Project Report

On

"Face Mask Detection Using Deep Learning"

Submitted for partial fulfilment of Diploma Course in

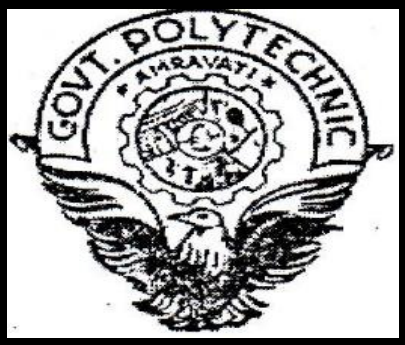
Information Technology

By

AKHILESH J. WAGARE	**[18IF003]**
RAJ A. AMBERE	**[18IF004]**
SARTHAK G. BAND	**[18IF007]**
ARPITA D. BORSE	**[18IF012]**
TOSHIKA I. CHIKHALE	**[18IF014]**

Under the Guidance of

Prof. P. P. KATGAONKAR



Department of Information Technology

GOVERNMENT POLYTECHNIC AMRAVATI

(An Autonomous Institute of Government of Maharashtra)

2020-2021

Project Report

On

"Face Mask Detection Using Deep Learning"

Submitted for partial fulfilment of Diploma Course in

Information Technology

By

AKHILESH J. WAGARE	**[18IF003]**
RAJ A. AMBERE	**[18IF004]**
SARTHAK G. BAND	**[18IF007]**
ARPITA D. BORSE	**[18IF012]**
TOSHIKA I. CHIKHALE	**[18IF014]**

Under the Guidance of

Prof. P. P. Katgaonkar

Department of Information Technology

Government Polytechnic Amravati

(An Autonomous Institute of Government of Maharashtra)

2020-2021

Department of Information Technology

Government Polytechnic Amravati

(An Autonomous Institute of Government of Maharashtra)

2020-2021

CERTIFICATE

This is to certify that the Project entitled

"Face Mask Detection Using Deep Learning"

is submitted by

AKHILESH J. WAGARE	**[18IF003]**
RAJ A. AMBERE	**[18IF004]**
SARTHAK G. BAND	**[18IF007]**
ARPITA D. BORSE	**[18IF012]**
TOSHIKA I. CHIKHALE	**[18IF014]**

In the partial fulfilment of the course *IF5468 – Project Execution and Report Writing* of the Diploma Programme in *Information Technology*

Dr. P. P. Karde Head of Department, Information Technology	Prof. P. P. Katgaonkar (Name & Sign of the Guide)	(Name & sign of the External Examiner)	Dr. R. P. Mogre Principle, Government Polytechnic Amravati

ACKNOWLEDGEMENT

It gives us immense pleasure in submitting the Project Execution & Report Writing on topic “FACE MASK DETECTION USING DEEP LEARNING” to our guide Prof. P. P. Katgaonkar who was a constant source of guidance and inspiration for devolving the Project Execution and for preparation of report.

We are also thankful to all the staff members of Information Technology Department, who have indirectly guided and helped us in preparation of the project.

We are extremely thankful to Dr. R. P. Mogre, Pricipal and Dr. P. P. Karde, Head of Department of Information Technology, for providing all the required sources for the successfully completion of our project.

At last, we are thankful to our friends whose encouragement and constant inspiration helped us so for the preparation of the project.

Thanking You

AKHILESH J. WAGARE [18IF003]

RAJ A. AMBERE [18IF004]

SARTHAK G. BAND [18IF007]

ARPITA D. BORSE [18IF012]

TOSHIKA I. CHIKHALE [18IF014]

Department of Information Technology,

Government Polytechnic, Amravati

(An Autonomous Institute of Government of Maharashtra)

YEAR (2020 -2021)

VISION AND MISSION

GOVERNMENT POLYTECHNIC AMRAVATI

Vision

To be vibrant technical institute of global reputes contributing towards the needs of industries and societies.

Mission

- To develop competent diploma engineers suitable for contemporary industrial environment.
- To inculcate socially accept ethics and values among budding engineers.
- To Nurture innovations and entrepreneurship.
- To produce engineers with psychomotor and cognitive skills committed to long-life learning.

INFORMATION TECHNOLOGY

Vision

To ensure excellent education environment with the technical capability in the field of IT Engineering to serve Vibrant Industry and Society.

Mission

- Facilitate students to learn the basics of IT Engineering entailing the technical gain.
- Train students with technical skills with rational capacity to meet the requirements of industry with technological aspect.
- Motivate the students for an advance knowledge in IT Engineering and other value-added programs for their holistic development.
- Imparting Ethical values, leadership and Social Value in Students which transform them into good human being.
- Provide an academic environment and consultancy services to the industrial society in the area of IT and Computer Engineering.

COURSE OUTCOMES

At the end of the course, students will be able to

1. Implement the project planning activities planned in previous course.
2. Finalize the problem for the project.
3. Collect the relevant information and data for identified project topic.
4. Analyze and compile the collected data.
5. Generate project report based on the experiences and project execution carried out.
6. Present the project report using ppt.

INDEX

LIST OF IMAGES / FIGURES

CHAPTER 1

Abstract

And

Introduction..

1.1 ABSTRACT

The Corona Virus pandemic is causing a global health crisis so the effective protection methods is wearing a Face Mask in public areas according to the World Health Organisation (WHO). The COVID-19 forced the government across the world to impose lockdowns to prevent virus transmissions. Reports indicate wearing the face mask while at work or in public placing clearly reduces the risk of transmission. An efficient and economic approach of using AI to create a safe environment in manufacturing setup. A hybrid model using deep and classical machine learning for face mask detection will be presented.

