A Major Project Report on

SOCIAL DISTANCING AND FACE MASK DETECTION

Submitted to partial fulfillment of the requirements for the award of the degree of

BACHELOR OF TECHNOLOGY
in
COMPUTER SCIENCE & ENGINEERING

By

G. PRADYUMNA REDDY	(17911A0521)
M. JITIN KRISHNA	(17911A0534)
M. ARAVINDH SAI	(17911A0539)
T. PRAVEEN	(17911A0553)

Under the Esteemed Guidance of

P. RAJASHEKAR
Assistant Professor

Department of Computer Science & Engineering
VIDYA JYOTHI INSTITUTE OF TECHNOLOGY
(An Autonomous Institution)
(Approved by AICTE, New Delhi & Affiliated to JNTUH,
Hyderabad) Aziz Nagar Gate, C.B. Post, Hyderabad-500075

2020-2021

A Major Project Report on

SOCIAL DISTANCING AND FACE MASK DETECTION

Submitted to partial fulfilment of the requirements for the award of the degree

of

BACHELOR OF TECHNOLOGY

in

COMPUTER SCIENCE & ENGINEERING

By

G. PRADYUMNA REDDY	(17911A0521)
M. JITIN KRISHNA	(17911A0534)
M. ARAVINDH SAI	(17911A0539)
T. PRAVEEN	(17911A0553)

Under the Esteemed Guidance of

P. RAJASHEKAR

Assistant Professor

Department of Computer Science & Engineering

VIDYA JYOTHI INSTITUTE OF TECHNOLOGY

(An Autonomous Institution)

(Approved by AICTE, New Delhi & Affiliated to JNTUH, Hyderabad)

Aziz Nagar Gate, C.B. Post, Hyderabad-500075

2020-2021

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

CERTIFICATE

This is to certify that the project report titled "SOCIAL DISTANCING AND FACE MASK DETECTION" is being submitted by **G. PRADYUMNA REDDY (17911A0521), M. JITIN KRISHNA (17911A0534), M. ARAVINDH SAI (17911A0539), T. PRAVEEN (17911A0553)** in partial fulfillment for the award of the Degree of Bachelor of Technology in Computer Science & Engineering, is a record of bonafide work carried out by them under my guidance and supervision. The results embodied in this project report have not been submitted to any other University or Institute for the award of any degree of diploma.

Internal Guide

P. Rajashekar

Assistant Professor

Head of Department

Dr. B. Vijayakumar

Professor

Head of the Department
Computer Science and Engineering
VJIT, Hyderabad-50075.

External Examiner

DECLARATION

We, G. PRADYUMNA REDDY, M. JITIN KRISHNA, M. ARAVINDH SAI, T. PRAVEEN hereby declare that the project entitled, "SOCIAL DISTANCING AND FACE MASK DETECTION" submitted for the degree of Bachelor of Technology in Computer Science and Engineering is original and has been done by us and this work is not copied and submitted anywhere for the award of any degree.

Date: 10 | 07 | 2021

Place: Hyderabad

G. PRADYUMNA REDDY	**(17911A0521)**
M. JITIN KRISHNA	**(17911A0534)**
M. ARAVINDH SAI	**(17911A0539)**
T. PRAVEEN	**(17911A0553)**

ACKNOWLEDGEMENT

We wish to express our sincere gratitude to the project guide, **P. Rajashekar**, Assistant Professor, Vidya Jyothi Institute of Technology, Hyderabad for his timely cooperation and valuable suggestions while carrying out this work. It is his kindness that made us learn more from him.

We are grateful to **Dr. B. Vijayakumar**, Professor and the HOD, department of CSE, for his help and support during our academic year.

We whole-heartedly convey our gratitude to the Principal **Dr. A. Padmaja** for her constructive encouragement.

We would like to take this opportunity to express our gratitude to the Director **Dr. E. Sai Baba Reddy** for providing necessary infrastructure to complete this project.

We thank our parents and all the faculty members who have contributed to our progress through the course to come to this stage.

G. PRADYUMNA REDDY	**(17911A0521)**
M. JITIN KRISHNA	**(17911A0534)**
M. ARAVINDH SAI	**(17911A0539)**
T. PRAVEEN	**(17911A0553)**

ABSTRACT

SOCIAL DISTANCING AND FACE MASK DETECTION

Social Distancing and Face Mask Detection is a Machine Learning project. It is used to detect social distancing and face mask violations identifying input images, videos and input from webcam. It can be used in identifying and monitoring people violating social distancing and face masks from CCTV footage also. It also generates an alarm when the social distancing or face mask violation is detected. It also gives statistics of violations. In this pandemic period, to prevent the spread of virus in various places and alert people to wear masks and maintain social distance, this application can be used. This application is used to aware people who are not maintaining social distance and not wearing masks in public places. It plays a vital role in reducing transmission of virus.

INDEX

LIST OF FIGURES

CHAPTER-1

INTRODUCTION

After Covid-19 outbreak, to protect ourselves and the people around us we need to wear mask and follow social distancing at every place. Masks are the barrier to help prevent the respiratory droplets reaching from one person to another person. They play a vital role in reducing the viral transmission. It also slows the spread of virus. Social Distancing also stops the spread of the virus. Social distancing puts space between individuals. If someone is sick and there are no people around, the virus cannot spread in that situation. It helps in staying away from people so as to limit the spread of virus.

The main objective of this application is to detect the social distance and face mask violations. The major reason for spread of virus is violating social distancing and not wearing a face mask. So, our project detects those violations and generates an alarm. In this way, the person who is violating the social distancing and face mask protocols will be alerted. Hence, the person realizes that he is violating the rules and then stops violating. In this way, the spread of virus can be reduced.

This application provides a friendly user interface. It mainly focuses on detecting the violations of social distancing and face mask and generates the alarm to alert the person who is violating.

CHAPTER-2

LITERATURE SURVEY

The Social Distancing and Face Mask Detection application is a Machine Learning application. It detects the violations of social distancing and face mask protocols. This section discusses findings and observations done by some research works on the application. The already existing application sends an SMS to the registered user mobile if the person is violating the social distancing or faces mask protocols. But, the person may or may not check the message that he is violating the protocol. The already existing application is only suitable to be used at a specific premise such as in a company. Because, in the company all the employee details will be known and it is easy for the application to identify the person who is violating the protocol and can send the SMS to that particular person. But this application is not suitable to be used in public places like vegetable markets, streets etc. because it is impossible to identify the unknown person from the public place and send him SMS alert.

The purpose of these literature reviews was to collect information on how social distancing and face masks protocols can reduce the transmission of virus and also the drawbacks of existing system that detect the social distancing and face mask violations. Based on the reviews, it was found out that if a system is developed that can generate an alarm immediately at the place where social distancing or face mask protocols are violated, it would be more effective especially at public places. In this way, the spread of virus can be reduced.

CHAPTER-3

FEASIBILITY STUDY

The feasibility of the project is studied during this phase and some cost estimates are analyzed with a normal plan for the project is taken forward. During the system analysis the feasibility study is carried to make sure that the project is done smoothly and not going to be a load for the company.

Following are the four considerations involved in the feasibility study:

1. ECONOMICAL FEASIBILITY
2. TECHNICAL FEASIBILITY
3. SOCIAL FEASIBILITY
4. SCHEDULE FEASIBILITY

3.1 ECONOMICAL FEASIBILITY:

This study is mainly stressed to check the monetary impact of the system on the organization. It deals with the sum of money that the company can put forth on the research and also for the enhancement is checked. Here we should make sure that the ideal framework should be developed within the spending limit, this can be obtained by using some of the advancements which are freely accessible and only purchasing the products that are unreservedly accessible.

3.2 TECHNICAL FEASIBILITY:

This study is mainly related to finding the technical feasibility, which means it entirely focuses on the technical requirements of the framework. There should not be extreme interest in the accessible technical resources for any of the systems.

If the interest is high on the accessible technical assets it will prompt to high demands of the client. The system which ought to have modest necessity, with only few or no changes are required to actualize the proposed system.

3..3 SOCIAL FEASIBILITY:

This study handles with the level of acknowledgement of the system by the user. This study also includes the way of training the user to utilize the system efficiently. The user ought to use the system productively and should not compromise with the system. The measure the user accepts the system relies on the way we educate the user by legitimate procedures and should make him acquainted with the system. So that the user feels that he knows the system and his certainty levels will rise to work with the system which is invited accomplishment for a project.

3.4 SCHEDULE FEASIBILITY:

In this study our main focus is centered on the time to complete the project. A project is said to be fizzled if it takes more computational time to complete before it is being useful. In accordance, this means it is an estimate to find out how much time the system will take to fully develop, and to find out whether it can be finished in the specified time by using few techniques like payback period. Schedule feasibility is a measure to find how meaningful the project schedule is designed. Whether the project is started in time? Is the deadline reasonable? Will it be finished in time? And whether a deadline is necessary or not a small deviation can be implied in the original schedule which was opted at the beginning of the project. The project development is doable as far as schedule is considered.

CHAPTER-4

SYSTEM REQUIREMENTS

4.1 EXISTING SYSTEM:

The existing system is similar to the covid-19 monitoring solution developed by Softweb Solutions, An Avnet Company. In this system, in case of violation of the social distancing protocols, the system sends notifications to the administrative or security teams. In the LeewayHertz application, a notification is sent mobile as an SMS which the violators may or may not see.

4.2 PROPOSED SYSTEM:

The proposed system is very much attainable and usable. The proposed system generates an alarm immediately whenever any violation is detected. Hence, sending notifications to the administrative or security teams or to the mobile as SMS is not required. By hearing the alarm, people can realize and stop violating the social distancing and face mask protocols.

4.3 SYSTEMREQUIREMENTS:

4.3.1 SOFTWARE REQUIREMENTS:

• Software	:	Anaconda – Jupyter Notebook, Python 3
• Editor	:	VS Code, PyCharm
• Environment	:	TensorFlow
• GPU Drivers	:	Any GPU Driver depending upon system configurations

4.3.2 HARDWARE REQUIREMENTS:

- GPU : Any Graphics Processor with min 8GB dedicated Memory
- Camera : CCTV/ Webcam
- Storage Disk : 500GB HDD is Sufficient

4.4 REQUIREMENTS DEFINITION:

After the severe continuous analysis of the problems that arose in the existing system, we are now familiar with the requirements that are required by the current system. The requirements that the system needs is categorized into the functional and non-functional requirements. These requirements are listed below:

4.4.1 FUNCTIONAL REQUIREMENTS:

Functional requirements define which functions or features that is to be incorporated in any system to fulfill the business requirements and to be acknowledged by the clients. On the premise, the functional requirements specify the relationship between the inputs and outputs. All the operations to be performed on the input data to obtain output are to be specified. This includes specifying the validity checks on the input and output data, parameters affected by the operations and the other operations, which must be used to transform the inputs into outputs. Functional requirements specify the behavior of the system for valid input and outputs.

4.4.2 NON-FUNCTIONAL REQUIREMENTS:

Non-functional requirements provide a description of features, characteristics and capacity of the system and furthermore it may constraints the boundaries of the proposed system.

The following are the non-functional requirements that are essential depending on the performance, cost, control and gives security efficiency and services.

Based on the above explained non-functional prerequisites are as follows:

- User friendly.
- System should provide better accuracy.
- To perform efficiently with better throughput and response time.

CHAPTER-5

SYSTEM DESIGN

5.1 DATA FLOW DIAGRAMS

5.1.1 SOCIAL DISTANCING DATA FLOW DIAGRAM

Input device captures the video. YOLO object detection detects people. The model then calculates the social distancing between people and detects violations.

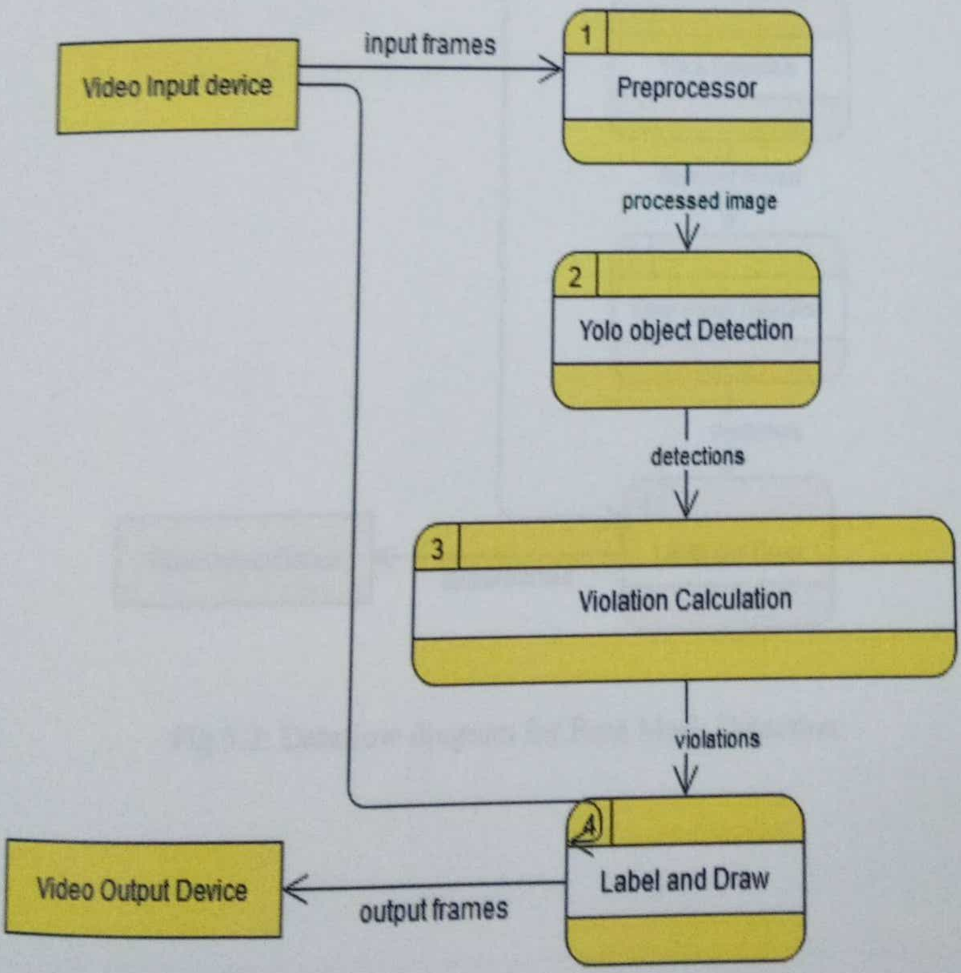


Fig 5.1: Dataflow diagram for Social Distancing Detection

5.1.2 FACE MASK DETECTION DATA FLOW DIAGRAM

Input device captures the video. Tensorflow Neural Network model classifies the person in input frame whether he/she is wearing mask or not.

