# Face Mask Detection and Social Distancing Tracker using Computer Vision and Deep Learning

*A Project Report*

*Submitted in the partial fulfillment of the requirement for the award of the Degree*

**BACHELOR OF ENGINEERING**

**IN**

**ELECTRONICS AND COMMUNICATION ENGG.**

BY


**SAURAV KUMAR AGARWAL (BE/10070/17)**

**VIVEK BASANT (BE/10048/17)**

**SAGAR KUMAR SAHU (BE/10031/17)**

**DEPT. OF ELECTRONICS AND COMMUNICATION ENGG.**

**BIRLA INSTITUTE OF TECHNOLOGY**

**MESRA-835215, RANCHI**

# DECLARATION CERTIFICATE

This is to certify that the work presented in the project entitled "**Face Mask Detection and Social Distancing Tracker using Computer Vision and Deep Learning**" in partial fulfillment of the requirement for the award of Degree of **Bachelor of Engineering in Electronics and Communication Engineering** from Birla Institute of Technology, Mesra, Ranchi is an authentic work carried out under my supervision and guidance.

To the best of my knowledge, the content of this project report does not form a basis for the award of any previous Degree to anyone else.

Date: 09 May, 2021

9.5.2021

Dr. Sandeep Singh Solanki
Dept. of ECE
Birla Institute of Technology
Mesra, Ranchi-835215

9.5.2021

Head
Dept. of ECE
Birla Institute of Technology
Ranchi-835215

Dean
(Under Graduate Studies)
Birla Institute of Technology Mesra,
Mesra, Ranchi-835215

# CERTIFICATE OF APPROVAL

The foregoing project entitled "**Face Mask Detection and Social Distancing Tracker using Computer Vision and Deep Learning**", is hereby approved as a creditable study of research topic and has been presented in satisfactory manner to warrant its acceptance as prerequisite to the degree for which it has been submitted.

It is understood that by this approval, the undersigned do not necessarily endorse any conclusion drawn or opinion expressed therein, but approve the project report for the purpose for which it is submitted.

9.5.2021

**(Internal Examiner)** **(External Examiner)**

**(Chairman)**

**Head of the Department**

# ACKNOWLEDGEMENT

We owe our deepest gratitude to our guide **DR. SANDEEP SINGH SOLANKI**, who helped us throughout the project. He made sure that we learn by practice, and always kept motivating us to think a step further, work a little harder. Our interactions with him always resulted in newer ideas and proved beneficial towards our work. Without his constant presence and supervision our work would not have been successful.

He has taught us how to approach innovative research topics keeping novelty and quality in mind and has given us a lot of insights into this domain of work. Working under his guidance has been an enriching experience which will benefit us greatly in the future.

This project is dedicated to our parents and families for keeping faith in us. Without their support none of this would have been ever possible.

Sagar Kumar Sahu

Vivek Basant

Saurav Kumar Agarwal

# ABSTRACT

All of us are aware with the current COVID-19 pandemic which has bring the world into a halt. Two main methods to prevent COVID-19 are Face Mask and Social Distancing. But since unlock has begun, people have become more careless about the disease. Many can be seen without masks and no social distancing is observed at various places like markets. So, we need to take action against these people by identifying them. This project is comprised of two parts- Detecting face mask and Social Distancing Tracker. For face mask detection, we are using Convolutional Neural Network, which is a part of Deep Learning and for social distancing detector, object detection along with distance measurement algorithm is being used. This project can have large application in areas such as malls, offices, etc., when it is embedded in a microcontroller such as Raspberry Pi. It can be very useful in prevention of this deadly disease which has killed lakhs of people in the entire world.

# CONTENTS

# LIST OF FIGURES

# CHAPTER -1

## 1.1 Background

Many people now-a-days do not take the COVID-19 pandemic seriously and do not wear face mask and do not maintain social distance, especially in malls and market places. Since, it is not possible for the organization to monitor every person, this task can be automated by the help of this project. A study has shown that face masks and social distance can reduce the risk of being infected by COVID-19 to 70%. This makes monitoring of people even more important.

## 1.2 Motivation

Computer vision can be described as finding and telling features from images to help discriminate objects or classes of objects. What computers see are just numbers as can be seen from the Figure.1.1, so we can say that computer vision is mainly the processing of these numbers and extracting useful information from these numbers for various operations such as object recognition.

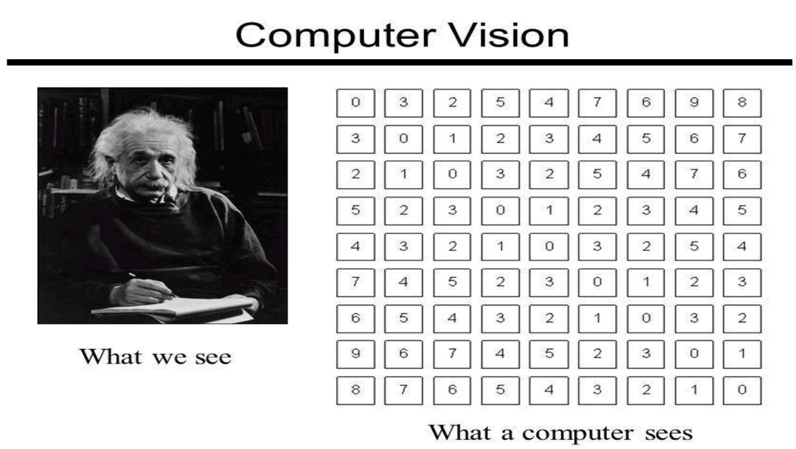


Figure 1.1 - What computer sees

Deep Learning is a subset of machine learning which is derived from the structure and functioning of a human brain. It is meant to simulate neural networks found in our brains. It is essentially copying the pattern recognition abilities of the human brain by processing thousands or even millions of data points.