A face mask detection dataset consists of **with mask** and **without mask** images. We are going to use OpenCV to do real-time face detection from a live stream via our webcam. We will use the dataset to build a COVID-19 face mask detector with computer vision with Python, OpenCV, TensorFlow and Keras. Our goalis to identify whether a person or image/video stream is wearing a face mask or not with the help of computer vision and deep learning.

In this project we have proposed Face Mask Detection system which consists of 2 phases, namely training and deployment. The first stage detects human faces, while the second stage uses deep learning to firstly, identify the ROI (Reign Of Interest) being the person's face and secondly identify the faces detected in the first stage as either 'With Mask' or 'Without Mask' and draws the boundary of colours, either green or red, depending on the output generated. The project can give 99.99% correct result if set up with CCTV camera to track people without masks to ensure the safety and well being of others, thus it can help in controlling the spread of virus.

1.2 INTRODUCTION

After the breakout of the worldwide pandemic COVID-19, there arises a severe need of protection mechanisms, face mask being the primary one. The basic aim of the project is to detect the presence of a face mask on human faces on live streaming video as well as on images. We have used deep learning to develop our face detector model. Alongside this, we have used basic concepts of transfer learning in neural networks to finally output presence or absence of a face mask in an image.

Children and old people have proved to be at the highest risk of contract to the disease, which may even result to be death. Hence, it has been made a priority to stay safe from the virus. The virus spreads through the air, transmitted by person to person not only by touch, but also by speaking and coughing. Putting a face mask on can rude the risk of getting infected by a great extent, not only to the one who is wearing the mask, but it also protect the other person who comes in contact with him/her.

Figure 1.1.1 :- Face Mask Detection

CHAPTER 2

Literature Survey..

LITERATURE SURVEY

As many countries continue their desperate fight to control the infection rate and spread of virus. We have seen the interaction of many unfamiliar and somewhat inconvenient, protection measures being introduced from the past 1 – 1½ years. One of these measures is the mandatory requirement of wearing a protective face mask in shops, cafes, restaurants and the other compact or enclosed social environment. This project aims at automating the task of checking whether someone is wearing a protective mask through the development, training and deployment of a Computer Vison Machine Learning Model. The specific Machine Learning model used in this project is the widely popular Convolutional Neural Network (CNN).

2.1 Machine Learning

Machine Learning or ML is a study of computer algorithms that learns and enhance automatically through experience. It seems to be a subset of artificial intelligence. A machine learning algorithm builds a mathematical model based on "training data", in order to make decisions or predictions without being explicitly programmed to do so.

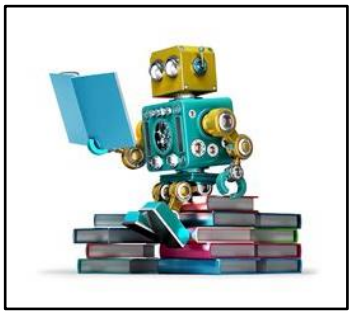


Figure 2.1.1 :- Machine Learning

Machine Learning algorithms are used in variety of applications from email filtering to computer recognition, where it is difficult or impossible to develop general skills to perform the required tasks. These studies are closely related to computer statistics, which focus on computer-generated domain. The data prediction and mining is a coherent field of study, focusing on the analysis of experimental data by unsupervised learning. Hence, as it helps in prediction, it is also called as **predictive analytics.**

Machine learning approaches are traditionally divided into three broad categories, depending on the nature of the "signal" or "feedback" available to the learning system. The categories are as follows :-

1. Supervised Learning
2. Unsupervised Learning
3. Reinforcement Learning

2.2 Deep Learning

Deep Learning is a subset of Machine Learning, which is essentially a neural network with three or more layers. These neural network attempts to stimulate the behaviour of human brain allowing it to learn from large amount of data. While a neural network with a single layer can still make approximate predictions, additional hidden layers can help to optimize and refine for accuracy.

Deep Learning neural network or artificial neural networks are the elements which work together to accurately recognize, classify and describe objects within the data. Deep neural networks consist of multiple layers of interconnected nodes, each building upon the previous layer to refine and optimize the prediction for categorization. This progression of computations through the network is called as **forward propagation.** Another process called **backbone propagation** uses the algorithms, like gradient descent, to calculate errors in predictions and then adjust the weights and biases of the function by moving backwards through the layers in an effort to train the model.

The input and output layers of deep neural network are called as **visible layers.** The input layers is where the deep learning model ingests the data for processing, and the output layer is where the final prediction or classification is made. Together forward propagation and backward propagation allow neural network to make predictions and correct for any errors accordingly.