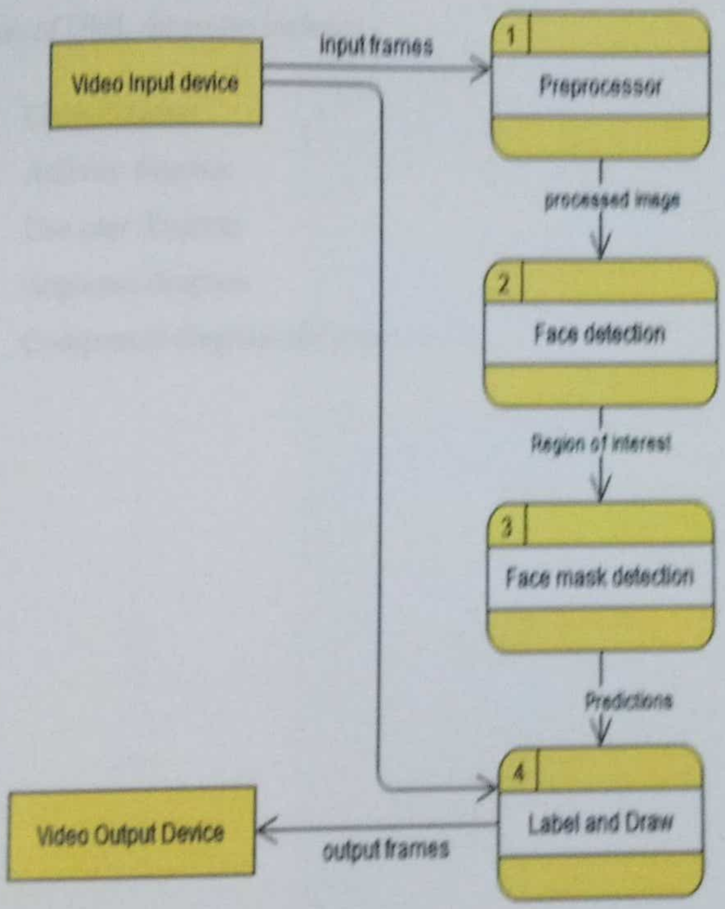


Fig 5.2: Dataflow diagram for Face Mask Detection

5.2 UML DIAGRAMS:

UML diagram is a diagram that is designed based on Unified Modeling language with the aim to visually represent the system with roles, actors, anchors etc. to understand and maintain the system easily. By using this we can better understand flaw so errors in the system so that we can maintain or alter the system properly.

Different types of UML diagrams include:

- Class diagram
- Activity diagram
- Use case diagram
- Sequence diagram
- Component diagram and many more.

5.2.1 USE CASE DIAGRAM:

Usecase diagrams also represented as behavior diagrams. These diagrams are used to explain the set of actions that the system needs to be performed in accordance with the external user.

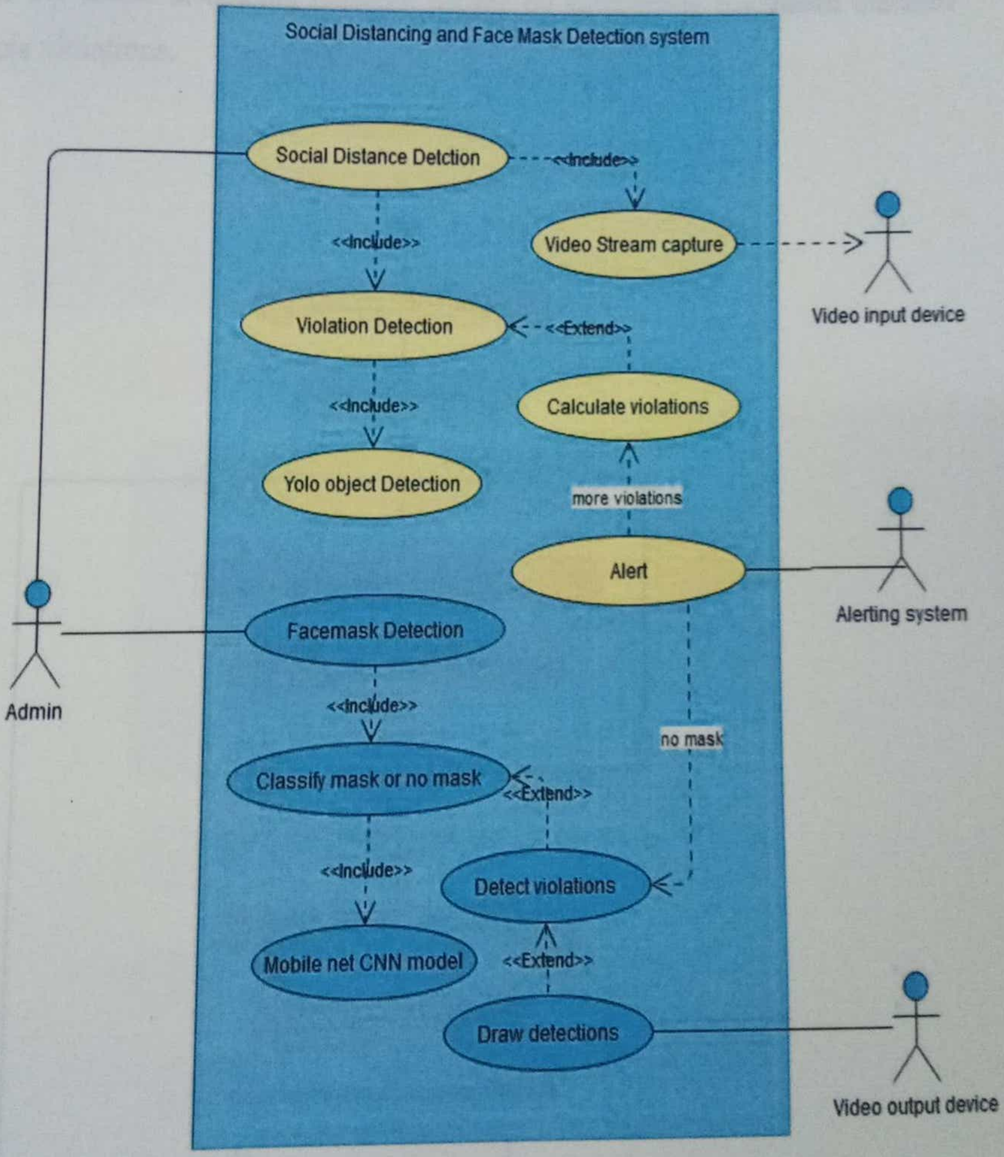


Fig 5.3: Usecase Diagram

5.2.2 ACTIVITY DIAGRAM:

Activity diagram is one of the important UML diagrams, which is used to represent the dynamic aspects of the system. It is basically a flow chart to represent the flow of activity from one activity to the other.

5.2.2.1 Activity diagram for Social Distancing Detection:

YOLO object detection detects people. The model then calculates the social distancing between people by calculating Euclidean distance and detects violations.

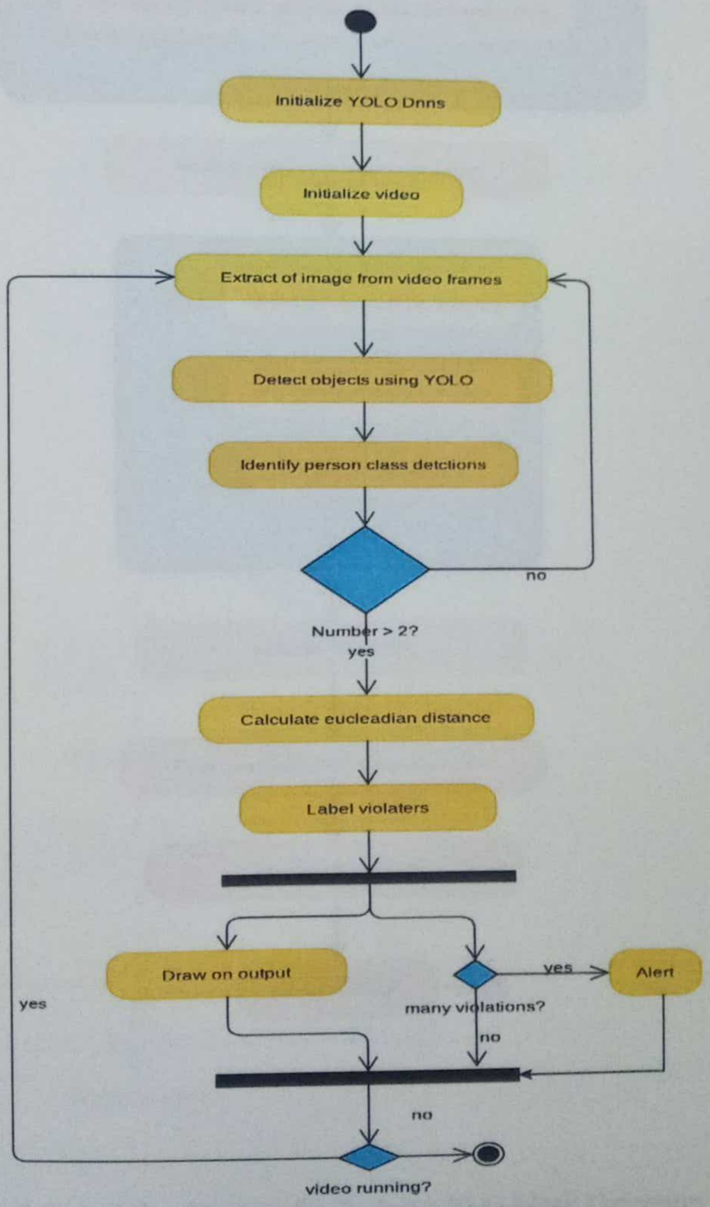


Fig 5.4: Activity Diagram for Social Distancing Detection

5.2.2.2 Activity diagram for Training Face Mask Detection:

MobileNet processes the input dataset and develops layers in the Neural Network Model which is used to classify the person in input frame whether the he/she is wearing mask or not.

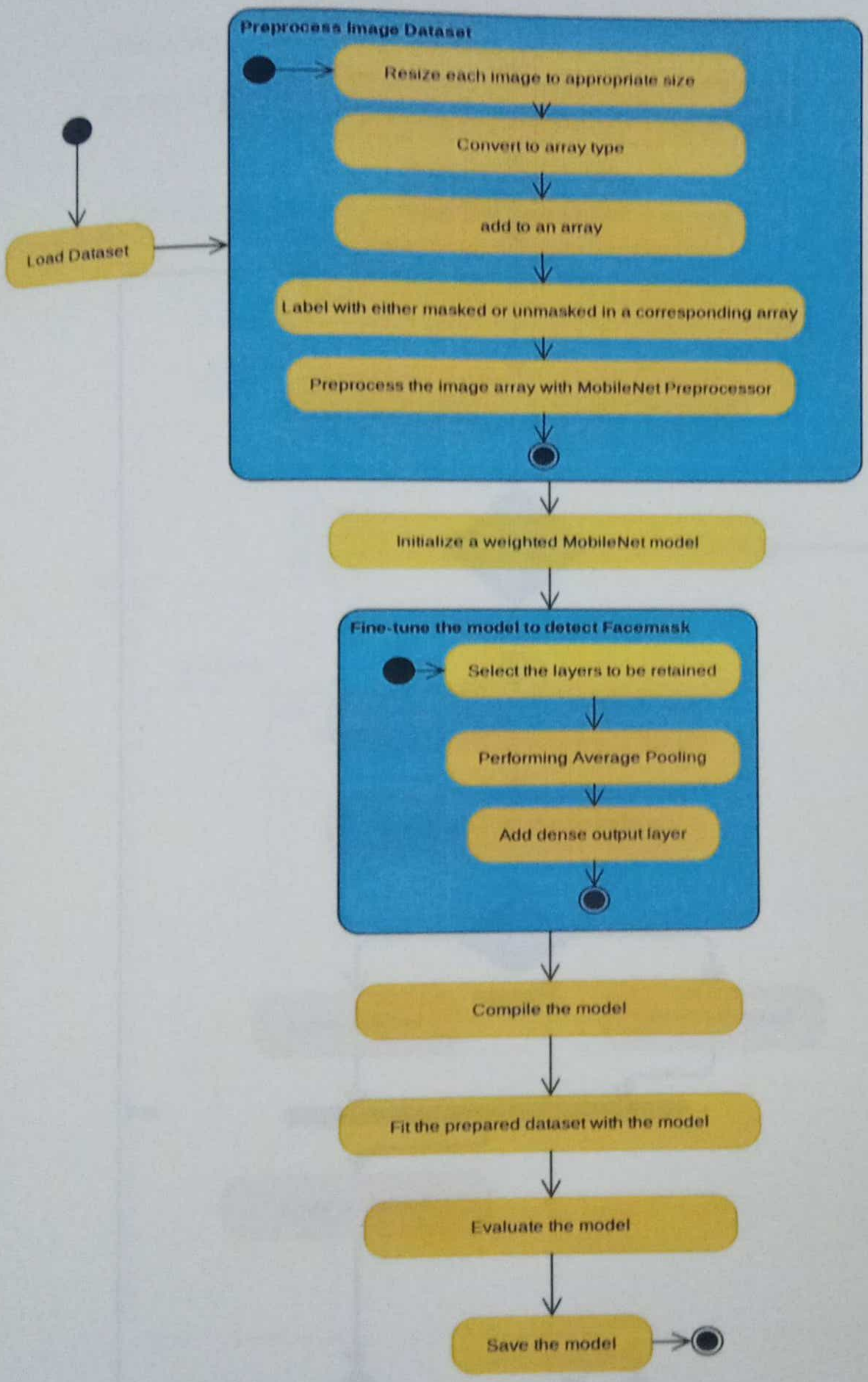
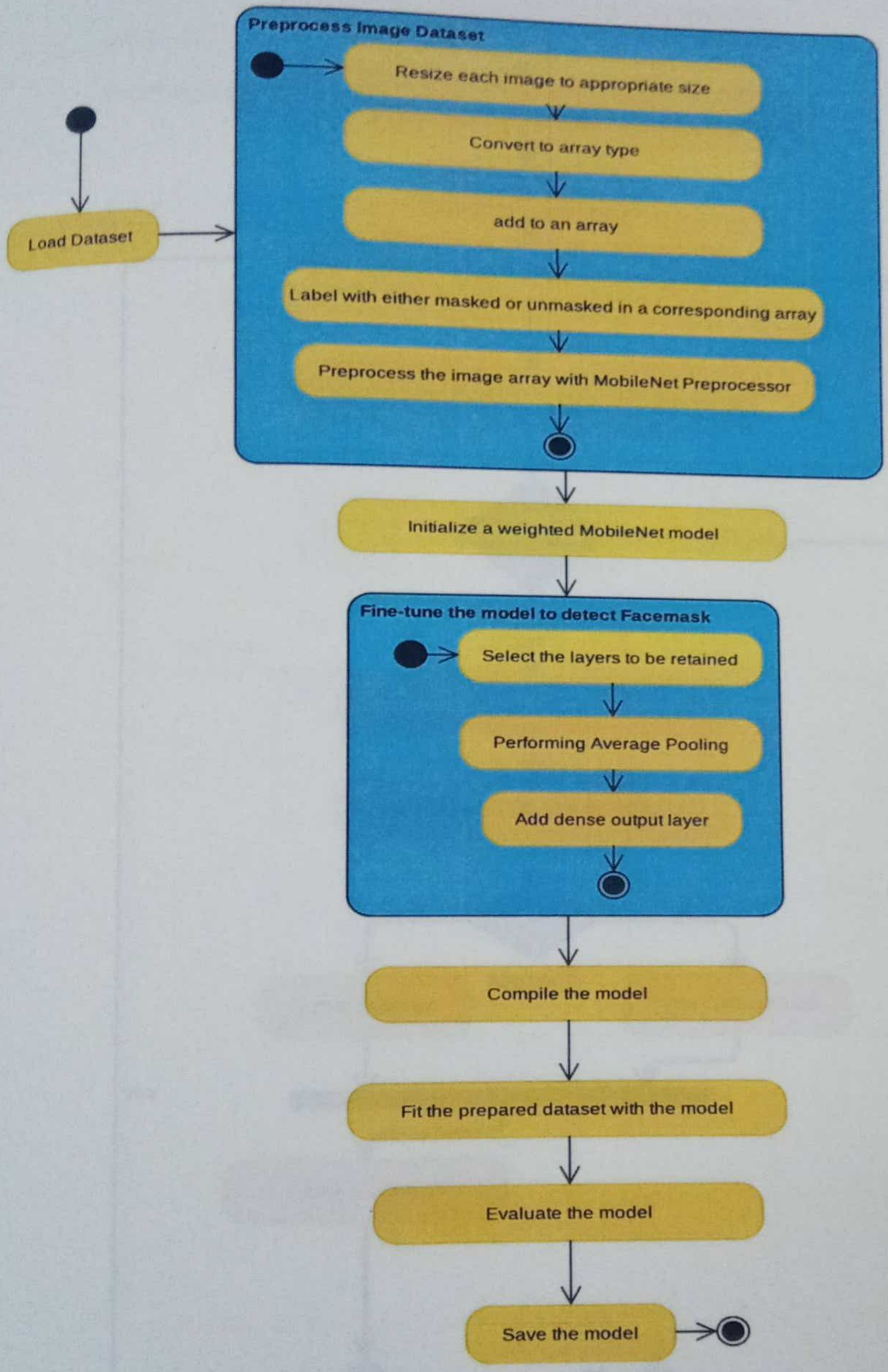