Let’s say we want to identify a dog in an image, so in traditional machine learning will work by first identifying a nose, then an ear and then other features and finally it tells us that the image is a dog or not. In deep learning, we just show the system a large number of pictures of dogs and the machine can learn just like a human brain.

Basic of deep learning is a Perceptron as shown in Figure.1.2.

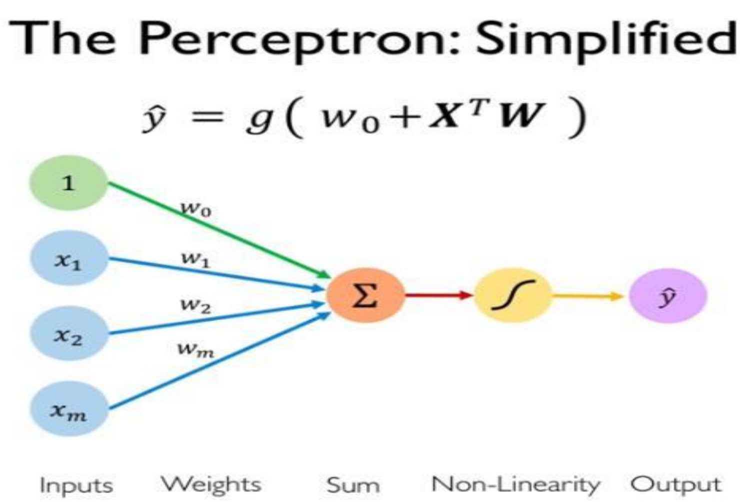


Figure 1.2 - A Perceptron.

It mimics a neuron of the human brain. These two skills help us to think of an idea that would help us to reduce the spread of corona-virus

## 1.3 Objective

The foremost goal of this project is to design a software system with a webcam on top so that it should be able to detect whether a person is maintaining a social distance or wearing a mask. This project report presents the design of such a system having better performance. Parameters varied for accuracy are optimizers, loss function or increases the number of layers. The comparison of the presented system with the other system proposed in different literatures have also been done [1].

# CHAPTER - 2

## 2.1 INTRODUCTION

In this paper, implementation of vision-based face mask detection and social distance tracker has been discussed. COVID-19 virus has the tendency to spread through droplets caused by cough and sneeze [2]. Face mask detection is the most efficient way to reduce the spread of virus. For mask detection, first face of the person is detected from the video frame and the rest of the backgrounds are cropped and then on that face mask classifier is applied. Various dataset has been analyzed like Real World Masked Face Dataset, Prajna bhandary face mask dataset etc. But this paper discusses face mask detection with the help of Prajna bhandary face mask dataset as it shows better accuracy for training the model with a wide variety of images of a person with or without mask. It has 1376 images with 690 images of a person wearing a mask and 686 images of a person without a mask.

For face mask detection, whole process is divided into three part:

- Face detection
- Face mask classifier
    - ❖ Data preprocessing
    - ❖ Training the Convolutional Neural Network (CNN).
- Applying the system to detect face masks.

## 2.2 Face Detection

Face detection has a wide range of applications from smart-phone security purpose to verify the user to getting the emotion of the face, biometrics etc. The system uses Viola Jones real time face detection algorithm from a video footage, also called as Haar cascade method [3]. Pre-trained model has been used for our application as it has great accuracy and much faster than any deep learning techniques.

### 2.2.1 Haar Cascade

Haar Cascade is a machine learning based approach where a cascade function is trained from a lot of positive and negative images. It is then used to detect objects in other images. In this, the first step is to collect the Haar Features. A Haar feature considers adjacent rectangular regions at a selected location in a detection window, sums up the pixel intensities in each region and calculates the difference between these sums. But among all these features calculated, most of them are irrelevant. So, we need to have features which are useful to us, which is accomplished using a concept called “Adaboost” which both selects the best features and trains the classifiers that use them. During the detection phase, a window of the target size is moved over the input image, and for each subsection of the image and Haar features are calculated. This difference is then compared to a learned threshold that separates non-objects from objects.

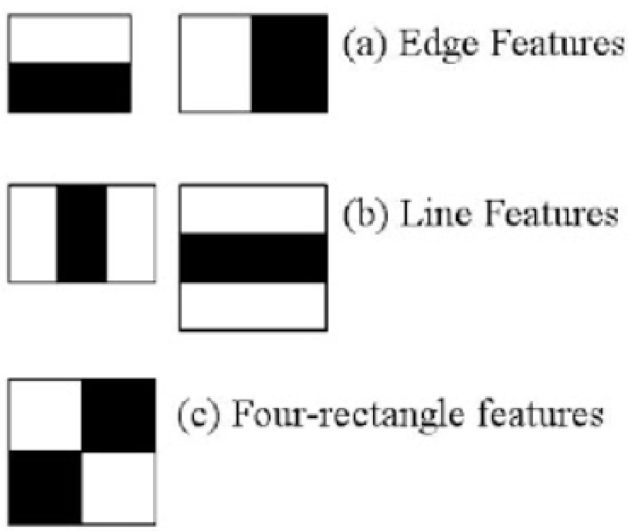


Figure 2.1- Haar Features

## 2.3 Face mask classifier

This classifier is designed to know the differences on the detected face and to classify them as mask/no mask person.

### 2.3.1 DATA PRE-PROCESSING

The data images are collected from the internet and are in random shape. So, for easy processing of the data, the images are firstly resized to a common shape of 100x100. The data images are colored but for face mask detection color does not present any challenge. So, again for easy computation of the data the colored images are converted into gray-scale images. This reduces the 3-D array (because of each array for RGB) to 1-D array (due to grayscale image), which further reduces the computation by very much.