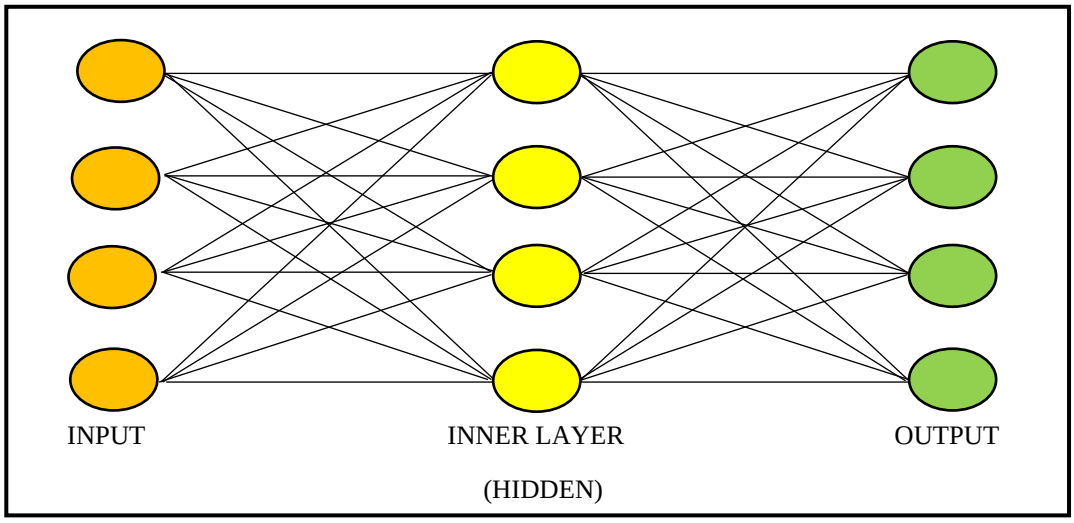


Figure 2.2.1 :- Deep Learning Architecture

2.3 Computer Vision

Computer Vision is an interdisciplinary specific field that deals with how computers can gain high level understanding from the digital images or videos. From the perspective of engineering, it seeks to understand and automate tasks that the human visual system can do. Computer Vision tasks include methods for acquiring, processing, analysing and understanding digital images and extraction of high dimensional data from the real world in order to produce numerical or symbolic information e.g. in the forms of decisions. This image understanding can be seen as the disentangling of symbolic information from image data using models constructed with the aid of geometry, physics, statistics and learning theory.

Computer Vision is concerned with the automatic extraction, analysis and understanding of useful information from a single image or a sequence of images. It involves the development of theoretical and algorithmic basis to achieve automatic visual understanding. As a scientific discipline, computer vision is concerned with the theory behind artificial systems that extract information from images.

Computer Vision works much the same as human vision, except humans have a head start. Human sight has the advantage of lifetimes of context to train how to tell objects apart, how far away they are, whether they are moving and whether there is something wrong in an image. Computer Vision trains machine to perform these functions, but it has to do it in much less time with cameras, data and algorithms rather than retina, optic nerves and visual cortex.

Figure 2.3.1 :- Computer Vision Scenario

2.4 Open-CV

OpenCV is a Python Library that is used to solve computer vision problems. In **OpenCV**, **CV** is an abbreviation for Computer Vision. Computer Vision include understanding and analysing digital images by the computer and process the images or provide relevant data after analysing the image. OpenCV is a open-source library used in machine learning and image processing. It performs various tasks such as recognizing handwritten digits, human faces and objects. It was built to provide the common infrastructure for computer vision applications and to accelerate the use of machine perception in the commercial products.

The library has more than 2500 optimized algorithms which includes a comprehensive set of both classic and state-of-the-art computer vision and machine learning algorithms. These algorithms can be used to detect and recognize faces, identify objects, classify human actions in videos, etc. This library is used extensively in companies, research groups and by governmental bodies. It has C++, Python, Java and MATLAB interfaces and supports Windows, Linux, Android and Mac OS. OpenCV leans mostly towards real-time vision applications.

We use OpenCV library to execute infinite loops using our webcam, which detects faces using cascade classifications. It is continuously adding new modules to the latest algorithms from machine learning.

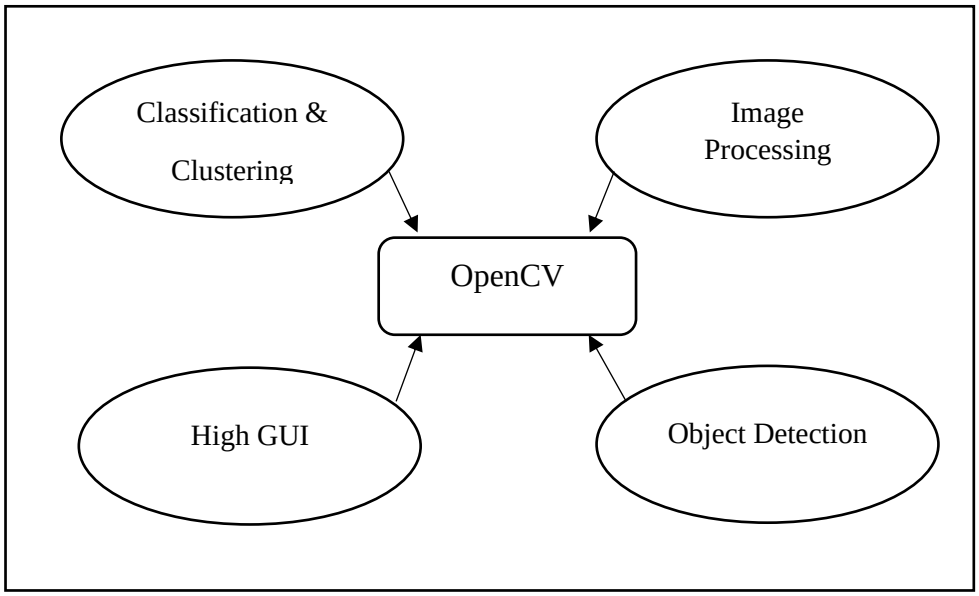


Figure 2.4.1 :- OpenCV Main Functions

2.5 TensorFlow

TensorFlow is a free and open-source software library for machine learning. It can be used across a range of tasks but has a particular focus on training and inference of deep neural networks. TensorFlow is a symbolic math library based on dataflow and differentiable programming. While reference implementation runs on single devices, TensorFlow can run on multiple CPUs and GPUs.

TensorFlow is a great system for handling all aspects of a machine learning system. However, this class focuses on using the unique Tensor Flow API to train and deploy machine learning models. Here, we used TensorFlow and Keras to train the classifier to automatically identify if a person is wearing a mask or not. Since reference implementation runs on single devices, TensorFlow is able to run on multiple processing units and GPUs having extension regarding general use.

The name Tensor Flow derives from the operations that such neural networks perform on multidimensional data arrays, which are referred to as **tensors.** Unlike the other numerical libraries intended for use in Deep Learning like Theano, TensorFlow was designed for use both in research and development and in production systems.