Preprocess Image Dataset
Resize each image to appropriate size
Convert to array type
add to an array
Load Dataset
Label with either masked or unmasked in a corresponding array
Preprocess the image array with MobileNet Preprocessor
Initialize a weighted MobileNet model
Fine-tune the model to detect Facemask
Select the layers to be retained
Performing Average Pooling
Add dense output layer
Compile the model
Fit the prepared dataset with the model
Evaluate the model
Save the model

Fig 5.5: Activity diagram for Training Face Mask Detection

5.2.2.2 Activity diagram for Training Face Mask Detection:

MobileNet processes the input dataset and develops layers in the Neural Network Model which is used to classify the person in input frame whether the he/she is wearing mask or not.



Load Dataset
Preprocess Image Dataset
Resize each image to appropriate size
Convert to array type
add to an array
Label with either masked or unmasked in a corresponding array
Preprocess the image array with MobileNet Preprocessor
Initialize a weighted MobileNet model
Fine-tune the model to detect Facemask
Select the layers to be retained
Performing Average Pooling
Add dense output layer
Compile the model
Fit the prepared dataset with the model
Evaluate the model
Save the model

Fig 5.5: Activity diagram for Training Face Mask Detection

5.2.2.3 Activity diagram for Face Mask Detection:

The trained model will detect the face from the frame of the input video and classifies whether the person is wearing a mask or not.

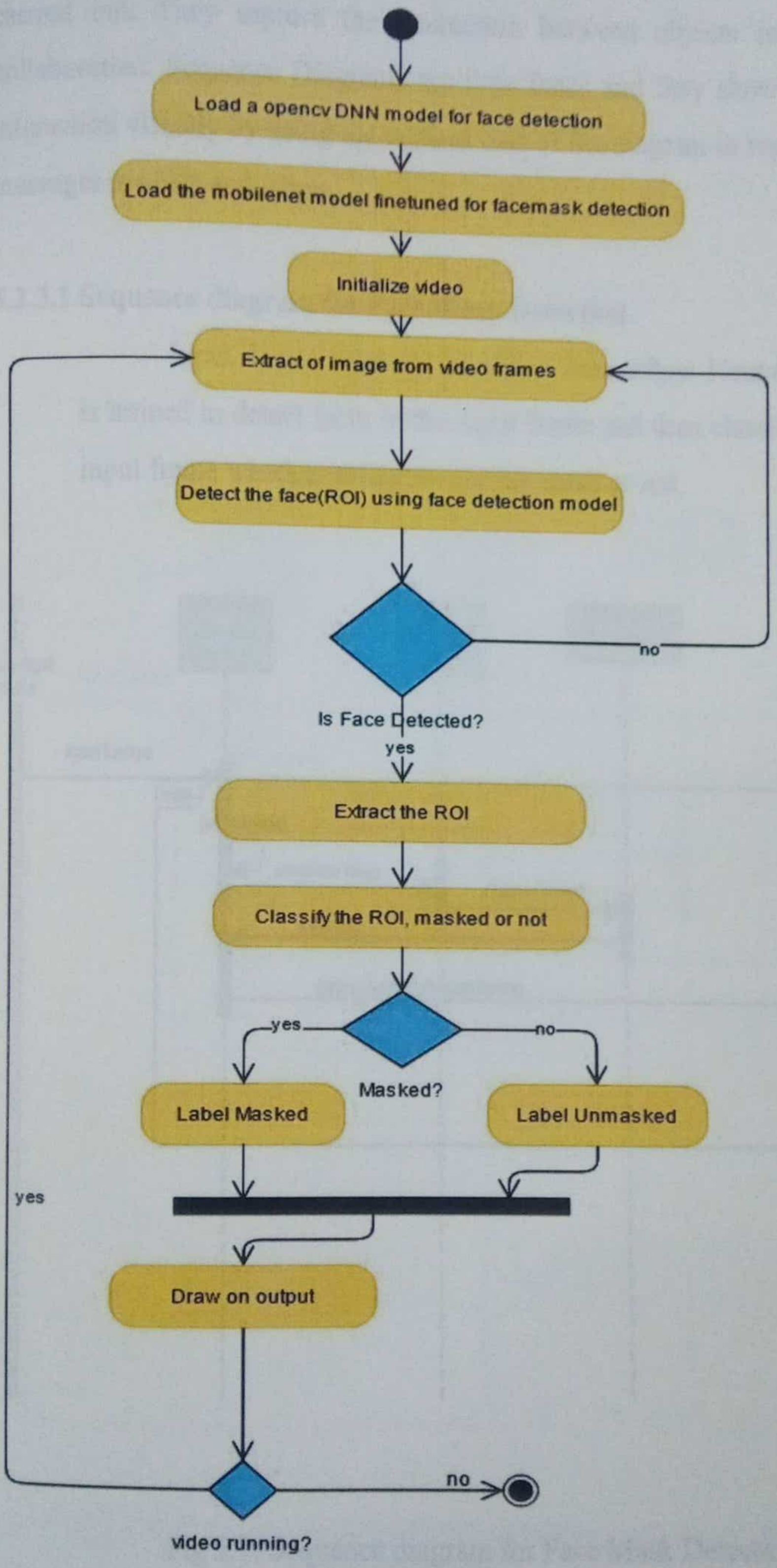

Load a opencv DNN model for face detection
Load the mobilenet model finetuned for facemask detection
Initialize video
Extract of image from video frames
Detect the face(ROI) using face detection model
no
Is Face Detected?
yes
Extract the ROI
Classify the ROI, masked or not
yes
no
Masked?
Label Masked
Label Unmasked
yes
Draw on output
no
video running?

Fig 5.6: Activity diagram for Face Mask Detection

5.2.2 SEQUENCE DIAGRAM

Sequence Diagrams are interaction diagrams that detail how operations are carried out. They capture the interaction between objects in the context of collaboration. Sequence Diagrams are time focus and they show the order of the interaction visually by using the vertical axis of the diagram to represent time what messages are sent and when.

5.2.3.1 Sequence diagram for Face Mask Detection

Input device captures the video. Tensorflow Neural Network model is trained to detect faces in the input frame and then classifies the person in input frame whether he/she is wearing mask or not.

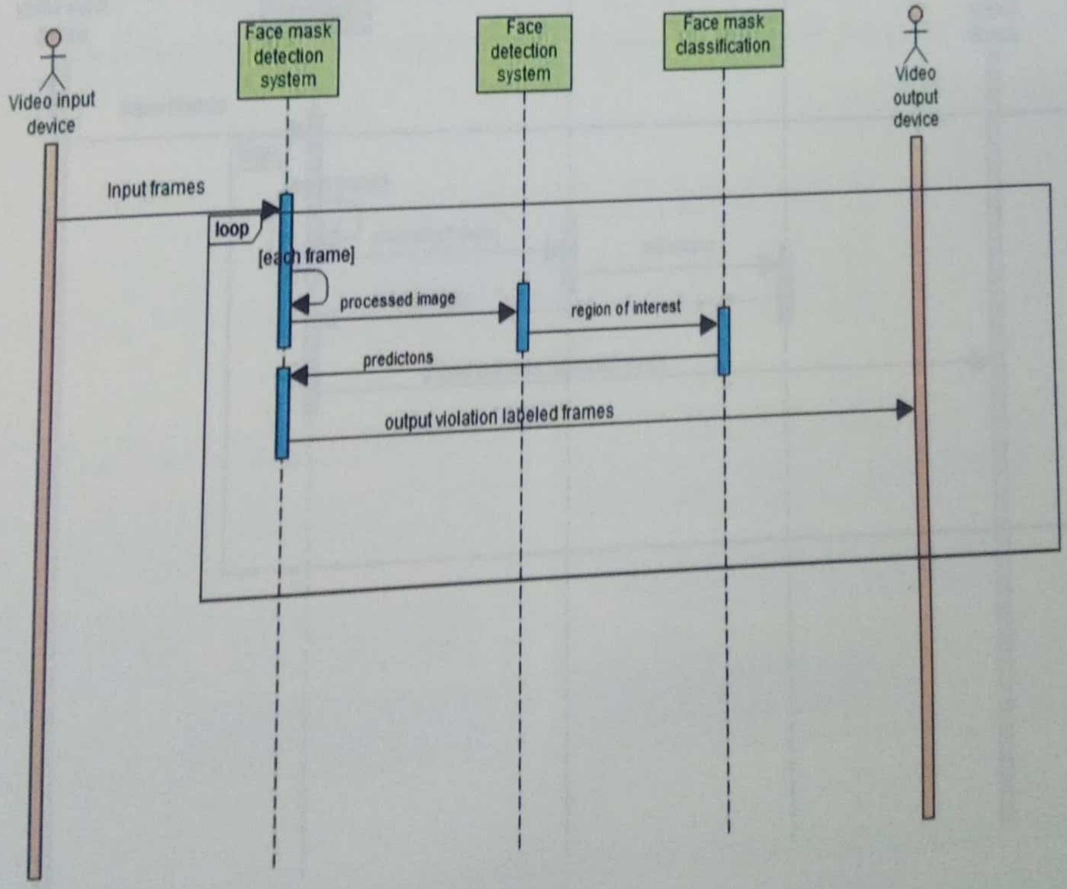


Fig 5.7: Sequence diagram for Face Mask Detection

5.2.3.2 Sequence diagram for Social Distancing Detection

Input device captures the video. YOLO object detection detects people. The model then calculates the Euclidean distance between people to identify distance between person to person and then detects the violations if the Euclidean distance between two persons equals or exceeds the given threshold.