The whole process is summarized in Figure.2.2

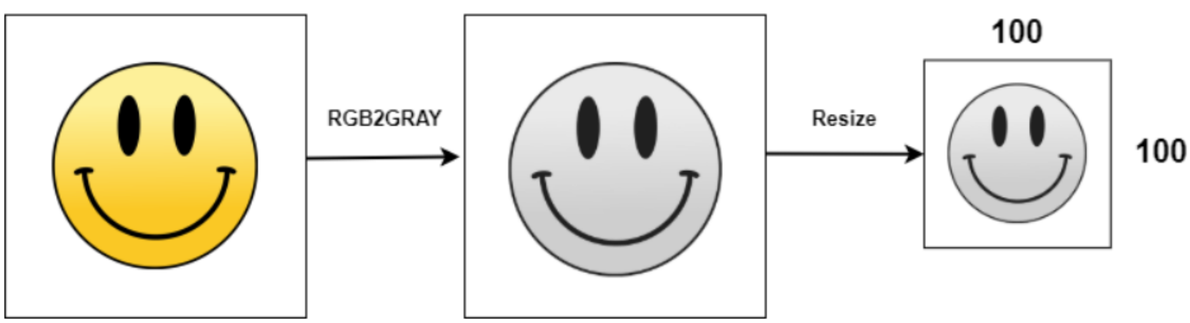


Figure 2.2- Pre-processing of the data

After resizing and conversion to gray-scale, each image was then converted to an array of numbers, with category 0 as with mask and category 1 as without mask. Data array and category was then separated and normalized by 255, then training was performed on the data.

### 2.3.2 TRAINING THE CONVOLUTIONAL NEURAL NETWORK

**Why Convolutional Neural Network?**

Using a neural network in image classification for large dataset is very challenging.

For Input = 1000*1000*3

Number of Features = 3 million.

So, we have large number of parameters to handle in case of memory as well as computation.

This is where Convolution operation comes into picture.

**What is Convolution?**

Convolution is a linear operation that involves the multiplication of a group of weights with the input, very similar to a traditional neural network. Given that the technique was designed for two-dimensional input, the multiplication is performed between an array of input data and a two-dimensional array of weights, called a filter or a kernel. Convolution is done to detect features in an image dataset.

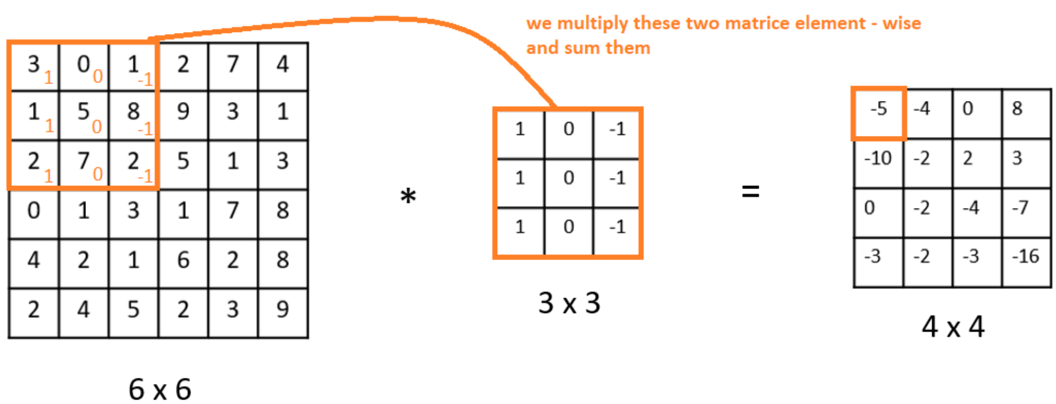


Figure 2.3 Convolution Operation

Convolution is done to detect features like vertical edges, horizontal edges or edges at different angles. Basically, there are filter related to that like vertical edge detection is described in figure 2.4

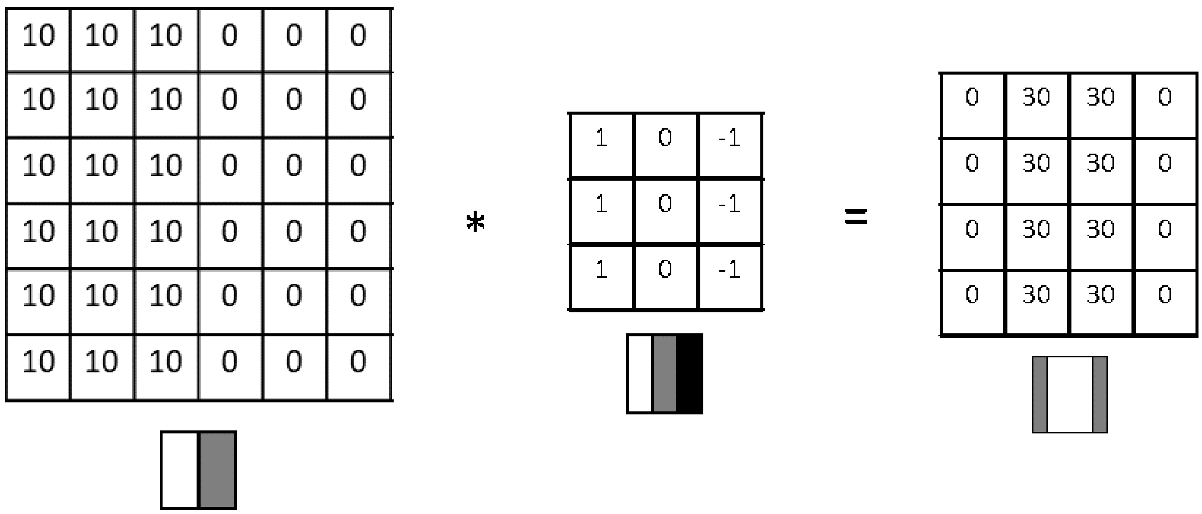


Figure 2.4- Vertical edge detection

Middle white line in the output of the above figure describes the vertical edges between 10- and 0-pixel values of the input image. The white line is wide as the input image is small, if an image with higher resolution is provided then a sharp and narrow edge is detected as shown in figure 2.5.