2.6 Keras

Keras is an open-source software library that provides a Python interface for artificial neural networks. Keras acts as an interface for the TensorFlow library. Designed to enable fast experimentation with deep neural networks, it focuses on being user-friendly, modular and extensible.

Keras contains numerous implementations of commonly used neural-network building blocks such as layers, objectives, activation functions, optimizers and a host of tools to make working with image and text data easier to simplify the coding necessary for writing deep neural network code. In addition to standard neural networks, Keras has support for convolutional and recurrent neural networks. It supports other common utility layers like dropout, batch normalization and pooling.

Keras allows users to productize deep models on smartphones, on the web or on the Java Virtual Machine (JVM). It also allows use of distributed training of deep-learning models on clusters of Graphical Processing Units (GPUs) and Tensor Processing Units (TPUs).

CHAPTER 3

Methodology..

METHODOLOGY

This Face Mask Detection system focuses on how to identify whether a person on the image or video stream is wearing face mask or not with the help of computer vision, deep learning and machine learning algorithms by using the libraries OpenCV, TensorFlow and Keras.

3.1 Approach

To implement our model, we have to approach our system in 2 important phases which are as follows :-

1. Train the model using the libraries TensorFlow and Keras.
2. Deploy the trained model and applying the detector over the images or live video stream.

3.2 Flow Chart

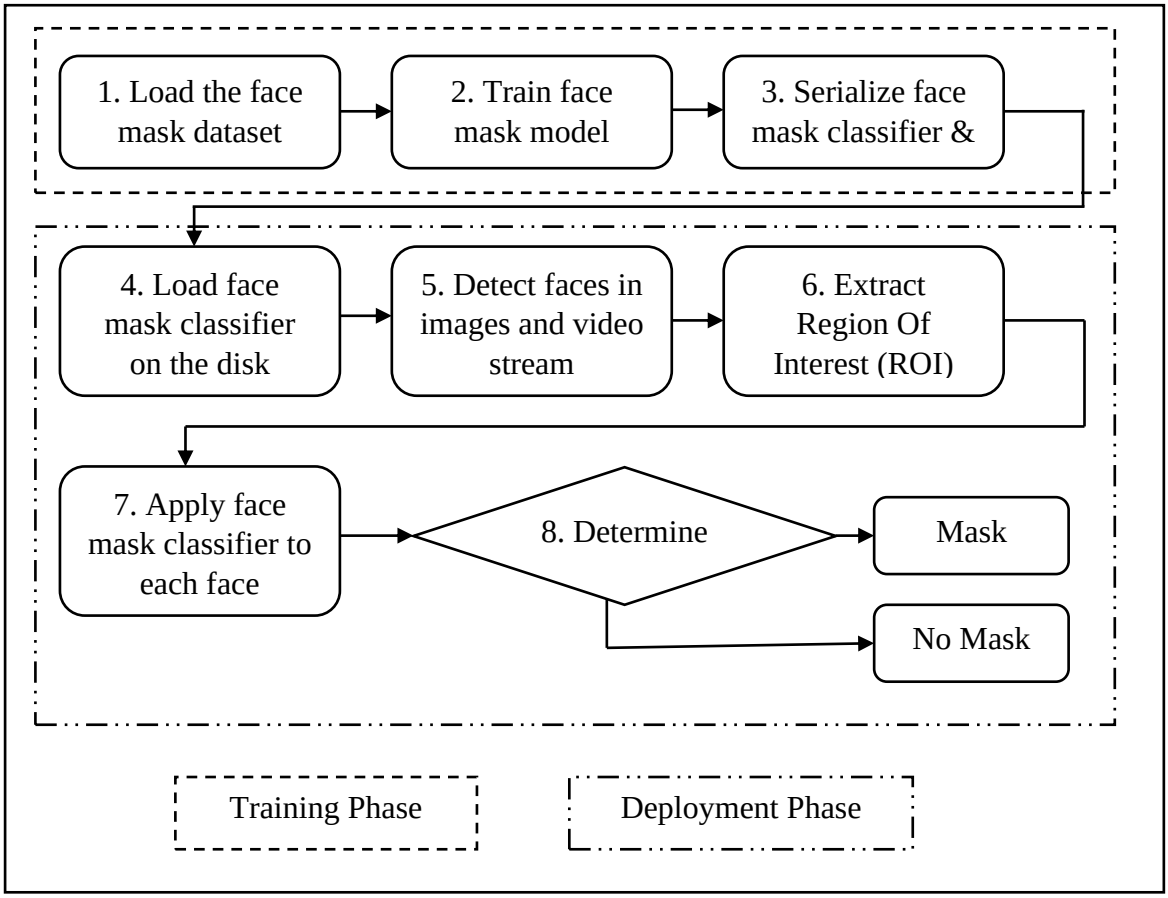


Figure 3.2.1 :- Flow Chart of Implementation

3.3 Source Data

In this project, we have given a group of lot of images which are augmented (or built up) by the library OpenCV. The set of images are labelled or included in the two different folders called “with_mask” and “without_mask”. The source images are of different sizes and resolutions, since they are accessed from the different websites or machines or cameras of different resolutions.

3.4 Dataset Building

Phase 1:

The dataset is first loaded in the training phase. Training and modelling are streamlined during the training phase. Here, the different images (with_mask and without_mask) are added and separated for the evaluation and the different grayscale values are generated for the different types of images.