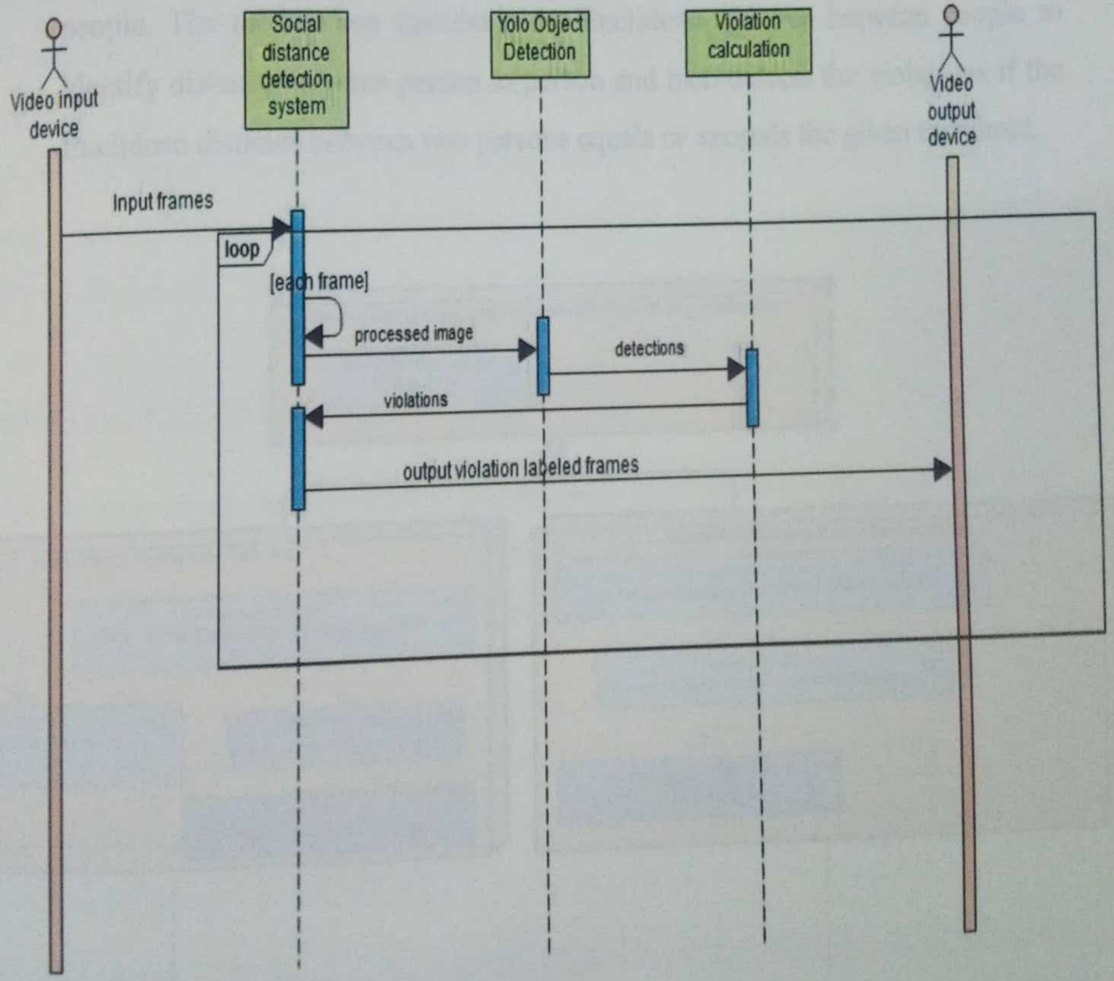


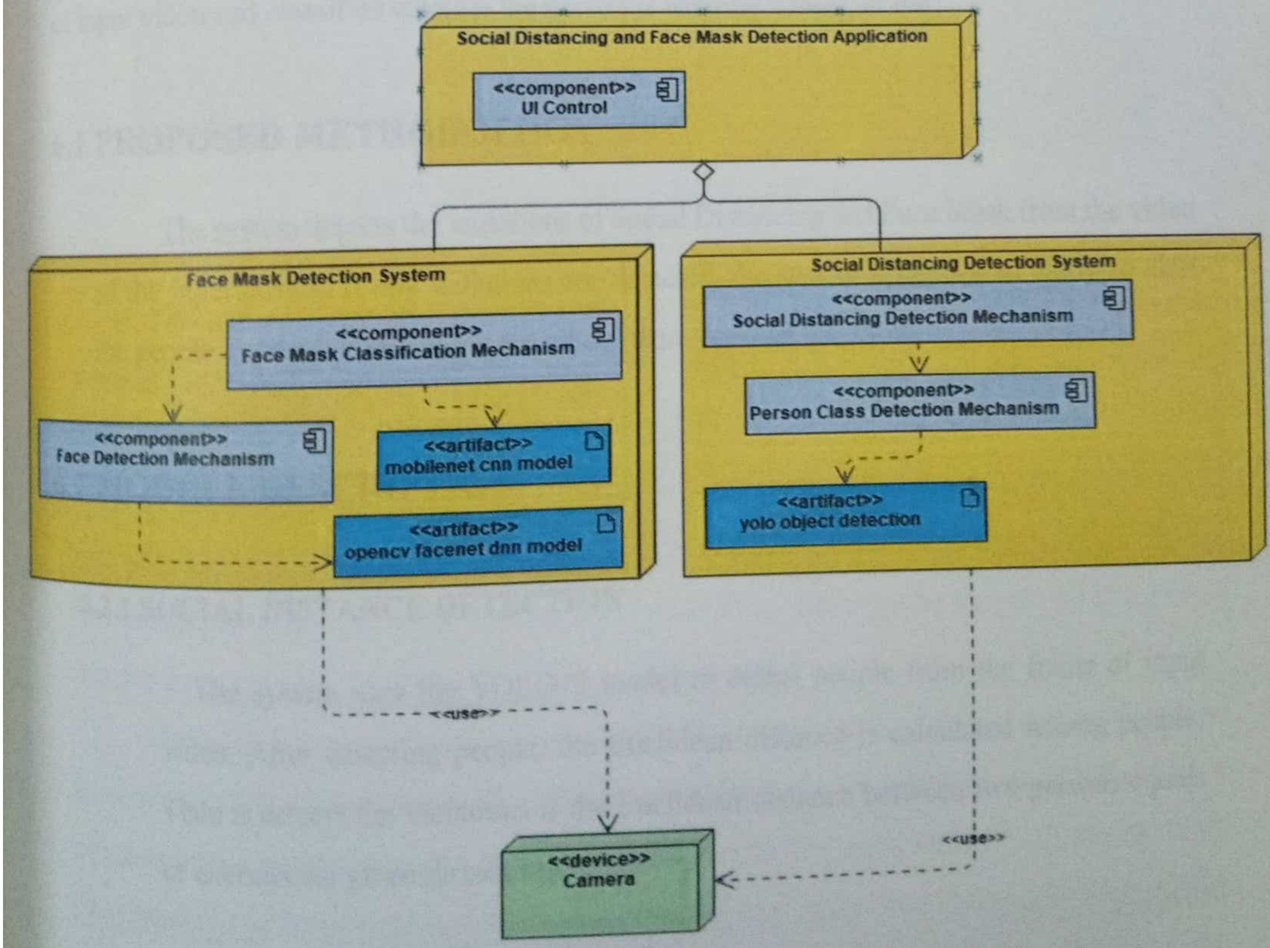
Fig 5.8: Sequence diagram for Social Distancing Detection

5.2.4. DEPLOYMENT DIAGRAM

Deployment diagram is a diagram that shows the configuration of run time processing nodes and the components that live on them. Deployment diagrams are a kind of structure diagram used in modeling the physical aspects of an object-oriented system. They are often used to model the static deployment view of a system.

5.2.4.1 Deployment Diagram for Social Distancing and Face Mask Detection

Input device captures the video. Tensorflow Neural Network model is trained to detect faces in the input frame and then classifies the person in the input frame whether he/she is wearing a mask or not. YOLO object detection detects people. The model then calculates the Euclidean distance between people to identify distance between person to person and then detects the violations if the Euclidean distance between two persons equals or exceeds the given threshold.



Social Distancing and Face Mask Detection Application
<<component>> UI Control
Face Mask Detection System
<<component>> Face Mask Classification Mechanism
<<component>> Face Detection Mechanism
<<artifact>> mobilenet cnn model
<<artifact>> opencv facenet dnn model
Social Distancing Detection System
<<component>> Social Distancing Detection Mechanism
<<component>> Person Class Detection Mechanism
<<artifact>> yolo object detection
<<use>>
<<use>>
<<device>> Camera

Fig 5.9: Deployment Diagram for Social Distancing and Face Mask Detection

CHAPTER 6

IMPLEMENTATION

The system detects the violations in the public areas and generates an alarm if violations are detected. To ensure social distancing protocol in public places and workplace, the developed social distancing detection tool that can monitor if people are keeping a safe distance from each other by analyzing real time video streams from the camera. For example: People at workplaces, factories, shops can integrate this tool to their security camera systems and can monitor whether people are keeping a safe distance from each other or not. The system uses the YOLOv3 model to detect people from the frame of input video. After detecting people, the Euclidean distance is calculated among people. Then it detects the violations if the Euclidean distance between two persons equals or exceeds the given threshold. As the input video may be taken from an arbitrary perspective view, the first step is to transform perspective of view to a bird's-eye (top-down) view.

The Neural Network model is trained using a dataset consisting of various images of people with masks and without masks. Then the trained model detects the face from the frame of input video and classifies whether the person is wearing a mask or not.

6.1 PROPOSED METHODOLOGY

The system detects the violations of Social Distancing and Face Mask from the video of the input device. If any violations are detected, the system generates an alarm to alert the people in case the violations are higher than a certain threshold.

6.2 MODULE DESCRIPTION

6.2.1 SOCIAL DISTANCE DETECTION

The system uses the YOLOv3 model to detect people from the frame of input video. After detecting people, the Euclidean distance is calculated among people. Then it detects the violations if the Euclidean distance between two persons equals or exceeds the given threshold.

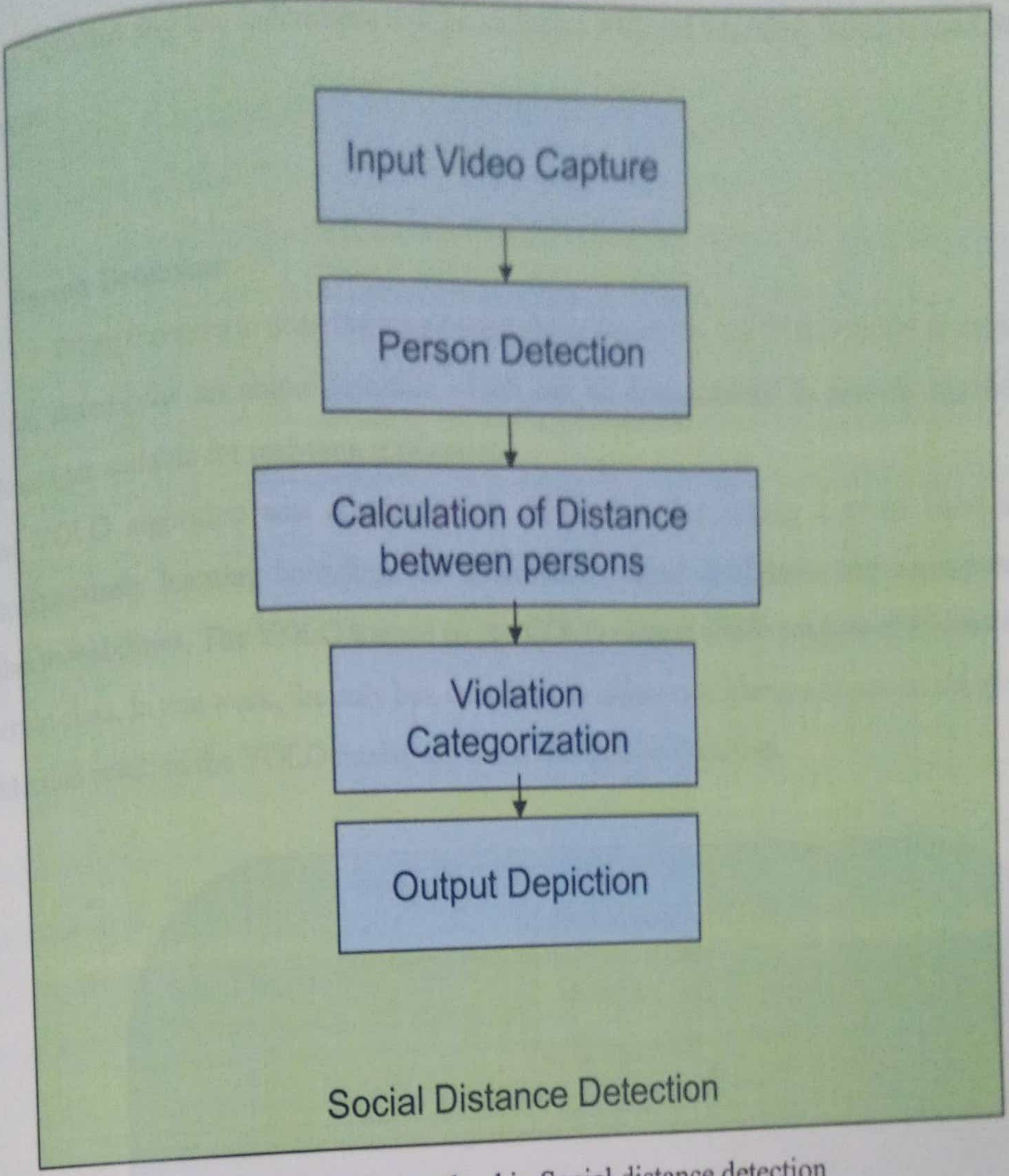


Fig 6.1: Steps involved in Social distance detection

This social distancing detection tool was developed to detect the safety distance between people in public spaces. The deep CNN method and computer vision techniques are employed in this work. Initially, an open-source object detection network based on the YOLOv3 algorithm was used to detect the objects in the video frame. From the detection result, only the person class was used and other object classes are ignored in this application. Hence, the bounding box best fits for each detected person can be drawn in the image, and this data of detected persons will be used for the distance measurement.

For camera setup, the camera is captured at fixed angle as the video frame, and the video frame was treated as a perspective view and transformed into a two-dimensional top-down view for more accurate estimation of distance measurement. In this methodology, it is assumed that the persons in the video frame are walking on the same flat plane. The location for each person can be estimated based on the top-down view. The distance between persons can be measured and scaled. Depending on the preset minimum distance, any distance less than the acceptable distance between any two individuals will be indicated with red bounding boxes around them, else green.

- **Person Detection**

When it comes to deep learning-based object detection, the YOLO model is considered one of the state-of-the art object detectors which can be demonstrated to provide significant speed advantages suitable for real-time application.

The YOLO algorithm was considered as object detection taking a given input image and simultaneously learning bounding box coordinates, object confidence and corresponding class label probabilities. The YOLO trained on the COCO dataset which consists of 80 labels including person class. In this work, the only box coordinates, object confidence and person object class from detection result in the YOLO model were used for person detection.

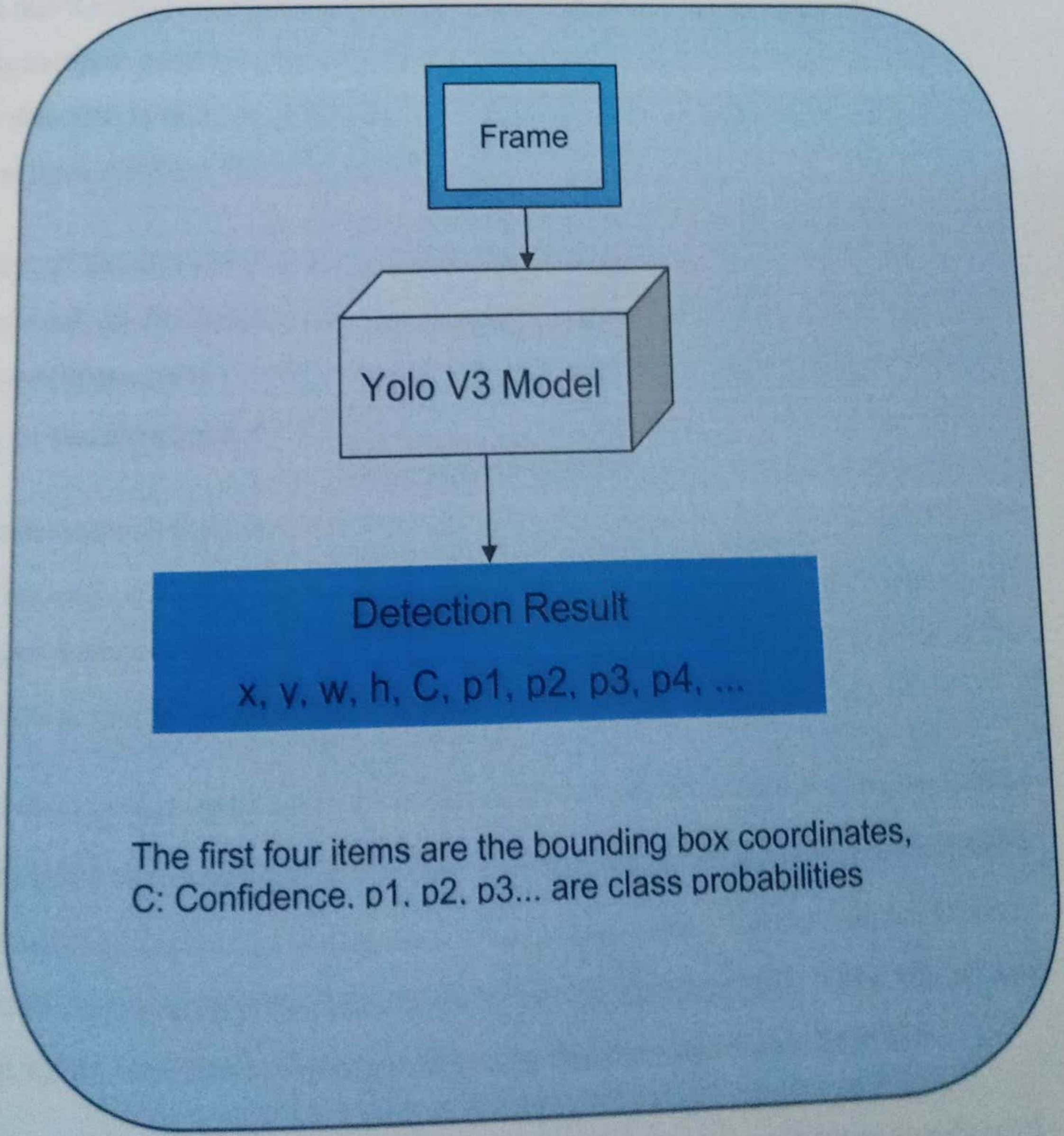

Frame
Yolo V3 Model
Detection Result
x, y, w, h, C, p1, p2, p3, p4, ...
The first four items are the bounding box coordinates,
C: Confidence, p1, p2, p3... are class probabilities

Fig 6.2: The input and output of YOLO algorithm

The region of interest (ROI) of an image focused on the person walking the street was transformed into a 2D view that contains 416×416 pixels. Camera view calibration is applied which works by computing the transformation of the perspective view into a top-down view. In OpenCV, the perspective transformation is a simple camera calibration method which involves selecting four points in the perspective view and mapping them to the corners of a rectangle in the 2D image view. Hence, every person is assumed to be standing on the same level flat plane. The actual distance between persons corresponds to the number of pixels in the top-down view can be estimated.