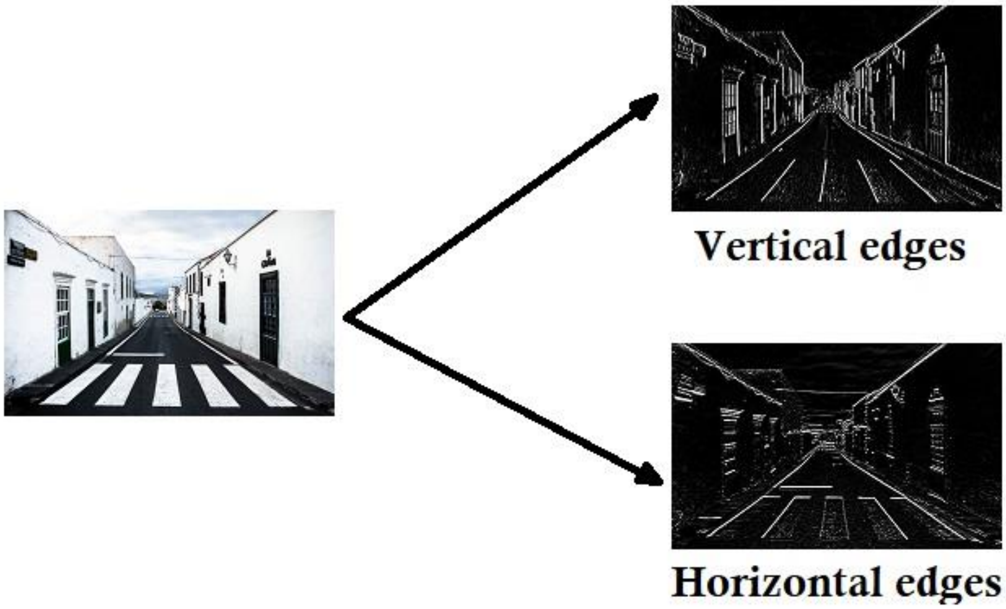


Figure-2.5 Features in image dataset

So, for detecting the features of a person with a face mask or not, weights are assigned to the filter and the algorithm is built to learn the parameters.

$$\underbrace{\begin{array}{|c|c|c|c|c|c|}\hline 3 & 0 & 1 & 2 & 7 & 4 \\ \hline 1 & 5 & 8 & 9 & 3 & 1 \\ \hline 2 & 7 & 2 & 5 & 1 & 3 \\ \hline 0 & 1 & 3 & 1 & 7 & 8 \\ \hline 4 & 2 & 1 & 6 & 2 & 8 \\ \hline 2 & 4 & 5 & 2 & 3 & 9 \\ \hline \end{array}}_{6 \times 6} \overset{\text{convolution}}{*} \underbrace{\begin{array}{|c|c|c|}\hline W_1 & W_1 & W_1 \\ \hline W_1 & W_1 & W_1 \\ \hline W_1 & W_1 & W_1 \\ \hline \end{array}}_{3 \times 3} = \underbrace{\begin{array}{|c|c|c|c|}\hline 0 & 0 & 0 & 0 \\ \hline 30 & 10 & -10 & -30 \\ \hline 30 & 10 & -10 & -30 \\ \hline 0 & 0 & 0 & 0 \\ \hline \end{array}}_{4 \times 4}$$

Figure-2.6 Convolution layer

**What is Convolutional Neural network?**

In deep learning, a convolutional neural network (CNN, or ConvNet) is a class of deep neural networks, most commonly applied to analyzing visual imagery. CNNs are regularized versions of multilayer perceptrons. Multilayer perceptrons usually mean fully connected networks, that is, each neuron in one layer is connected to all neurons in the next layer. The "fully-connectedness" of these networks makes them susceptible to overfitting data. Typical ways of regularization include adding some form of magnitude measurement of weights to the loss function. The preprocessing required in a CNN is far lower as compared to other classification algorithms. There are three types of layer in CNN:

- Convolution Layer: This layer has the responsibility to do convolution operation of the applied input [9] and pass the results to the next layer. It tends to concentrate all the important information and also reduces the image size