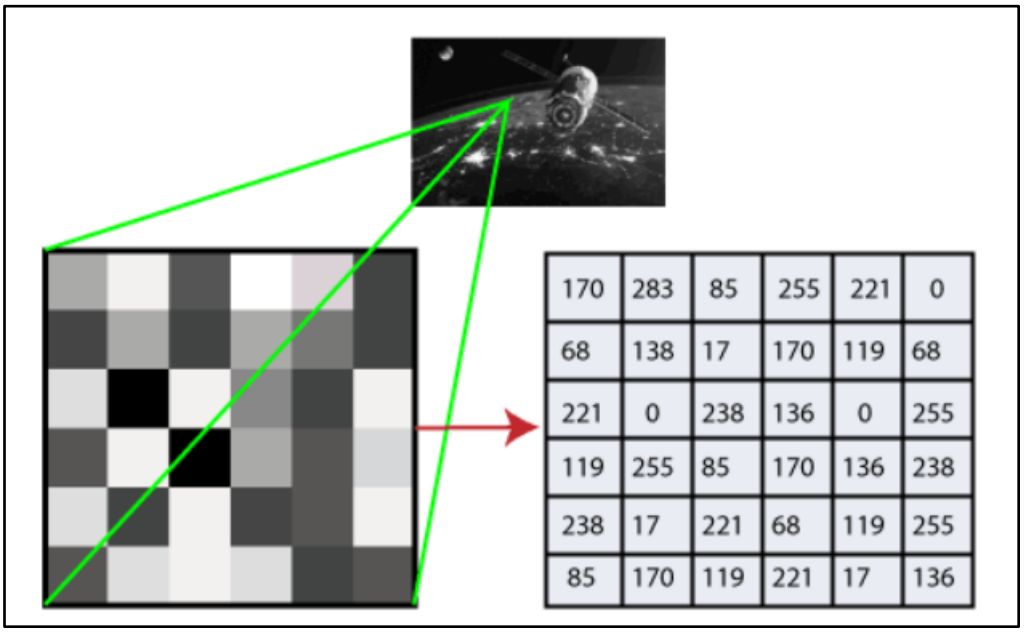


Figure 3.4.1 :- Phase 1

Phase 2:

After serializing face mask classifier to the disk, model is then loaded to detect the face mask on the images or real-time video. The model will calculate ROI (Region of Interest) for determination. We then compute bonding box value for a particular face and ensure that the box falls within the boundaries of image.

3.5 Face Mask Detection

We then determine the class label based on the predictions returned by the mask detector model and according to the predictions returned, the colours are assigned to the box for interpretation.

Colours assigned according to the predictions are as follows :-

1. If the predictions fall in “with_mask” value, the colour assigned is – GREEN.
2. If the predictions fall in “without_mask” value, the colour assigned is – RED.

NOTE :- The predictions are made with the probability of approximately 60 – 62%. If the person in front of the video stream is wearing the mask and if the values are generated for the mask are above 60 – 62%, the prediction will be on the GREEN side (that is, “with_mask” value) and same for the person who is without the mask.

Person with / without mask	**Probability for grayscale values**	**Result**	**Colour Generated**	**Remark**
With mask	> 60% for mask	True	GREEN	Mask in proper way
With mask	< 60% for non-mask	False	RED	Mask not in proper way
Without mask	> 60% for non-mask	True	RED	Mask not in proper way

Figure 3.5.1 :- Colour Generation Table

CHAPTER 4

Experimental Work...

EXPERIMENTAL WORK

4.1 Dataset

- Some images of "with_mask" side provided in the source dataset are as follows –

Figure 4.1.1 :- Dataset of "WITH_MASK"

- Some images of “without_mask” side provided in the source dataset are as follows –