About YOLO:

The architecture boasts of residual skip connections, and upsampling. The most salient feature of v3 is that it makes detections at three different scales. YOLO is a fully convolutional network and its eventual output is generated by applying a 1 x 1 kernel on a feature map. In YOLO v3, the detection is done by applying 1 x 1 detection kernels on feature maps of three different sizes at three different places in the network.

The shape of the detection kernel is 1 x 1 x (B x (5 + C)). Here B is the number of bounding boxes a cell on the feature map can predict, "5" is for the 4 bounding box attributes and one object confidence, and C is the number of classes. In YOLO v3 trained on COCO, B = 3 and C = 80, so the kernel size is 1 x 1 x 255.

The first detection is made by the 82nd layer. For the first 81 layers, the image is down sampled by the network, such that the 81st layer has a stride of 32. If we have an image of 416 x 416, the resultant feature map would be of size 13 x 13. One detection is made here using the 1 x 1 detection kernel, giving us a detection feature map of 13 x 13 x 255.

Then, the feature map from layer 79 is subjected to a few convolutional layers before being up sampled by 2x to dimensions of 26 x 26. This feature map is then depth concatenated with the feature map from layer 61. Then the combined feature map is again subjected to a few 1 x 1 convolutional layers to fuse the features from the earlier layer (61). Then, the second detection is made by the 94th layer, yielding a detection feature map of 26 x 26 x 255.

A similar procedure is followed again, where the feature map from layer 91 is subjected to few convolutional layers before being depth concatenated with a feature map from layer 36.

The region of interest (ROI) of an image focused on the person walking the street was transformed into a 2D view that contains 416×416 pixels. Camera view calibration is applied which works by computing the transformation of the perspective view into a top-down view. In OpenCV, the perspective transformation is a simple camera calibration method which involves selecting four points in the perspective view and mapping them to the corners of a rectangle in the 2D image view. Hence, every person is assumed to be standing on the same level flat plane. The actual distance between persons corresponds to the number of pixels in the top-down view can be estimated.

About YOLO:

The architecture boasts of residual skip connections, and upsampling. The most salient feature of v3 is that it makes detections at three different scales. YOLO is a fully convolutional network and its eventual output is generated by applying a 1 x 1 kernel on a feature map. In YOLO v3, the detection is done by applying 1 x 1 detection kernels on feature maps of three different sizes at three different places in the network.

The shape of the detection kernel is 1 x 1 x (B x (5 + C)). Here B is the number of bounding boxes a cell on the feature map can predict, "5" is for the 4 bounding box attributes and one object confidence, and C is the number of classes. In YOLO v3 trained on COCO, B = 3 and C = 80, so the kernel size is 1 x 1 x 255.

The first detection is made by the 82nd layer. For the first 81 layers, the image is down sampled by the network, such that the 81st layer has a stride of 32. If we have an image of 416 x 416, the resultant feature map would be of size 13 x 13. One detection is made here using the 1 x 1 detection kernel, giving us a detection feature map of 13 x 13 x 255.

Then, the feature map from layer 79 is subjected to a few convolutional layers before being up sampled by 2x to dimensions of 26 x 26. This feature map is then depth concatenated with the feature map from layer 61. Then the combined feature map is again subjected to a few 1 x 1 convolutional layers to fuse the features from the earlier layer (61). Then, the second detection is made by the 94th layer, yielding a detection feature map of 26 x 26 x 255.

A similar procedure is followed again, where the feature map from layer 91 is subjected to few convolutional layers before being depth concatenated with a feature map from layer 36.

Like before, a few 1 x 1 convolutional layers follow to fuse the information from the previous layer (36). We make the final of the 3 at 106th layer, yielding a feature map of size 52 x 52 x 255.

- **Distance Calculation**

In this step of the pipeline, the location of the bounding box for each person (*x*, *y*, *w*, *h*) in the perspective view is detected. For each pedestrian, the position in the top-down view is estimated based on the bottom-center point of the bounding box. The distance between every pedestrian pair can be computed from the top-down view and the distance is scaled by the scaling factor estimated from camera view calibration.

Given the position of two persons in an image as
(x1, y1) and (x2,y2) respectively, the distance between the two persons, d can be computed as:

SquareRoot((x2-x1)^2 + (y2-y1)^2)

D = (W * F)/P

D - distance in pixels
W - Real width of an object
P - Pixel width covered by object on average

With this we can estimate the distance value to be used as cut off. The pair of persons whose distance is below the minimum acceptable distance d is marked in red, and the rest is marked in green.

6.2.2 FACE MASK DETECTION

The Neural Network model is trained using a dataset consisting of various images of people with masks and without masks. Then the trained model detects the face from the frame of the input video and classifies whether the person is wearing a mask or not.

The face mask recognition in this project is developed with a machine learning algorithm through the image classification method: MobileNetv2. The model accepts faces which are preprocessed. Hence, we must first detect the Region of Interest (that is faces) and send only that part as input to the model.

The MobileNet model is not originally trained to detect Masks on people. But we can fine tune it to detect the mask.

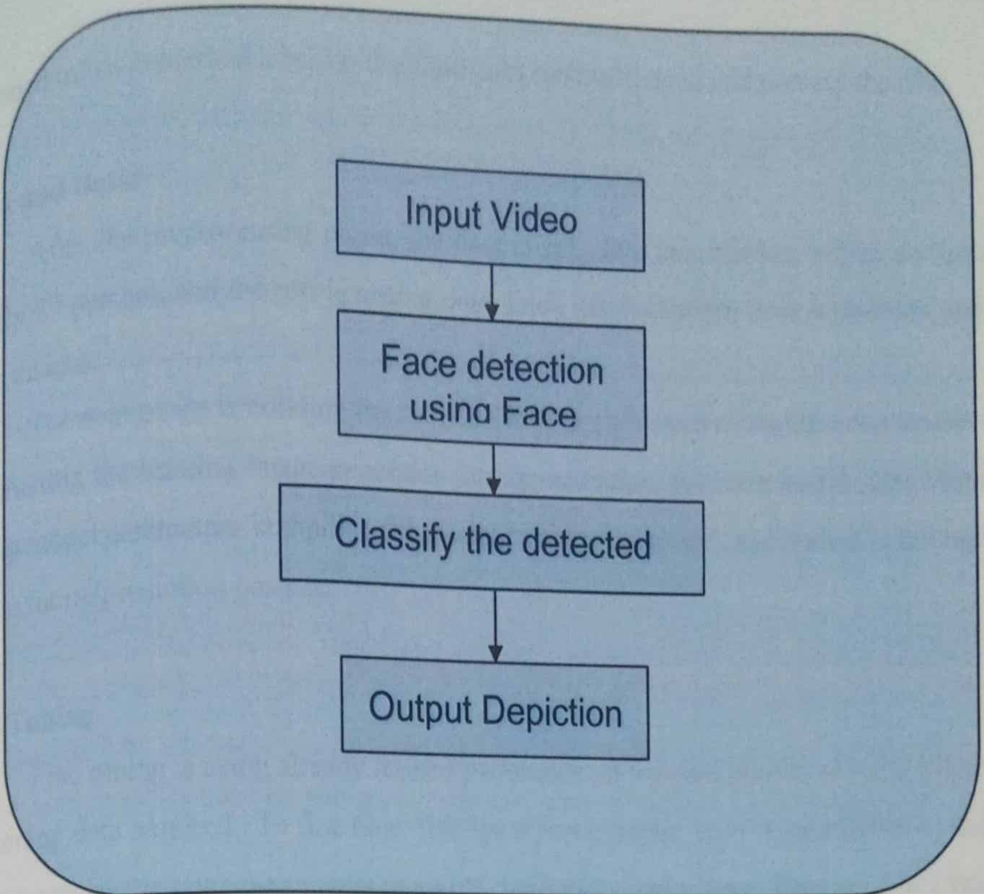


Fig:6.3: Steps involved in Face mask detection

The fine tuning of Mobile net is a very simple process, once we prepare our data to train. The data can be preprocessed as follows, but we can use the ImageGenerator of Keras directly too.

- **Preprocess**

The pre-processing phase is a phase before the training and testing of the data. There are four steps in the preprocessing which are resizing image size, converting the image to the array, pre-processing input using MobileNetV2, and the last is performing hot encoding on labels.

The resizing image is a critical pre-processing step in computer vision due to the effectiveness of training models. The smaller size of the image, the better the model will run.

The next step is to process all the images in the dataset into an array. The image is converted into the array for calling them by the loop function. After that, the image will be used to pre-process input using MobileNetV2.

And the last step in this phase is performing hot encoding on labels because many machine learning algorithms cannot operate on data labeling directly. They require all input variables and output variables to be numeric, including this algorithm. The labeled data will be transformed into a numerical label, so the algorithm can understand and process the data.

- **Train and Build**

After the preprocessing phase, the data is split into two batches, which are training data namely 75 percent, and the rest is testing data. Each batch contains both with-mask and without-mask images.

The next phase is building the model. There are six steps in building the model which are constructing the training image generator for augmentation, the base model with MobileNetV2, adding model parameters, compiling the model, training the model, and the last is saving the model for the future prediction process.

- **Fine Tuning**

Fine tuning is using already trained parameters of a model inorder to reduce the amount of training data required. To fine tune, first we select a certain layer in mobileNet to train from, so we consider the same parameters (weights) until that certain layer. Then we add a few layers in order to best represent our case.

In this project, we added a pooling and dense layer.

Pooling averages the values of a certain number of neighbouring pixels (size of kernel) and marks the average of new pixel values.

Dense layers also called a fully connected layer are connected from all neurons of previous layers.

Drop is used to fine tune in case of overfitting. It drops the neurons to be considered for output.

About MobileNet:

MobileNet is a CNN architecture model for Image Classification and Mobile Vision.There are other models as well but what makes MobileNet special that it very less computation power to run or apply transfer learning to.This makes it a perfect fit for Mobile devices,embedded systems and computers without GPU or low computational efficiency with compromising significantly with the accuracy of the results.It is also best suited for web browsers as browsers have limitation over computation,graphic processing and storage.

The core layer of MobileNet is depthwise separable filters, named as Depthwise

Convolution. The network structure is another factor to boost the

ance. Finally, the width and resolution can be tuned to trade off between

and accuracy.

- **Classify using saved model**

  The model implemented in the video. The video reads from frame to frame, then the face detection algorithm works. If a face is detected, it proceeds to the next process. From detected frames containing faces, reprocessing will be carried out including resizing the image size, converting to the array, pre-processing input using MobileNetV2.

  The next step is predicting input data from the saved model. Predict the input image that has been processed using a previously built model. Besides, the video frame will also be labeled that the person is wearing a mask or not along with the predictive percentage.

6.2.3 HOME PAGE

This is User Interface. Violations of Social Distancing and Face Mask Detection can be monitored from this page. This is done using Flask Web Framework.

In this application, it consists of three methods:

- home method
- socialDistancing method
- faceMask method

home method is used to display the content on the browser. The look and feel of the content on the browser is improved by using CSS. The User Interface consists of two containers which consists of one button each. These buttons are used to invoke the socialDistancing Method and facemask method.

Example code for Flask to display Hello World on the browser:

```
from flask import Flask
app = Flask(__name__)

@app.route("/")
def hello() -> str:
        return "Hello World"

if __name__ == "__main__":
        app.run(debug=False)
```

7.6 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. It is used for both research and production at Google. TensorFlow was developed by the Google Brain team for internal Google use. It was released under the Apache License 2.0 in 2015.

TensorFlow is Google Brain's second-generation system. Version 1.0.0 was released on February 11, 2017. While the reference implementation runs on single devices, TensorFlow can run on multiple CPUs and GPUs (with optional CUDA and SYCL extensions for general-purpose computing on graphics processing units). TensorFlow is available on 64-bit Linux, macOS, Windows, and mobile computing platforms including Android and iOS.

Its flexible architecture allows for the easy deployment of computation across a variety of platforms (CPUs, GPUs, TPUs), and from desktops to clusters of servers to mobile and edge devices. TensorFlow computations are expressed as stateful dataflow graphs. The name TensorFlow derives from the operations that such neural networks perform on multidimensional data arrays, which are referred to as tensors. During the Google I/O Conference in June 2016, Jeff Dean stated that 1,500 repositories on GitHub mentioned TensorFlow, of which only 5 were from Google.

In December 2017, developers from Google, Cisco, RedHat, CoreOS, and CaiCloud introduced Kube Flow at a conference. Kubeflow allows operation and deployment of TensorFlow on Kubernetes. In March 2018, Google announced TensorFlow.js version 1.0 for machine learning in JavaScript. In Jan 2019, Google announced TensorFlow 2.0. It became officially available in Sep 2019. In May 2019, Google announced TensorFlow Graphics for deep learning in computer graphics.

Separable Convolution. The network structure is another factor to boost the performance. Finally, the width and resolution can be tuned to trade off between latency and accuracy.

- **Classify using saved model**

The model implemented in the video. The video reads from frame to frame, then the face detection algorithm works. If a face is detected, it proceeds to the next process. From detected frames containing faces, reprocessing will be carried out including resizing the image size, converting to the array, pre-processing input using MobileNetV2.

The next step is predicting input data from the saved model. Predict the input image that has been processed using a previously built model. Besides, the video frame will also be labeled that the person is wearing a mask or not along with the predictive percentage.

6.2.3 HOME PAGE

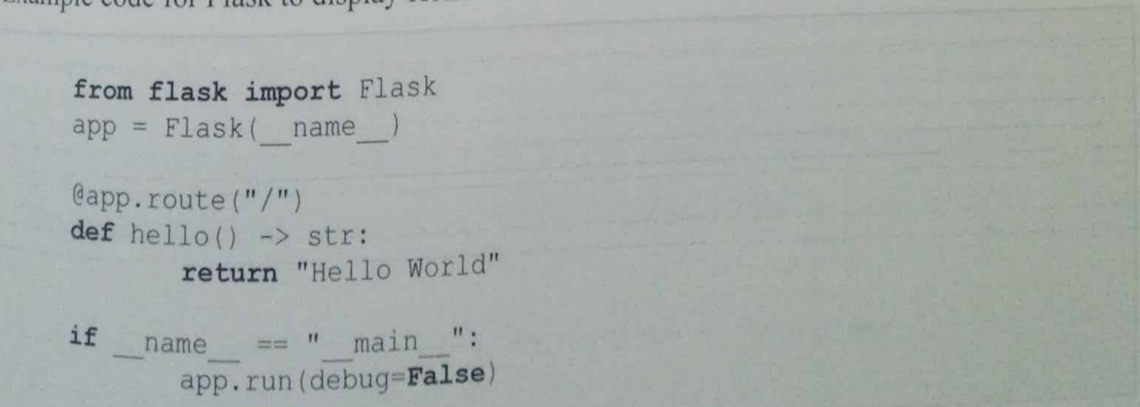
This is User Interface. Violations of Social Distancing and Face Mask Detection can be monitored from this page. This is done using Flask Web Framework.