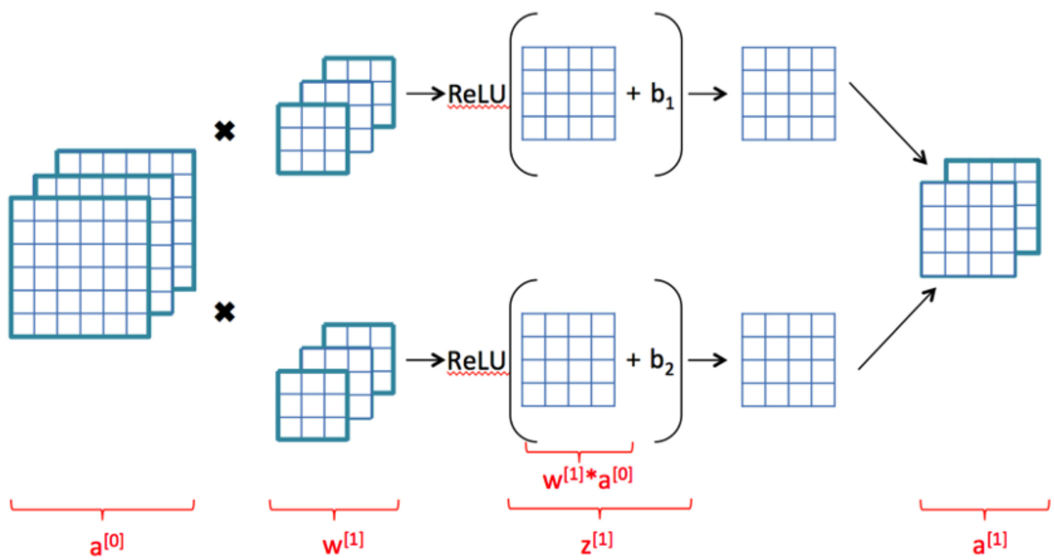


Figure 2.7 – Convolutional Layer.

- Pooling Layer: This layer slides the filter all over the image and concentrates the information surrounded by that filter into one pixel. It reduces the dimension of the image, hence decreases the parameters to learn during training of the model which in turn reduces the unnecessary computation.

- Fully Connected Layer

**CNN Architecture**

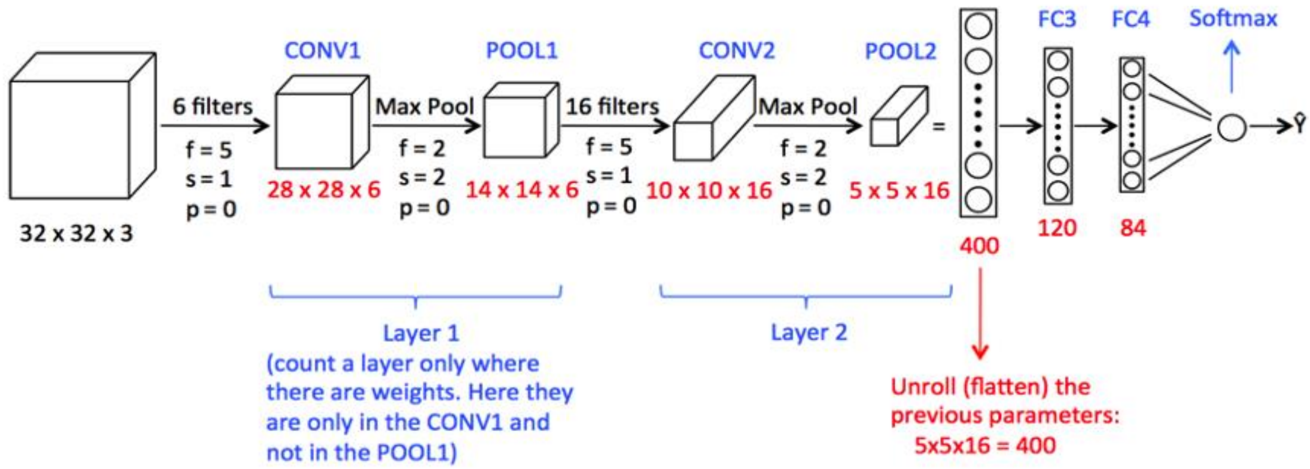


Figure 2.8 – CNN Architecture

## 2.4 Applying the system to detect face mask.

After detection of the face is done, the face image is provided to the model that was created earlier and the results are predicted. The activation function used in the last layer is based on the probability, if probability is greater than 0.5 then a red box is created around a face indicating that the person is not wearing a mask and if probability is less than 0.5 then a green box is created around a face indicating that a person is wearing a mask.

```
In [3]: model.summary()

Model: "sequential_1"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
conv2d_1 (Conv2D)            (None, 98, 98, 256)       2560
_________________________________________________________________
activation_1 (Activation)    (None, 98, 98, 256)       0
_________________________________________________________________
max_pooling2d_1 (MaxPooling2 (None, 49, 49, 256)       0
_________________________________________________________________
conv2d_2 (Conv2D)            (None, 47, 47, 256)       590080
_________________________________________________________________
activation_2 (Activation)    (None, 47, 47, 256)       0
_________________________________________________________________
max_pooling2d_2 (MaxPooling2 (None, 23, 23, 256)       0
_________________________________________________________________
flatten_1 (Flatten)          (None, 135424)            0
_________________________________________________________________
dense_1 (Dense)              (None, 64)                8667200
_________________________________________________________________
dense_2 (Dense)              (None, 1)                 65
_________________________________________________________________
activation_3 (Activation)    (None, 1)                 0
=================================================================
Total params: 9,259,905
Trainable params: 9,259,905
Non-trainable params: 0
```

Figure 2.9 - Model with different layers.

## 2.5 Conclusion

This chapter gives a basic overview of face detection algorithms and Convolutional neural networks to be used for face mask detection. It also deals with the compilation of model with different optimizers. Optimizers vary the weights in to reduce the losses. Loss signifies the difference between actual and the predicted probability distribution. Adam is considered the best optimizer among different algorithms and the fastest one to converge to minima.