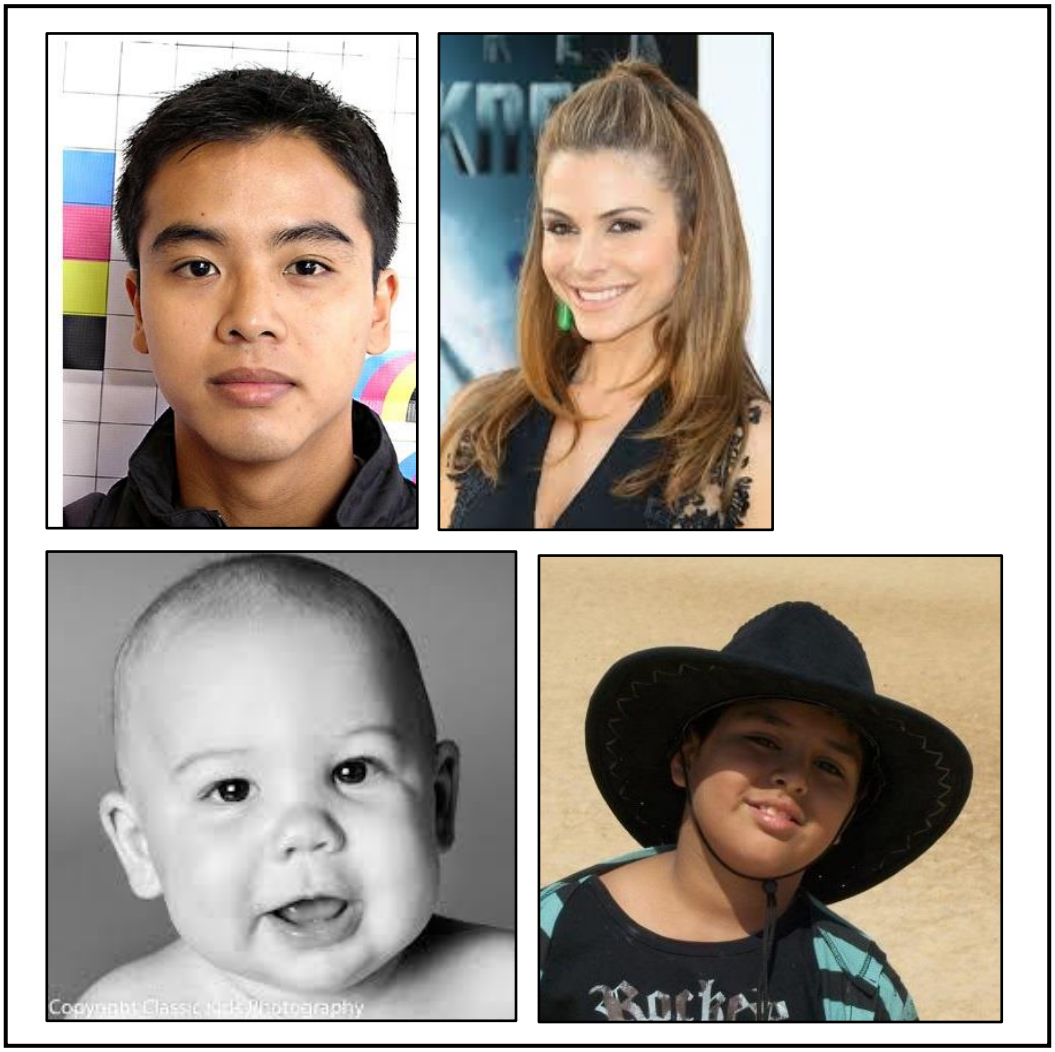


Figure 4.1.2 :- Dataset of "WITHOUT_MASK"

4.2 Code

Code of the Face Mask Detection system is uploaded on the GitHub. The link is given below. You can download it from here :-

Link – https://github.com/AkhiWagare/FaceMaskDetectionUsingDeepLearning

CHAPTER 5

Result..

RESULT

The result of our Face Mask Detection system (when home.py file is executed) is returned in various ways which are as follows :-