In this application, it consists of three methods:

- home method
- socialDistancing method
- faceMask method

home method is used to display the content on the browser. The look and feel of the content on the browser is improved by using CSS. The User Interface consists of two containers which consists of one button each. These buttons are used to invoke the socialDistancing Method and facemask method.

Example code for Flask to display Hello World on the browser:



```
from flask import Flask
app = Flask(__name__)

@app.route("/")
def hello() -> str:
        return "Hello World"

if __name__ == "__main__":
        app.run(debug=False)
```

7.7 OPENCV

OpenCV (Open Source Computer Vision Library) is a library of programming functions mainly aimed at real-time computer vision. The library is cross-platform and free for use under the open-source Apache 2 License. Starting with 2011, OpenCV features GPU acceleration for real-time operations. OpenCV is written in C++ and its primary interface is in C++, but it still retains a less comprehensive though extensive older C interface. All of the new developments and algorithms appear in the C++ interface. There are bindings in Python, Java and MATLAB/OCTAVE. The API for these interfaces can be found in the online documentation. Wrappers in several programming languages have been developed to encourage adoption by a wider audience. In version 3.4, JavaScript bindings for a selected subset of OpenCV functions were released as OpenCV.js, to be used for web platforms.

If the library finds Intel's Integrated Performance Primitives on the system, it will use these proprietary optimized routines to accelerate itself. A CUDA-based GPU interface has been in progress since September 2010. An OpenCL-based GPU interface has been in progress since October 2012, documentation for version 2.4.13.3 can be found at docs.opencv.org.

OpenCV runs on the following mobile operating systems: Android, iOS, Maemo, BlackBerry 10. The user can get official releases from SourceForge or take the latest sources from GitHub. OpenCV uses CMake.

7.8 YOLOV3

You only look once (YOLO) is a state-of-the-art, real-time object detection system. On a Pascal Titan X it processes images at 30 FPS and has a map of 57.9% on COCO test-dev. YOLOv3 is extremely fast and accurate. In map measured at .5 IOU YOLOv3 is on par with Focal Loss but about 4x faster. Moreover, you can easily tradeoff between speed and accuracy simply by changing the size of the model. Object detection is a computer technology related to computer vision and image processing that deals with detecting instances of semantic objects of a certain class (such as humans, buildings, or cars) in digital images and videos. Well-researched domains of object detection include face detection and person detection. Object detection has applications in many areas of computer vision, including image retrieval and video surveillance. Every object class has its own special features that helps in classifying the class – for example all circles are round. Object class detection uses these special features. For example, when looking for circles, objects that are at a particular distance from a point (i.e. the center) are sought. Similarly, when looking for squares, objects that are perpendicular at corners and have

CHAPTER – 7

SOFTWARE ENVIRONMENT

7.1 HTML:

HTML or HyperText Mark-up Language is the standard mark-up language used to create web pages.HTML was created in 1991 by Tim Berners-Lee at CERN in Switzerland. It was designed to allow scientists to display and share their research.

HTML is written in the form of HTML elements consisting of tags enclosed in angle brackets (like <html>). HTML tags most commonly come in pairs like <h1> and </h1>, although some tags represent empty elements and so are unpaired, for example <img>. The first tag in a pair is the start tag, and the second tag is the end tag (they are also called opening tags and closing tags).

The purpose of a web browser is to read HTML documents and compose them into visible or audible web pages. The browser does not display the HTML tags, but uses the tags to interpret the content of the page. HTML describes the structure of a website semantically along with cues for presentation, making it a mark-up language rather than a programming language.

HTML elements form the building blocks of all websites. HTML allows images and objects to be embedded and can be used to create interactive forms. It provides a means to create structured documents by denoting structural semantics for text such as headings, paragraphs, lists, links, quotes and other items. It can embed scripts written in languages such as Java Script which affect the behavior of HTML web pages.

HTML is a descriptive mark-up language. Library of various mark-up languages is defined in various browsers.

equal side lengths are needed. A similar approach is used for face identification where eyes, nose, and lips can be found and features like skin color and distance between eyes can be found.

It is widely used in computer vision tasks such as image annotation, vehicle counting, activity recognition, face detection, face recognition, video object co-segmentation. It is also used in tracking objects, for example tracking a ball during a football match, tracking movement of a cricket bat, or tracking a person in a video.

You only look once, or YOLO, is one of the faster object detection algorithms out there. Though it is no longer the most accurate object detection algorithm, it is a very good choice when you need real-time detection, without loss of too much accuracy.

7.9 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. As of version 2.4, only TensorFlow is supported. Designed to enable fast experimentation with deep neural networks, it focuses on being user-friendly, modular, and extensible. It was developed as part of the research effort of project ONEIROS (Open-ended Neuro-Electronic Intelligent Robot Operating System), and its primary author and maintainer is François Chollet, a Google engineer. Chollet is also the author of the XCeption deep neural network model.

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. The code is hosted on GitHub, and community support forums include the GitHub issues page, and a Slack channel. 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 (iOS and Android), on the web, or on the Java Virtual Machine. It also allows use of distributed training of deep-learning models on clusters of Graphics processing units (GPU) and tensor processing units (TPU).

7.10 SCIPY

SciPy is a free and open-source Python library used for scientific computing and technical computing. NumPy provides some functions for linear algebra, Fourier transforms, and random number generation, but not with the generality of the equivalent functions in SciPy.

7.2 CSS:

CSS tutorial for CSS 3 tutorial provides basic and advanced concepts of CSS technology. Our CSS tutorial is developed for beginners and professionals. The major points of CSS are given below:

CSS stands for Cascading Style Sheet. CSS is used to design HTML tags.

CSS is a widely used language on the web.

HTML, CSS and JavaScript are used for web designing. It helps the web designers to apply style on HTML tags.

Cascading Style Sheets (CSS) is a style sheet language used for describing the look and formatting of a document written in a mark-up language. While most often used to style web pages and user interfaces written in HTML and XHTML, the language can be applied to any kind of XML document, including plain XML, SVG and XUL. CSS is a corner stone specification of the web and almost all webpages use CSS stylesheets to describe their presentation.

CSS is designed primarily to enable the separation of document content from document presentation, including elements such as the layout, colors, and fonts. This separation can improve content accessibility, provide more flexibility and control in the specification of presentation characteristics, enable multiple pages to share formatting, and reduce complexity and repetition in the structural content (such as by allowing for table less webdesign).

CSS can also allow the same mark-up page to be presented in different styles for different rendering methods, such as on-screen, in print, by voice (when read out by a speech-based browser or screen reader) and on Braille-based, tactile devices. It can also be used to allow the web page to display differently depending on the screen size or device on which it is being viewed. While the author of a document typically links that document to a CSS file, readers can use a different style sheet, perhaps one on their own computer, to override the one the author has specified. With plain HTML you define the colors and sizes of text and tables throughout your pages. If you want to change a certain element you will therefore have to work your way through the document and change it. With CSS you define the colors and sizes in "styles". Then as you write your documents you refer to the styles. Therefore: if you change a certain

7.2 CSS:

CSS tutorial for CSS 3 tutorial provides basic and advanced concepts of CSS technology. Our CSS tutorial is developed for beginners and professionals. The major points of CSS are given below:

CSS stands for Cascading Style Sheet. CSS is used to design HTML tags.

CSS is a widely used language on the web.

HTML, CSS and JavaScript are used for web designing. It helps the web designers to apply style on HTML tags.

Cascading Style Sheets (CSS) is a style sheet language used for describing the look and formatting of a document written in a mark-up language. While most often used to style web pages and user interfaces written in HTML and XHTML, the language can be applied to any kind of XML document, including plain XML, SVG and XUL. CSS is a corner stone specification of the web and almost all webpages use CSS stylesheets to describe their presentation.

CSS is designed primarily to enable the separation of document content from document presentation, including elements such as the layout, colors, and fonts. This separation can improve content accessibility, provide more flexibility and control in the specification of presentation characteristics, enable multiple pages to share formatting, and reduce complexity and repetition in the structural content (such as by allowing for table less webdesign).

CSS can also allow the same mark-up page to be presented in different styles for different rendering methods, such as on-screen, in print, by voice (when read out by a speech-based browser or screen reader) and on Braille-based, tactile devices. It can also be used to allow the web page to display differently depending on the screen size or device on which it is being viewed. While the author of a document typically links that document to a CSS file, readers can use a different style sheet, perhaps one on their own computer, to override the one the author has specified. With plain HTML you define the colors and sizes of text and tables throughout your pages. If you want to change a certain element you will therefore have to work your way through the document and change it. With CSS you define the colors and sizes in "styles". Then as you write your documents you refer to the styles. Therefore: if you change a certain

NumPy can also be used as an efficient multidimensional container of data with arbitrary datatypes. This allows NumPy to seamlessly and speedily integrate with a wide variety of databases. Older versions of SciPy used Numeric as an array type, which is now deprecated in favor of the newer NumPy array code. SciPy is also a family of conferences for users and developers of these tools: SciPy (in the United States), EuroSciPy (in Europe) and SciPy.in (in India). Enthought originated the SciPy conference in the United States and continues to sponsor many of the international conferences as well as host the SciPy website. The SciPy library is currently distributed under the BSD license, and its development is sponsored and supported by an open community of developers. It is also supported by NumFOCUS, a community foundation for supporting reproducible and accessible science.

In the 1990s, Python was extended to include an array type for numerical computing called Numeric (This package was eventually replaced by Travis Oliphant who wrote NumPy in 2006 as a blending of Numeric and Numarray which had been started in 2001). As of 2000, there was a growing number of extension modules and increasing interest in creating a complete environment for scientific and technical computing.

In 2001, Travis Oliphant, Eric Jones, and Pearu Peterson merged code they had written and called the resulting package SciPy. The newly created package provided a standard collection of common numerical operations on top of the Numeric array data structure. Shortly thereafter, Fernando Pérez released IPython, an enhanced interactive shell widely used in the technical computing community, and John Hunter released the first version of Matplotlib, the 2D plotting library for technical computing.

Since then the SciPy environment has continued to grow with more packages and tools for technical computing.

7.11 PYGAME

Pygame is a cross-platform set of Python modules designed for writing video games. It includes computer graphics and sound libraries designed to be used with the Python programming language. Pygame was originally written by Pete Shinners to replace PySDL after its development stalled. It has been a community project since 2000 and is released under the free software GNU Lesser General Public License. Pygame version 2 was planned as "Pygame Reloaded" in 2009, but development and maintenance of Pygame completely stopped until the end of 2016 with version 1.9.1. After the release of version 1.9.5 in March 2019, development of a new version 2 is active on the roadmap. Pygame 2.0 released on 28 October 2020, on Pygame's 20th birthday.

Pygame uses the Simple DirectMedia Layer (SDL) library,[a] with the intention of allowing real-time computer game development without the low-level mechanics of the C programming language and its derivatives. This is based on the assumption that the most expensive functions inside games can be abstracted from the game logic, making it possible to use a high-level programming language, such as Python, to structure the game.

Other features that SDL doesn't have include vector math, collision detection, 2D sprite scene graph management, MIDI support, camera manipulation, transformations, filtering, advanced freetype font support, and drawing.

Applications using Pygame can run on Android phones and tablets with the use of Pygame Subset for Android. Sound, vibration, keyboard, and accelerometer are supported on Android.

7.12 JUPYTER NOTEBOOK

Jupyter Notebook is a project and community whose goal is to "develop open-source software, open-standards, and services for interactive computing across dozens of programming languages". It was spun off from IPython in 2014 by Fernando Pérez. Project Jupyter's name is a reference to the three core programming languages supported by Jupyter, which are Julia, Python and R, and also a homage to Galileo's notebooks recording the discovery of the moons of Jupiter. Project Jupyter has developed and supported the interactive computing products Jupyter Notebook, JupyterHub, and JupyterLab.

style it will change the look of your entire site. Another big advantage is that CSS offers much more detailed attributes than plain HTML for defining the look and feel of your site.

7.3 MOBILE NET VERSION 2:

MobileNet is a type of convolutional neural network designed for mobile and embedded vision applications. They are based on a streamlined architecture that uses depthwise separable convolutions to build lightweight deep neural networks that can have low latency for mobile and embedded devices.

It is based on an inverted residual structure where the residual connections are between the bottleneck layers. The intermediate expansion layer uses lightweight depthwise convolutions to filter features as a source of non-linearity. As a whole, the architecture of MobileNetV2 contains the initial fully convolution layer with 32 filters, followed by 19 residual bottleneck layers.

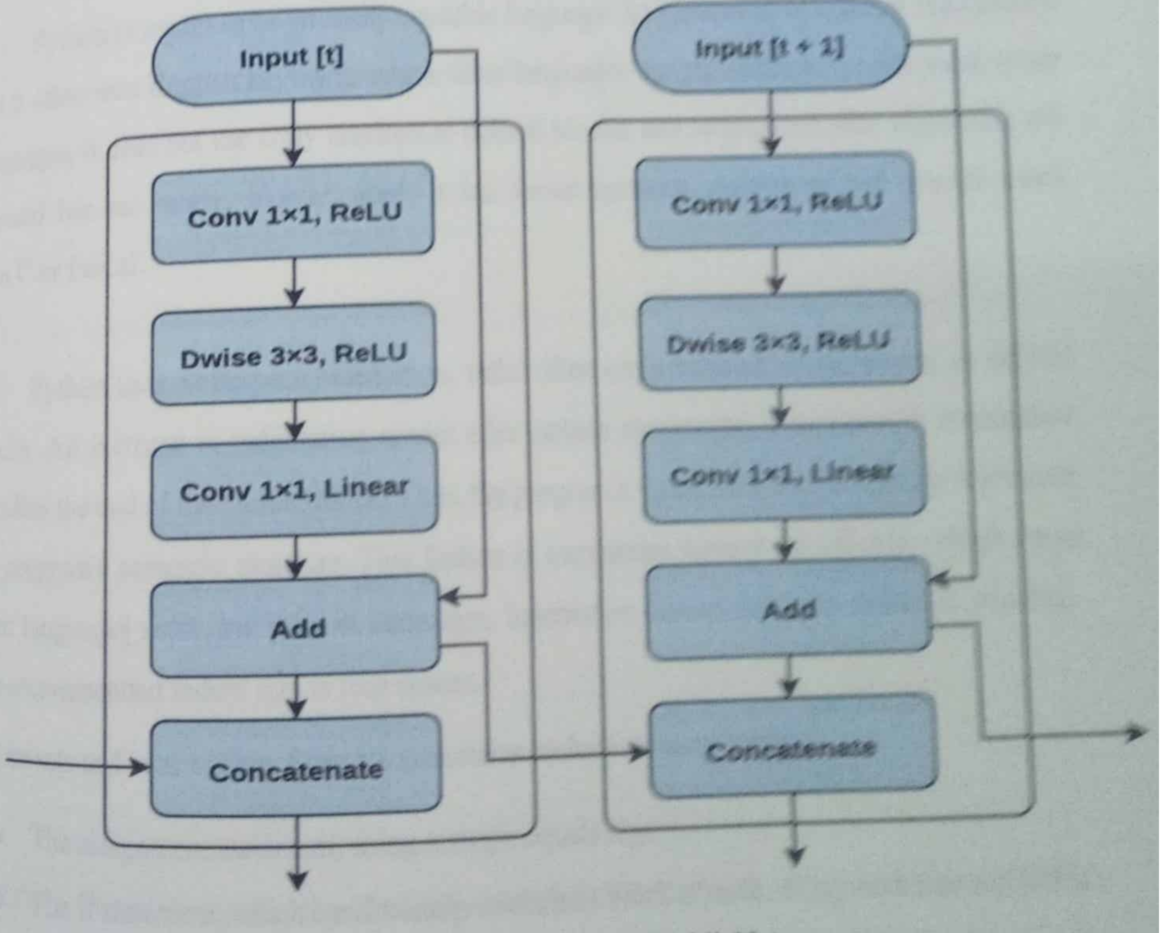


Fig 7.1: Architecture of MobileNet

In 2014, Fernando Pérez announced a spin-off project from IPython called Project Jupyter. IPython continues to exist as a Python shell and a kernel for Jupyter, while the notebook and other language-agnostic parts of IPython moved under the Jupyter name. Jupyter is language agnostic and it supports execution environments (aka kernels) in several dozen languages among which are Julia, R, Haskell, Ruby, and of course Python (via the IPython kernel).