# CHAPTER – 3

## 3.1 Introduction

Maintaining social distance is an important task in current pandemic specially in busy places such as malls. To flatten the exponentially increasing curves of covid-19 cases, very reliable practice is to maintain a safe distance between each person. This chapter presents a software system to monitor a safe distance as a precautionary measure. It helps to provide efficient solutions to detect whether a group of people are gathering or not. This surveillance is required in areas like railway stations, airports etc. The algorithm starts with detection of a person using a pre-trained object detection model and then modifying that object detection algorithm, so it also calculates the distance between the persons. The whole process is divided into four part:

- Object Detection
- Mathematical Processing
- Social Distance Tracking

## 3.2 Object Detection

Various Object detection applications already present are using pre-trained CNN models as they provide better accuracy and fastest frame rate. So, this section tries to present the existing models and comparison has been made to identify the suitable one [4]. Single shot detector (SSD) and Faster Region-based Convolutional Neural Network (Faster R-CNN) are compared in this section. COCO dataset [5] has been used for the input as a benchmark.

| Model Name | Execution Time (s) | Highest Accuracy | Object Detected |
|---|---|---|---|
| SSD Mobilenet V1 COCO | 219.58 | 94% | 2 |
| SSD Inception V2 COCO | 298.22 | 97% | 2 |
| Faster RCNN Inception V2 COCO | 420.71 | 99% | 3 |
| Mask RCNN Inception Resenet V2 Atrous | 6008.02 | 99% | 5 |

Figure 3.1- Comparative overview

As in the above figure, mask R CNN Inception resnet V2 atrous has the highest accuracy and detects a highest number of objects in a selected image. But considering the both parameters frame rate/execution time and accuracy SSD mobilenet V1 COCO has the better real time performance even on hardware limitation. Output of the algorithm is bounding box coordinates specifying the location of the person in a video frame and scores which provides the probability of the detected person belonging to person class.

Table 1. MobileNet Body Architecture

| Type / Stride | Filter Shape | Input Size |
|---|---|---|
| Conv / s2 | $3 \times 3 \times 3 \times 32$ | $224 \times 224 \times 3$ |
| Conv dw / s1 | $3 \times 3 \times 32$ dw | $112 \times 112 \times 32$ |
| Conv / s1 | $1 \times 1 \times 32 \times 64$ | $112 \times 112 \times 32$ |
| Conv dw / s2 | $3 \times 3 \times 64$ dw | $112 \times 112 \times 64$ |
| Conv / s1 | $1 \times 1 \times 64 \times 128$ | $56 \times 56 \times 64$ |
| Conv dw / s1 | $3 \times 3 \times 128$ dw | $56 \times 56 \times 128$ |
| Conv / s1 | $1 \times 1 \times 128 \times 128$ | $56 \times 56 \times 128$ |
| Conv dw / s2 | $3 \times 3 \times 128$ dw | $56 \times 56 \times 128$ |
| Conv / s1 | $1 \times 1 \times 128 \times 256$ | $28 \times 28 \times 128$ |
| Conv dw / s1 | $3 \times 3 \times 256$ dw | $28 \times 28 \times 256$ |
| Conv / s1 | $1 \times 1 \times 256 \times 256$ | $28 \times 28 \times 256$ |
| Conv dw / s2 | $3 \times 3 \times 256$ dw | $28 \times 28 \times 256$ |
| Conv / s1 | $1 \times 1 \times 256 \times 512$ | $14 \times 14 \times 256$ |
| 5× Conv dw / s1 | $3 \times 3 \times 512$ dw | $14 \times 14 \times 512$ |
| 5× Conv / s1 | $1 \times 1 \times 512 \times 512$ | $14 \times 14 \times 512$ |
| Conv dw / s2 | $3 \times 3 \times 512$ dw | $14 \times 14 \times 512$ |
| Conv / s1 | $1 \times 1 \times 512 \times 1024$ | $7 \times 7 \times 512$ |
| Conv dw / s2 | $3 \times 3 \times 1024$ dw | $7 \times 7 \times 1024$ |
| Conv / s1 | $1 \times 1 \times 1024 \times 1024$ | $7 \times 7 \times 1024$ |
| Avg Pool / s1 | Pool $7 \times 7$ | $7 \times 7 \times 1024$ |
| FC / s1 | $1024 \times 1000$ | $1 \times 1 \times 1024$ |
| Softmax / s1 | Classifier | $1 \times 1 \times 1000$ |

Figure 3.2- Architecture of SSD MobileNet V1

## 3.3 Mathematical Processing

Determining the position of a person’s bounding box using an object detection algorithm. Then this equation is used to determine the center position of the bounding box [1].

$$C(x,\ y) = \left(\frac{x_{min} + x_{max}}{2}, \frac{y_{min} + y_{max}}{2}\right)$$

Further if there are more than 1 person in a frame then there are several bounding box centers and the distance between each center is calculated using the following equation [1].

$$d(C_1,C_2) = \sqrt{(x_{max} - x_{min})^2 + (y_{max} - y_{min})^2}$$

## 3.4 Social Distance Tracking

Distance calculated is compared with the threshold i.e 250 pixels. If the distance between each pair is greater than the threshold value then a green line is created indicating that they are maintaining a safe distance and if the distance between each pair is less than the threshold value then a red line is created indicating that they are not maintaining a safe distance. The surveillance camera is put in a bird eye projection. Instead of bird eye view, if horizontal view is required for tracing the distance then generalization is done with the use of a stereo camera.

## 3.5 Conclusion

Social Distancing is the best precautionary measure to reduce physical contact which is the main cause for the spread of virus. A system has been built with the use of python and OpenCV library that help us to alert through software that groups of people are not maintaining a social distance. Due to low computational resources, object detection algorithms are not much accurate to be used on a commercial scale.

# CHAPTER - 4

## 4.1 Introduction

This chapter presents a comparative study of the results of the face mask detector and social distance tracker with the previous work. Results have a discussion on the accuracy and the performance of the proposed model.