1. Starting Interface –

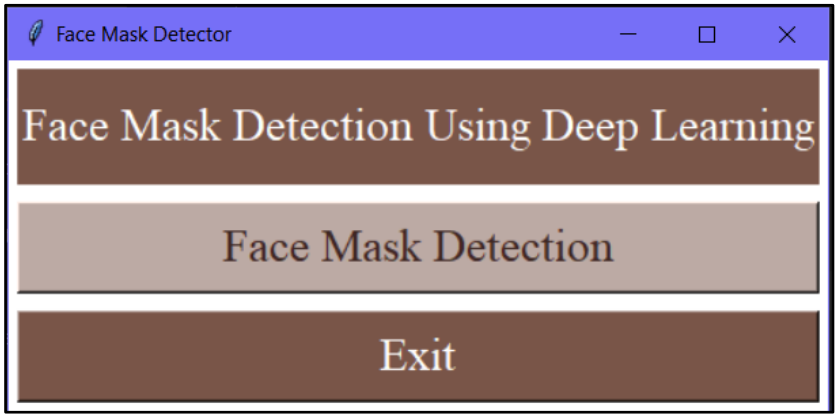


Figure 5.1.1 :- Starting Interface

2. Person in front of camera wearing mask in proper way –

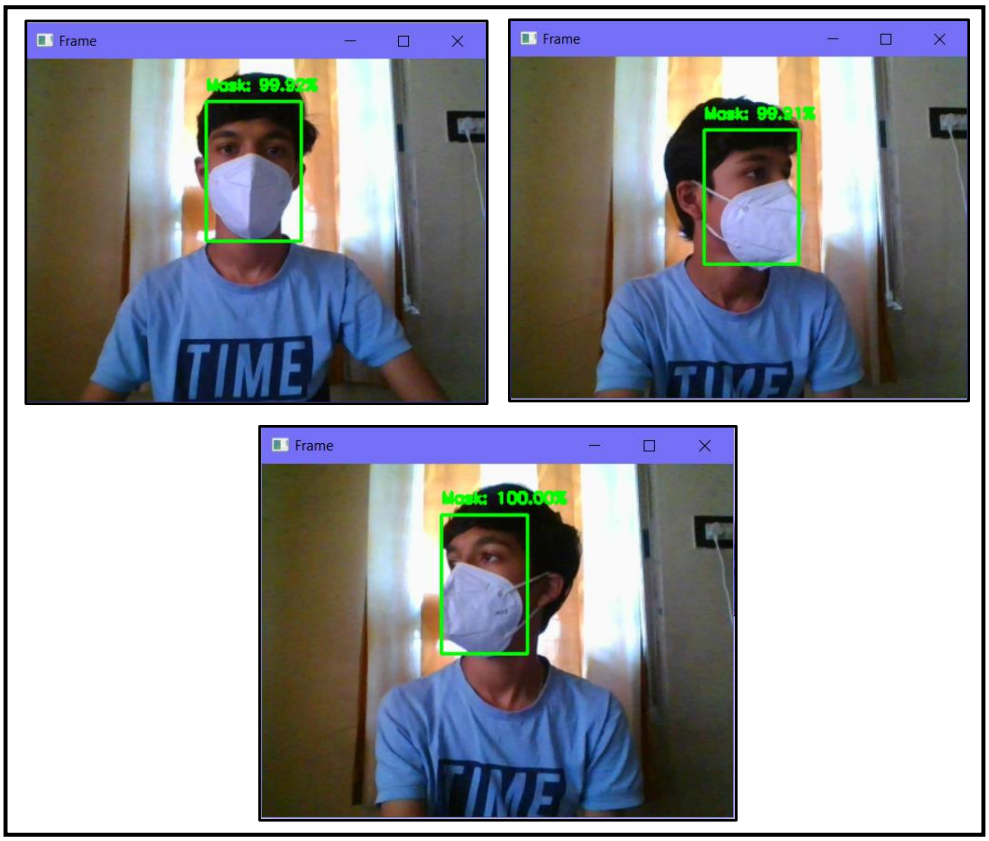


Figure 5.2.1 :- Mask in Proper Way

3. Person in front of camera wearing mask in improper way –

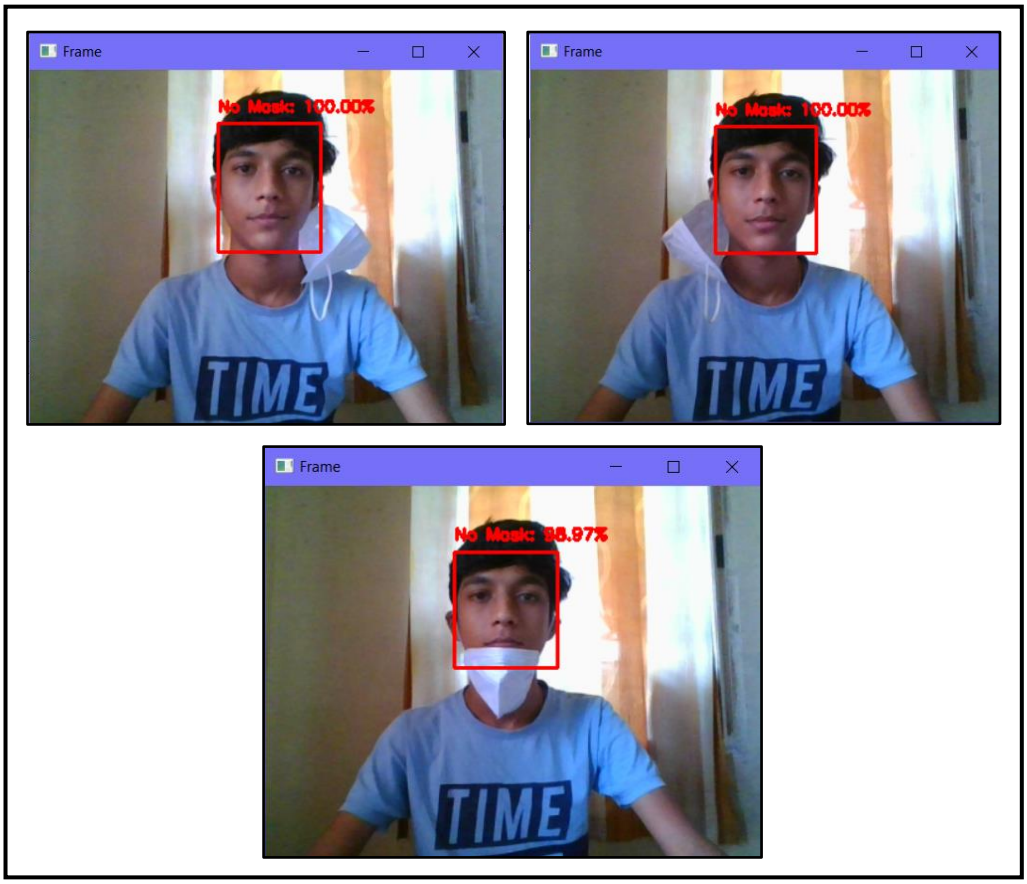


Figure 5.3.1 :- Mask in Improper Way

4. Person in front of camera using hand instead of mask –

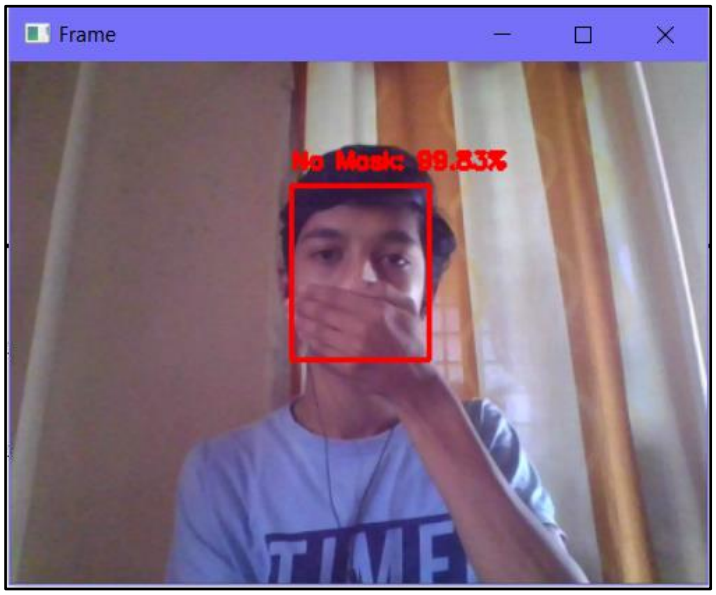


Figure 5.4.1 :- Hand instead of Mask

5. Person in front of camera without wearing mask –

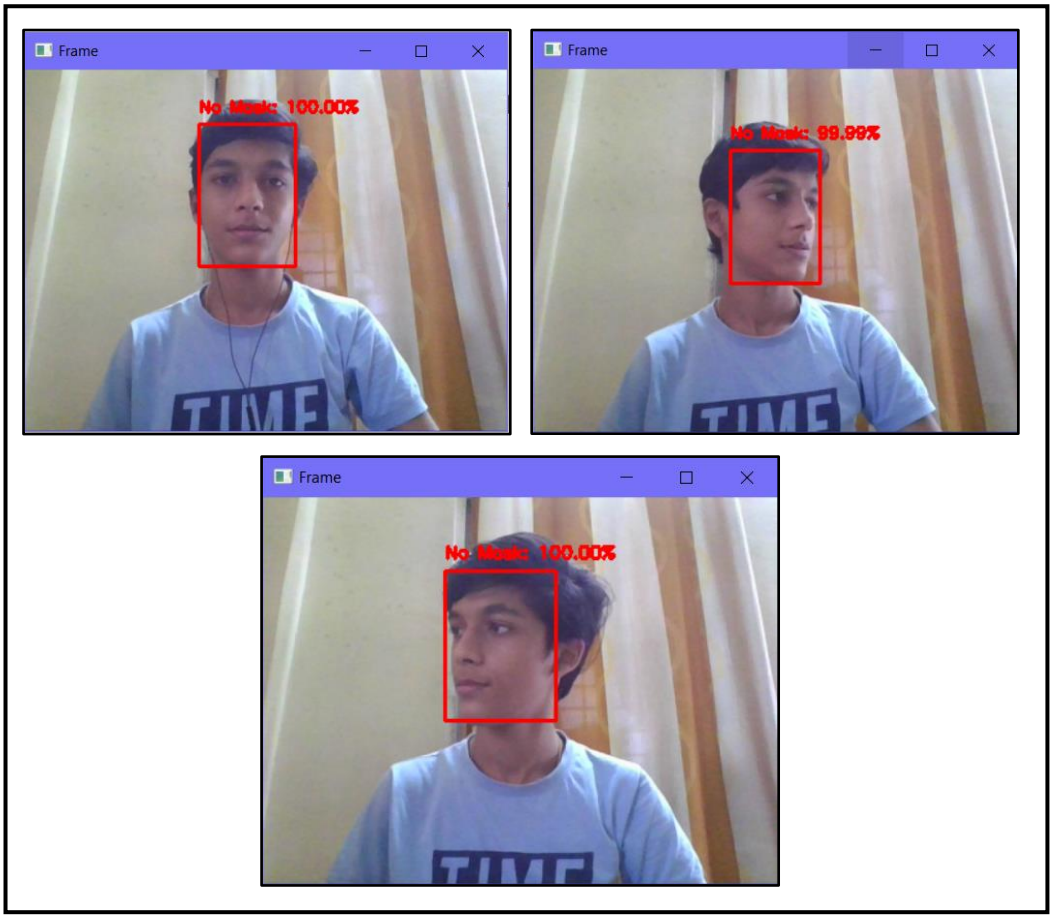


Figure 5.5.1 :- Without Wearing Mask

CHAPTER 6

Conclusion

And

Future Scope..

6.1 CONCLUSION

As we can see, the technology is increasing rapidly and so is the demand with the trends emerging day-by-day. Hence, we have made a Face Mask Detection System Using Deep Learning which can possibly contribute in the health care of the public in public areas in such situation of crisis like COVID-19. This system is built with the help of machine learning and its various libraries such as OpenCV, TensorFlow, Keras, etc. successfully. This system importantly consists of the library OpenCV which we can also called as the backbone of the system. This system is successfully tested with real-time video streams of various people and it can successfully provide the accurate result by extracting the individual images from the video streams.

6.2 FUTURE SCOPE

In this work, a deep learning-based approach for detecting masks over faces in a public place to curtail community spread of Coronavirus is presented. The proposed technique efficiently handles varying kinds of occlusions in the dense situation by making use of an ensemble of single and two stage detectors at the pre-processing level. Finally, the work opens interesting future directions for researchers. They can see who has not covered faces with masks through software, mobile app, device, or a website.