In 2015, GitHub and the Jupyter Project announced native rendering of Jupyter notebooks file format (.ipynb files) on the GitHub platform.

Project Jupyter's operating philosophy is to support interaction with the scientific computing across all programming languages via the development of open-source software. According to the Project Jupyter website, "Jupyter will always be 100% open-source software, free for all to use and released under the liberal terms of the modified BSD license".

Jupyter Notebook is a web-based interactive computational environment for creating Jupyter notebook documents. The "notebook" term can colloquially make reference to many different entities, mainly the Jupyter web application, Jupyter Python web server, or Jupyter document format depending on context. A Jupyter Notebook document is a JSON document, following a versioned schema, containing an ordered list of input/output cells which can contain code, text (using Markdown), mathematics, plots and rich media, usually ending with the ".ipynb" extension.

A Jupyter Notebook can be converted to a number of open standard output formats (HTML, presentation slides, LaTeX, PDF, ReStructuredText, Markdown, Python) through "Download As" in the web interface, via the nbconvert library or "jupyter nbconvert" command line interface in a shell. To simplify visualisation of Jupyter notebook documents on the web, the nbconvert library is provided as a service through NbViewer which can take a URL to any publicly available notebook document, convert it to HTML on the fly and display it to the user.

Jupyter Notebook can connect to many kernels to allow programming in different languages. By default Jupyter Notebook ships with the IPython kernel. As of the 2.3 release (October 2014), it was 49 Jupyter-compatible kernels for many programming languages, including Python, R, Julia and Haskell.

The notebook interface was added to IPython in the 0.12 release (December 2011), renamed to Jupyter notebook in 2015 (IPython 4.0 – Jupyter 1.0). Jupyter Notebook is similar to the notebook interface of other programs such as Maple, Mathematica, and SageMath, a computational interface style that originated with Mathematica in the 1980s. According to The

7.4 PYTHON

Python is an interpreted, high-level and general-purpose programming language. Python's design philosophy emphasizes code readability with its notable use of significant indentation. Its language constructs and object-oriented approach aim to help programmers write clear, logical code for small and large-scale projects.

Python is dynamically-typed and garbage-collected. It supports multiple programming paradigms, including structured (particularly, procedural), object-oriented and functional programming. Python is often described as a "batteries included" language due to its comprehensive standard library.

Python interpreters are supported for mainstream operating systems and available for a few more (and in the past supported many more). A global community of programmers develops and maintains CPython, a free and open-source reference implementation.

Python is meant to be an easily readable language. Its formatting is visually uncluttered, and it often uses English keywords where other languages use punctuation. Unlike many other languages, it does not use curly brackets to delimit blocks, and semicolons after statements are allowed but are rarely, if ever, used. It has fewer syntactic exceptions and special cases than C or Pascal.

Python uses whitespace indentation, rather than curly brackets or keywords, to delimit blocks. An increase in indentation comes after certain statements; a decrease in indentation signifies the end of the current block. Thus, the program's visual structure accurately represents the program's semantic structure. This feature is sometimes termed the off-side, which some other languages share, but in most languages, indentation doesn't have any semantic meaning. The recommended indent size is four spaces.

Statements and control flow Python's statements include (among others):

- The assignment statement, using a single equals sign =.
- The if statement, which conditionally executes a block of code, along with else and elif (a contraction of else-if).

Atlantic, Jupyter interest overtook the popularity of the Mathematica notebook interface in early 2018.

CHAPTER-8

SAMPLE CODE

8.1 TRAINING.PY

```
From tensorflow.keras.preprocessing.image
import ImageDataGenerator
from tensorflow.keras.applications import MobileNetV2
from tensorflow.keras.layers import AveragePooling2D
from tensorflow.keras.layers import Dropout
from tensorflow.keras.layers import Flatten
from tensorflow.keras.layers import Dense
from tensorflow.keras.layers import Input
from tensorflow.keras.models import Model
from tensorflow.keras.optimizers import Adam
from tensorflow.keras.applications.mobilenet_v2
import preprocess_input
from tensorflow.keras.preprocessing.image
import img_to_array
from tensorflow.keras.preprocessing.image import load_img
from tensorflow.keras.utils import to_categorical
from sklearn.preprocessing import LabelBinarizer
from sklearn.model_selection import train_test_split
from sklearn.metrics import classification_report
from imutils import paths
import matplotlib.pyplot as plt
import numpy as np
import os

INIT_LR = 1e-4
```

- The for statement, which iterates over an iterable object, capturing each element to a local variable for use by the attached block.
- The while statement, which executes a block of code as long as its condition is true.
- The try statement, which allows exceptions raised in its attached code block to be caught and handled by except clauses; it also ensures that clean-up code in a finally block will always be run regardless of how the block exits.
- The raise statement, used to raise a specified exception or re-raise a caught exception.
- The class statement, which executes a block of code and attaches its local namespace to a class, for use in object-oriented programming.
- The def statement, which defines a method.
- The with statement, which encloses a code block within a context manager (for example, acquiring a lock before the block of code is run and releasing the lock afterwards, or opening a file and then closing it), allowing resource-acquisition-is-initialization (RAII)-like behavior and replaces a common try/finally idiom.
- The break statement, exits from a loop.
- The continue statement, skips this iteration and continues with the next item.
- The del statement, removes a variable, which means the reference from the name to the value is deleted and trying to use that variable will cause an error. A deleted variable can be reassigned.
- The pass statement, which serves as a NOP. It is syntactically needed to create an empty code block.
- The assert statement, used during debugging to check for conditions that should apply.
- The yield statement, which returns a value from a generator function and yield is also an operator. This form is used to implement coroutines.
- The return statement, used to return a value from a function.
- The import statement, which is used to import modules whose functions or variables can be used in the current program.

7.5 FLASK

Flask is flexible. It doesn't require you to use any particular project or code layout. However, when first starting, it's helpful to use a more structured approach. This means that the tutorial will require a bit of boilerplate up front, but it's done to avoid many common pitfalls that new developers encounter, and it creates a project that's easy to expand on. Once you become more comfortable with Flask, you can step out of this structure and take full advantage of Flask's flexibility.

Flask is a micro web framework written in Python. It is classified as a micro framework because it does not require particular tools or libraries. It has no database abstraction layer, form validation, or any other components where pre-existing third-party libraries provide common functions. However, Flask supports extensions that can add application features as if they were implemented in Flask itself. Extensions exist for object-relational mappers, form validation, upload handling, various open authentication technologies and several common framework related tools.

Features of Flask include:

- Development server and debugger
- Integrated support for unit testing
- RESTful request dispatching
- Uses Jinja templating
- Support for secure cookies (client-side sessions)
- 100% WSGI 1.0 compliant
- Unicode-based
- Extensive documentation
- Google App Engine compatibility
- Extensions available to enhance features desired

Flask was created by Armin Ronacher of Pocoo, an international group of Python enthusiasts formed in 2004. According to Ronacher, the idea was originally an April Fool's joke that was popular enough to make into a serious application. The name is a play on the earlier Bottle framework. When Ronacher and Georg Brandl created a bulletin board system written in Python, the Pocoo projects Werkzeug and Jinja were developed. In April 2016, the Pocoo team was disbanded and development of Flask and related libraries passed to the newly formed Pallets project. Flask has become popular among Python enthusiasts. As of October 2020, it has second most stars on GitHub among Python web-development frameworks, only slightly behind Django, and was voted the most popular web framework in the Python Developers Survey 2018.

```
EPOCHS = 1
BS = 32
DIRECTORY = r"C:\Users\jitin\OneDrive\Desktop\facemask\Face-Mask-Detection\dataset"
CATEGORIES = ["with_mask", "without_mask"]
print("loading images...")
data = []
labels = []
for category in CATEGORIES:
  path = os.path.join(DIRECTORY, category)
  for img in os.listdir(path):
      img_path = os.path.join(path, img)
      image = load_img(img_path, target_size=(224, 224))
      image = img_to_array(image)
      image = preprocess_input(image)

      data.append(image)
      labels.append(category)

lb = LabelBinarizer()
labels = lb.fit_transform(labels)
labels = to_categorical(labels)

data = np.array(data, dtype="float32")
labels = np.array(labels)

(trainX, testX, trainY, testY) = train_test_split(data, labels,test_size=0.20, stratify=labels, random_state=42)
```

```
aug = ImageDataGenerator(
    rotation_range=20,
    zoom_range=0.15,
    width_shift_range=0.2,
    height_shift_range=0.2,
    shear_range=0.15,
    horizontal_flip=True,
    fill_mode="nearest")



baseModel = MobileNetV2(weights="imagenet", include_top=False,
input_tensor=Input(shape=(224, 224, 3)))


headModel = baseModel.output
headModel = AveragePooling2D(pool_size=(7, 7))(headModel)
headModel = Flatten(name="flatten")(headModel)
headModel = Dense(128, activation="relu")(headModel)
headModel = Dropout(0.5)(headModel)
headModel = Dense(2, activation="softmax")(headModel)
model = Model(inputs=baseModel.input, outputs=headModel)
```

8.2 DETECTION.PY

```
    def detect_people(frame, net, ln, personIdx=0):
            (H, W) = frame.shape[:2]
            results = []
            blob = cv2.dnn.blobFromImage(frame, 1 / 255.0, (416, 416),
            swapRB=True, crop=False)
```

```
        net.setInput(blob)
        layerOutputs = net.forward(ln)
        boxes = []
        centroids = []
        confidences = []
        for output in layerOutputs:
                for detection in output:
                        scores = detection[5:]
                        confidence=scores[classID]
                if classID == personIdx and confidence > MIN_CONF:
                        box = detection[0:4] * np.array([W, H, W, H])
                        (centerX, centerY, width, height) = box.astype("int")
                        x = int(centerX - (width / 2))
                        y = int(centerY - (height / 2))
                        boxes.append([x, y, int(width), int(height)])
                        centroids.append((centerX, centerY))

    if len(idxs) > 0:
        for i in idxs.flatten():
                (w, h) = (boxes[i][2], boxes[i][3])
                r = (confidences[i], (x, y, x + w, y + h), centroids[i])
                results.append(r)
    return results
```

CHAPTER-9

SYSTEM TESTING

The main use of testing is to find out errors. Testing is the way toward attempting to find each possible flaw or shortcoming in a work item. It gives a way to deal with check the helpfulness of parts, sub-assemblies, and social occasions just as a finished thing it is the path toward working on programming with the point of ensuring that the Software system satisfies its necessities and customer wants and does not bomb in an unacceptable manner. There are various sorts of test. Each test type keeps an eye on a specific testing need. Testing permits to expel the mistakes and improve the framework execution. There are numerous kinds of tests which enables us to improve our venture execution and to make it mistake free. What's more we likewise have tests which encourage us to check singular modules autonomously and furthermore to check complete framework together according to our convenience.

9.1 TYPES OF TESTS:

9.1.1 UNIT TESTING:

Unit testing incorporates the arrangement of analyses that favor that within program basis is working properly, and that program information sources produce significant yields. It checks whether little segments are working appropriately or not. Every single decision branch and inside code stream should be endorsed. It is the attempting of individual programming units of the application .it is done after the completion of an individual unit before fuse. This is an auxiliary attempting, that relies upon learning of its improvement and is prominent. Unit tests perform fundamental tests at section level and test a specific business system, application, or possibly structure plan. Unit tests ensure that all of a thoughtful method for a business technique performs unequivocally to the recorded points of interest and contains obviously portrayed data sources and fore seen results.

A unit test encourages you to discover which part is broken in your application and fixes it quicker.

9.1.2 INTEGRATION TESTING:

Integration tests are expected to test joined programming modules to choose whether they everything considered continue running as one program. Testing is an event driven an dynamically stressed over the crucial after effect of screens or fields. Combination tests show that in spite of the way that the sections were autonomously satisfied, as showed up by successfully unit testing, the gathering of portions are correct and unsurprising. Combination testing is expressly away for revealing the issues that rise up out of the gathering of these portions.

Integration testing permits to discover blunders because of unexpected communication between the framework and the sub-framework segments. We test the product in order to test and to identify all the potential mistakes in our undertaking once we complete the source code and before conveying it to the clients.

The techniques for performing tests. These techniques provide guidance for testing:

- To test the internal logic of the software components.
- To test the input and output domains of a programs and to uncover the errors in program function, behavior and performance.

We can test the software by using two methods:

- White Box testing: In this the internal logic program is being checked by using different test case design techniques.

- Black Box testing: In this the software requirements are tested by using different test case design techniques.

9.1.3 PERFORMANCE TESTING:

This test is done to find the run-time performance of the software with the context of the integrated system. These tests can be carried out throughout the testing process. For example, the performance of individual module is accessed during white box testing under unit testing.