## 4.2 Experiments and results

### 4.2.1 Face mask detection

The software system was built in an environment with a jupyter notebook as a web application to visualize the results obtained using python as a commanding language. The proposed system has a validation accuracy of 97.18% which is close to 98.5% accuracy as compared to the previous literature[6] . Figure.13 shows the accuracy and error of the model during training and validation. As we can see from the figure, as the number of epochs increased, the accuracy of the model increased and the loss decreased.

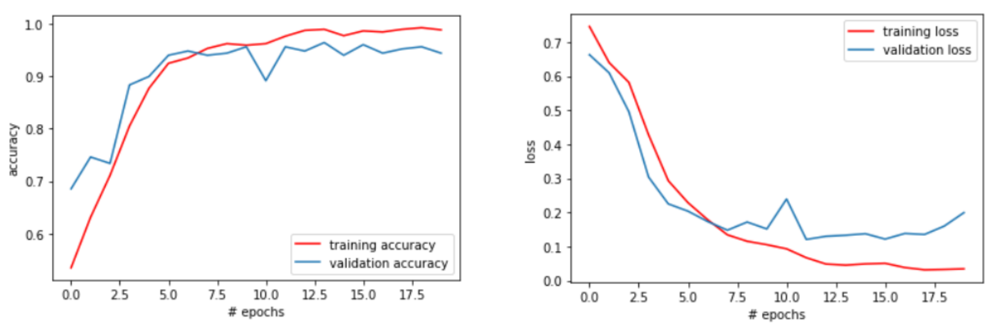


Figure 4.1- Accuracy vs epochs and Loss vs epochs

The real time performance of the system has been obtained with a frame rate of 2.85 (frame per sec) that is the maximum value obtained. Pre-processing is done by the program to provide some accurate results that is the reason for slow frame rate [7] and also because of hardware limitation.

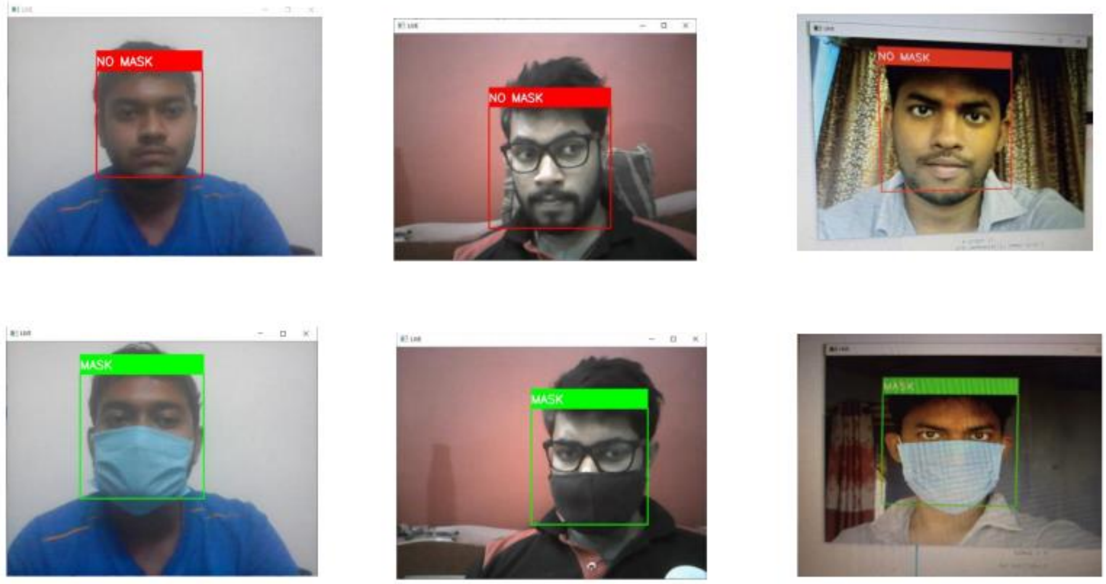


Figure 4.2 Results obtained with mask and without mask

### 4.2.1 Social Distance tracker

The experimental setup for the analysis of social distance tracker is shown in fig.4.3.

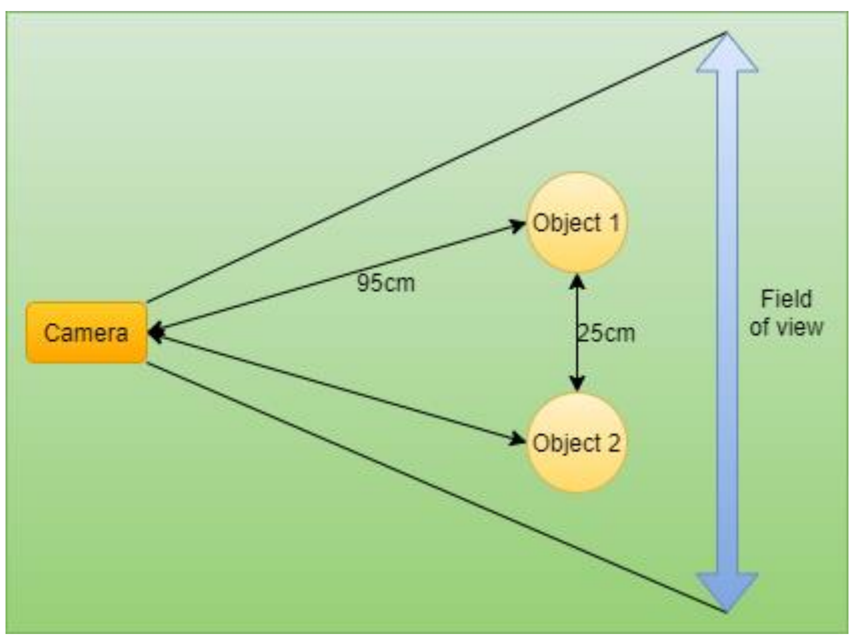


Figure 4.3 Experimental setup

Threshold value for social distancing is kept at 25 cm (real world unit) (Rd) for experimental analysis, keeping in mind the field of view of the camera and the object distance in front of the camera is kept at 95 cm (Od) . The pixel value corresponding to that is 200 pixels (Pd). So, for mapping pixel value to real world distance [8], this value has been used.

