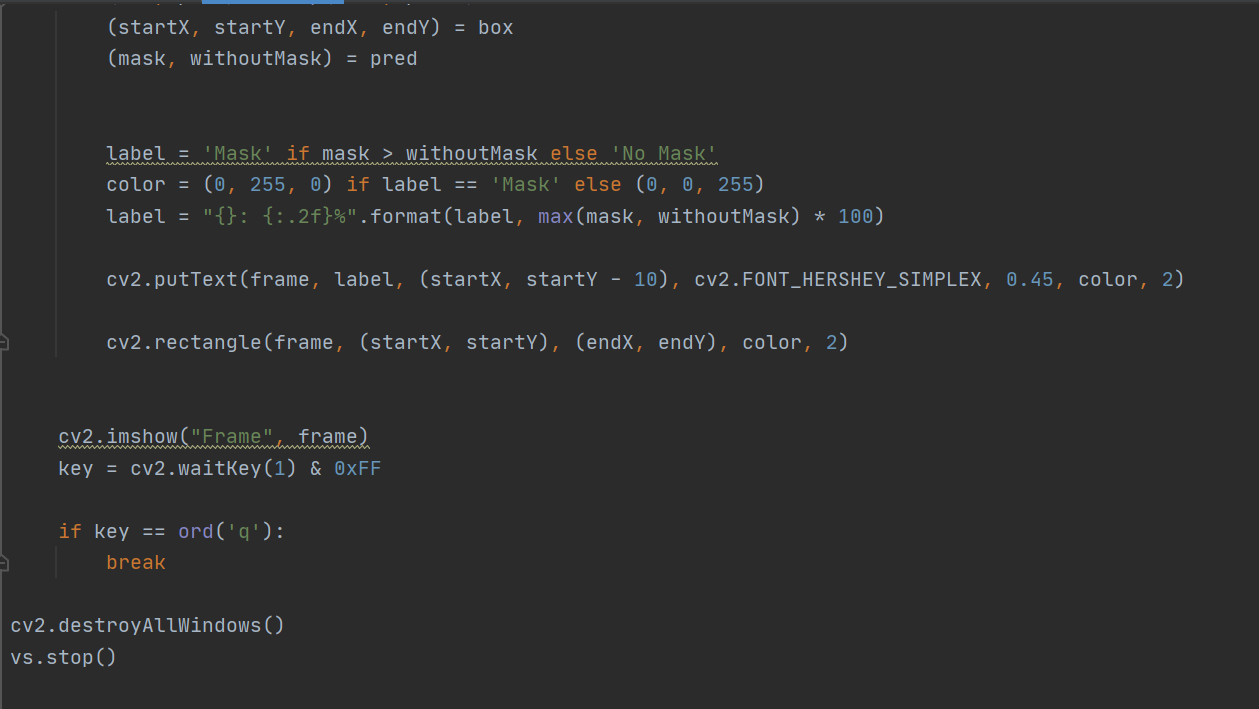
**Coding**

**Mask.py**

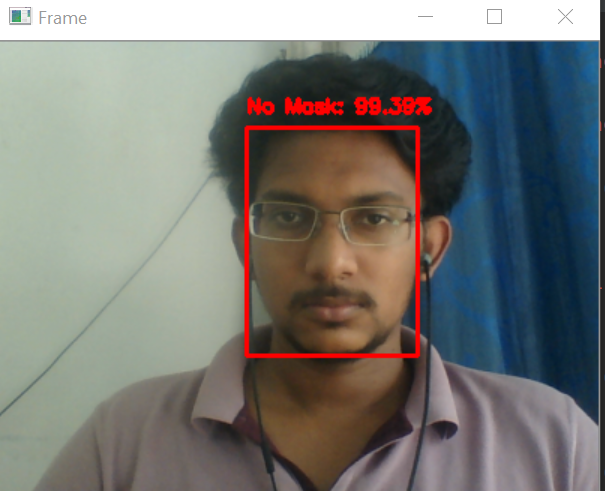
****

****

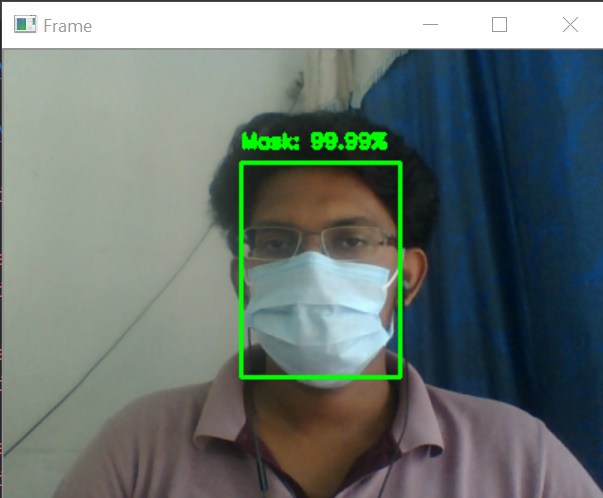
****

****

**9.RESULTS**

****

**Without Mask**

****

**With Mask**

**10.CONCLUSION**

In our project “FACE MASK DETECTION” we got the best results using Convolution Neural Networks (CNN) algorithm with an accuracy of 95%.

**11.REFERENCES**

1. P. Viola and M. Jones, “Rapid object detection using a boosted cascade of simple features,” in Proceedings of the 2001 IEEE computer society conference on computer vision and pattern recognition. CVPR 2001, vol. 1. IEEE, 2001, pp. I–I.
2. Multi-Stage CNN Architecture for Face Mask Detection Amit Chavda, Jason Dsouza, Sumeet Badgujar, Ankit Damani <https://arxiv.org/abs/2009.07627>
3. R. Girshick, J. Donahue, T. Darrell, and J. Malik, “Rich feature hierarchies for accurate object detection and semantic segmentation,” in Proceedings of the IEEE conference on computer vision and pattern recognition, 2014, pp. 580–587.
4. R. Girshick, “Fast r-cnn,” in Proceedings of the IEEE international conference on computer vision, 2015, pp. 1440–1448.
5. Facial Mask Detection Publisher: IEEE Toshanlal Meenpal; Ashutosh Balakrishnan; Amit Verma <https://ieeexplore.ieee.org/document/8888092/authors#authors>
6. S. Ren, K. He, R. Girshick, and J. Sun, “Faster r-cnn: Towards real-time object detection with region proposal networks,” in Advances in neural information processing systems, 2015, pp. 91–99.
7. J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, “You only look once: Unified, real-time object detection,” in Proceedings of the IEEE conference on computer vision and pattern recognition, 2016, pp. 779–788
8. Haddad, J., 2020. How I Built A Face Mask Detector For COVID-19 Using cnn. [online] Medium. Available at <https://towardsdatascience.com/covid-19-face-mask-detection-using-tensorflow-and-opencv-702dd833515b>
9. COVID-19: Face Mask Detector with OpenCV, Keras/TensorFlow, and Deep Learning
10. by Adrian Rosebrock <https://www.pyimagesearch.com/2020/05/04/covid-19-face-mask-detector-with-opencv-keras-tensorflow-and-deep-learning/>
11. Real-Time Face Mask Detector with Python, OpenCV, Keras by data flair <https://data-flair.training/blogs/face-mask-detection-with-python/>