9.1.4 VERIFICATION AND VALIDATION:

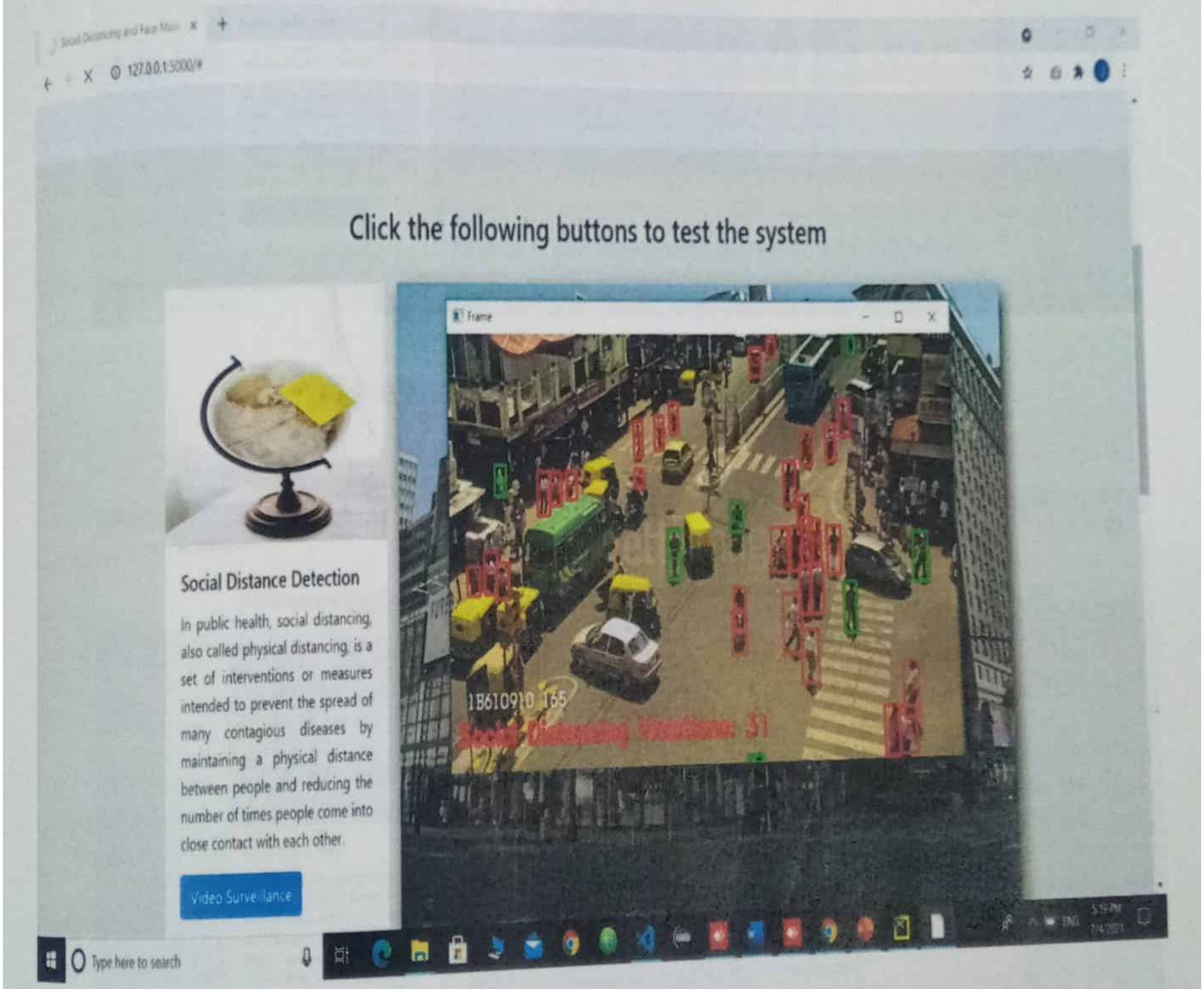
Testing procedure is a piece of subject alluding to checking and approval of our task. We have to find the framework determinations and we should attempt to meet the details of the client and to fulfill the client, for this reason, we need to check and approve the item and we have to ensure that everything is working appropriately. Check and approval are the two unique things. One is performed to guarantee that the product is working accurately and to implement a particular usefulness and the other is done to guarantee if the client prerequisites are appropriately met or not by the finished result.

CHAPTER-10

RESULTS AND OUTPUT SCREENS

10.1 SOCIAL DISTANCING DETECTION

Social distancing detection detects the social distancing violations and generates and alarm at the site where violations is detected to alert the people.



Click the following buttons to test the system

Social Distance Detection

In public health, social distancing, also called physical distancing, is a set of interventions or measures intended to prevent the spread of many contagious diseases by maintaining a physical distance between people and reducing the number of times people come into close contact with each other.

Video Surveillance

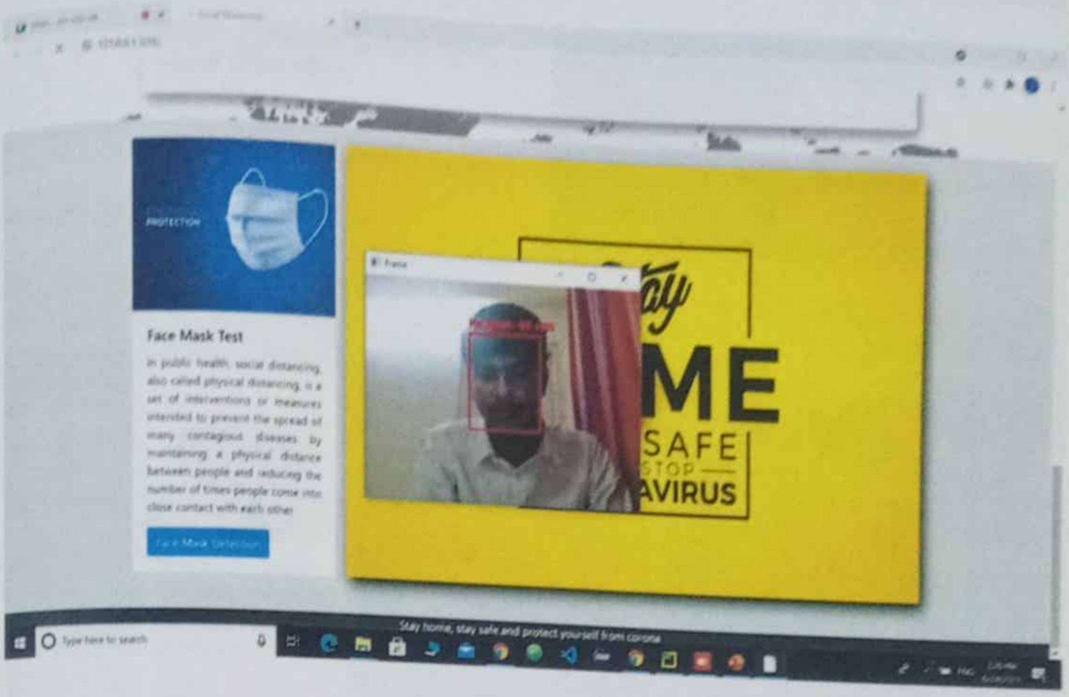
1B610910 165

Social Distancing Violations: 31

Fig 10.1: Social Distancing Detection.

10.2 FACE MASK DETECTION

Face Mask Detection detects whether the person in the input video frame is wearing a face mask or not.



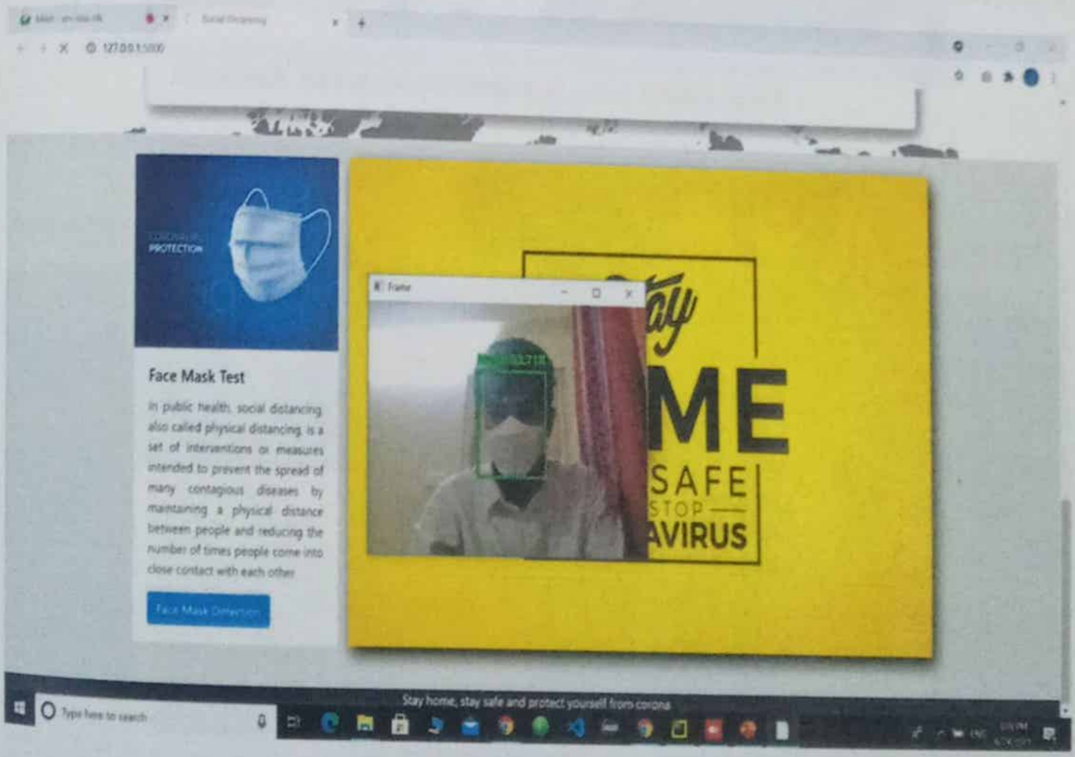
PROTECTION

Face Mask Test

In public health, social distancing, also called physical distancing, is a set of interventions or measures intended to prevent the spread of many contagious diseases by maintaining a physical distance between people and reducing the number of times people come into close contact with each other

Face Mask Detection

Fig 10.2: Face Mask Detection without Mask



CORONAVIRUS PROTECTION

Face Mask Test

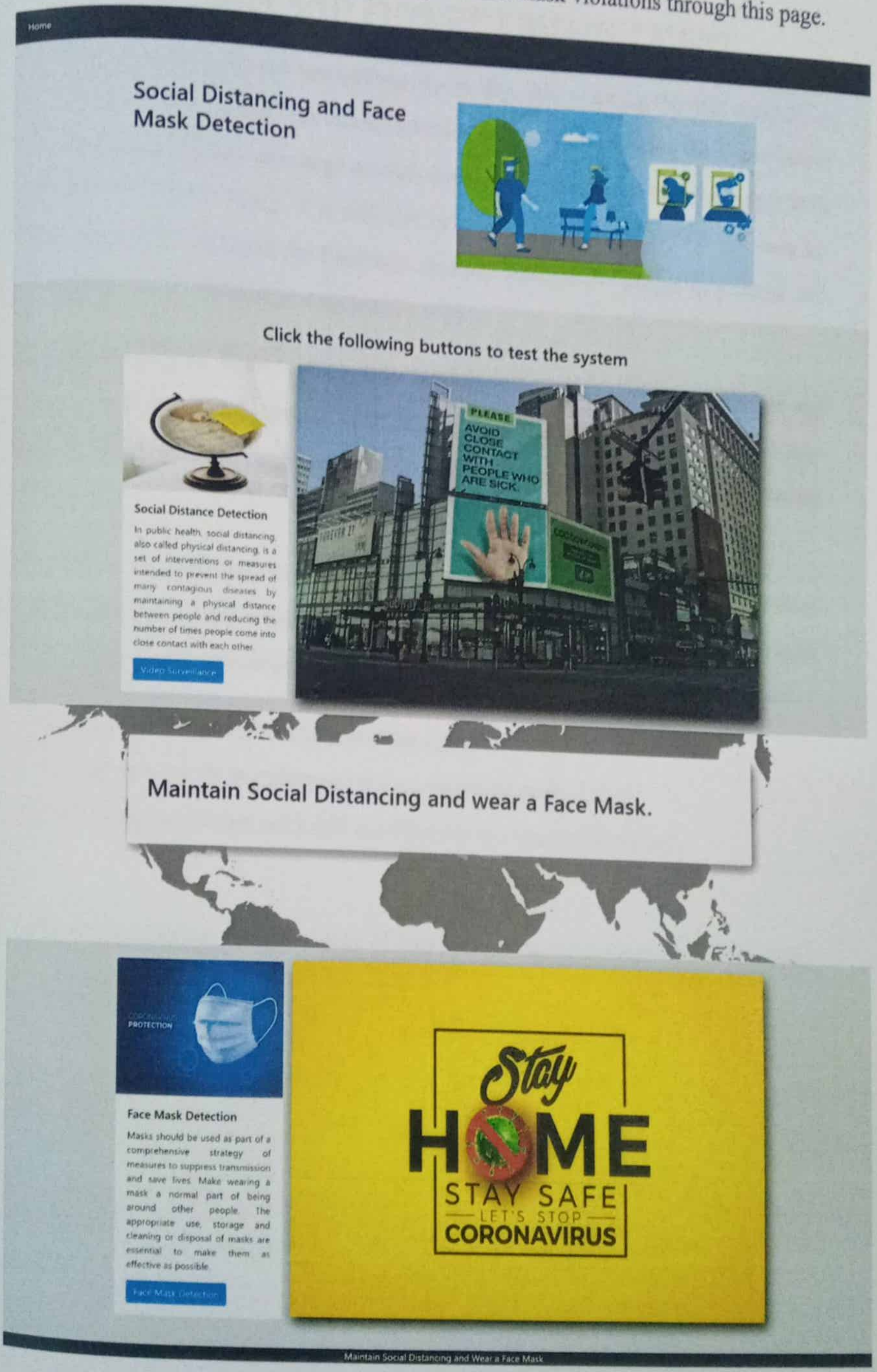
In public health, social distancing, also called physical distancing, is a set of interventions or measures intended to prevent the spread of many contagious diseases by maintaining a physical distance between people and reducing the number of times people come into close contact with each other

Face Mask Detection

Fig 10.3: Face Mask Detection with Mask

10.3 HOMEPAGE

The admin can monitor the social distancing and face mask violations through this page.



Social Distancing and Face Mask Detection

Click the following buttons to test the system

Social Distance Detection

In public health, social distancing, also called physical distancing, is a set of interventions or measures intended to prevent the spread of many contagious diseases by maintaining a physical distance between people and reducing the number of times people come into close contact with each other

Maintain Social Distancing and wear a Face Mask.

Face Mask Detection

Masks should be used as part of a comprehensive strategy of measures to suppress transmission and save lives. Make wearing a mask a normal part of being around other people. The appropriate use, storage and cleaning or disposal of masks are essential to make them as effective as possible.

Fig: 10.4: Homepage

CHAPTER-11

CONCLUSION AND FUTURE ENHANCEMENT

Technology is introducing new innovations day by day, thus reducing the time required to do things. Social Distancing and Face Mask Detection is helpful in reducing the transmission of virus. The model is trained with large datasets consisting of images of various people with masks and without masks. Hence, it is effective in identifying a person who is not wearing mask. The application calculates the Euclidean distance between the people in a frame and identifies whether social distance is maintained or not.

This application can be further enhanced by optimizing the person detection algorithm and calculating distance between the people more effectively, precisely and accurately. It can also be integrated with human body temperature detection so that more accurate results can be obtained.

It gave us the requisite practical knowledge to supplement the already taught theoretical concepts thus making us more competent as a computer engineer. The project from a personal point of view also helped us in understanding the following aspects of project development:

- The planning that goes into implementing a project.
- The importance of proper planning and an organized methodology.
- The key element of team spirit and coordination in a successful project.

The project also provided us the opportunity of interacting with our teachers and to gain from their best experience

REFERENCES

[1] https://getbootstrap.com/docs/5.0/getting-started/introduction/

[2] https://www.tensorflow.org/tutorials

[3] https://docs.opencv.org/4.5.2/d9/df8/tutorial_root.html

[4] https://flask.palletsprojects.com/en/2.0.x/tutorial/index.html