Firstly, the proposed technique can be integrated into any high-resolution video surveillance devices and not limited to mask detection only. Secondly, the model can be trained and upgraded to mask datasets that include different images related to correctly/incorrectly wear mask and achieve the ultimate purpose of detecting facemask for cutting down the risk of contagious diseases. We can use this system in the high-crowded public places such as malls where, there are very high chances of disease spreading.

CHAPTER 7

References..

REFERENCES

7.1 Books and Journals

[1] "Face Mask Detection using Transfer Learning of Inception" by – G. Jignesh Chowdary, Narinder Singh Punn, Sanjay Kumar Sonbhadra and Sonali Agarwal.

[2] "Face Detection Techniques: a review" by – A. Kumar, A. Kaur and M. Kumar.

[3] "Deep Learning and control algorithms of direct perception for autonomous driving" by – D. -H. Lee, K. -L. Chen, K. -H. Liou, C. -L, Liu and J. -L. Liu.

[4] "Machine Learning with Python for Everyone" by – Pearson.

[5] "Hands-On Machine Learning with Scikit-Learn, Keras & TensorFlow: Concepts, Tools and Techniques to Build Intelligent Systems" by – Aurelien Geron.

[6] "Deep Learning with Python" by – Francois Chollet.

[7] "Deep Learning" by – lan Goodfellow, Yoshua Bengio and Aaron Courville.

[8] "Learn TensorFlow 2.0: Implement Machine Learning and Deep Learning Models with Python" by – Pramod Singh and Avinash Manure.

[9] "Advanced Deep Learning with TensorFlow 2 and Keras: Apply DL, GANs, VAEs, deep RL, unsupervised learning, object detection and segmentation, and more" Second Edition by – Rowel Atienza.

[10] "Hands-On Computer Vision with TensorFlow2: Leverage deep learning to create powerful image processing apps with TensorFlow 2.0 and Keras" by – Benjamin Planche and Eliot Andres.

7.2 Websites

[1] https://www.ibm.com/cloud/learn/deep-learning

[2] https://www.oreilly.com/library/view/deep-learning/9781491924570/assets/dpln_0401.png

[3] https://www.ibm.com/topics/computer-vision

[4] https://www.tutorialspoint.com/how-to-install-opencv-in-python

[5] https://en.wikipedia.org/wiki/TensorFlow

[6] https://en.wikipedia.org/wiki/Keras

[7] https://keras.io/

[8] https://www.javatpoint.com/opencv

[9] https://en.wikipedia.org/wiki/Deep_learning

[10] https://www.investopedia.com/terms/d/deep-learning.asp

[11] https://www.tensorflow.org/

[12] https://docs.opencv.org/master/d0/de3/tutorial_py_intro.html