25cm is proportional to 200 pixels.

25 = k*200, so k = 0.125

Rcd = k*Pd

This equation has been used for real world distance mapping

where,

Rcd is the real calculated distance between each pair of objects

Od is the object distance in-front of the camera

k is proportionality constant for a particular value of Od(k increase as Od increases)

Pd is the euclidean distance between each pair of objects in pixel.

**Table 1. Social distance at two different object distance in front of the camera**

| Test case | Od(cm) | k | Pd (pixel) | Rcd(cm) | Rd(cm) | Error(Rd - Rcd ) |
|---|---|---|---|---|---|---|
| | 95 | 0.125 | 200 | - | 25 | - |
| 1 | - | - | 105 | 13.125 | 14 | 0.875 |
| 2 | - | - | 221 | 27.625 | 27 | -0.625 |
| | 130cm | 0.13409 | 186.43 | | 25 | |
| 3 | | | 95.458 | 12.8 | 14 | 1.2 |
| 4 | | | 208.74 | 27.99 | 27 | -0.99 |
| Mean absolute error = 0.9225 | | | | | | |

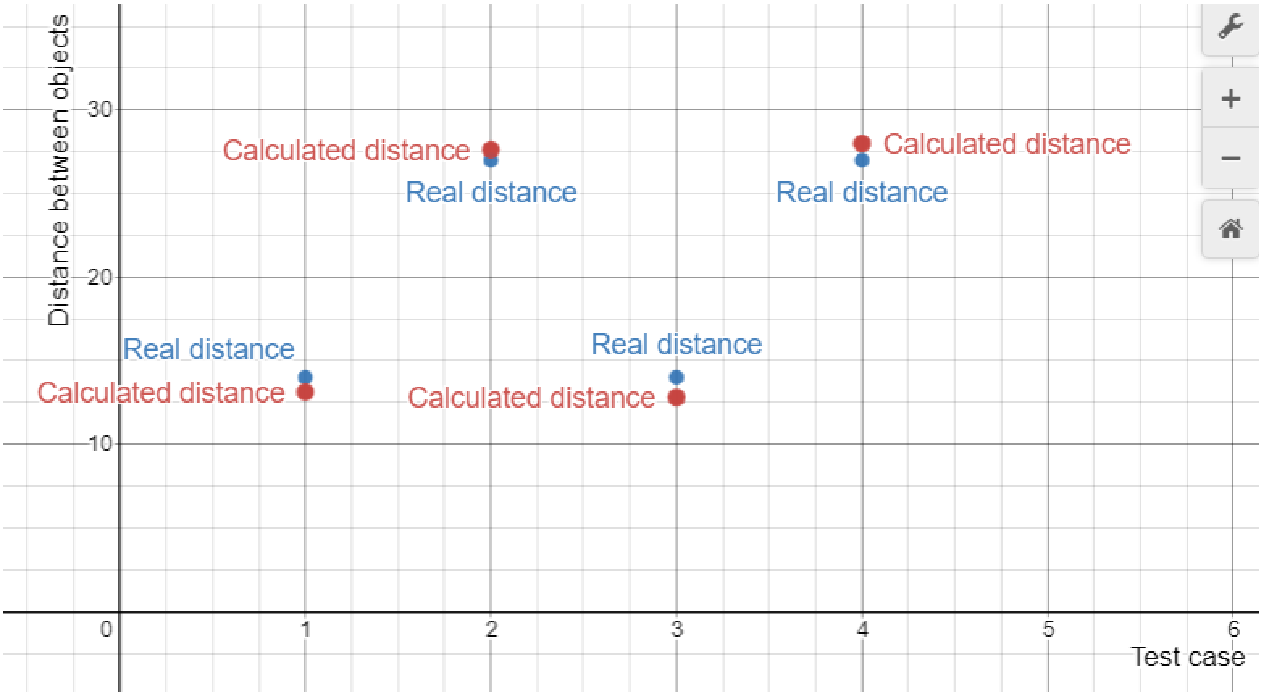


Figure 4.4 - Graph for measured and real distance vs no of test cases to show the error between them.

Real Time performance results are also shown in figure 4.5.


Figure 4.5 - Result of the proposed system with green (safe distance) and red (unsafe distance) line.

## 4.3 Conclusions

From the results presented above it is clear that this paper proposes an efficient solution for face mask detection and social distance tracking. For real time object detection, certain improvement is needed, in the proposed system SSD mobilenetv1 architecture is used trained on COCO dataset. For face mask detection, results differ in low light conditions but have good accuracy in other cases.

# CHAPTER - 5

## 5.1 FUTURE SCOPE

Further this system can be made more accurate and high performing. With the help of a stereo camera, horizontal projection can be improved and to be added to the existing system. The whole project can be embedded in a raspberry pi for easy to install and accessible to everyone. Field of view is restricted as using a single camera, further 360-degree view can be added to monitor the people more efficiently.

## 5.2 APPLICATIONS

- Bank cashier can use this software system to monitor the group of people in front of him in covid scenarios
- This system can be installed in malls, railway station etc. to automates the monitoring of people

# REFERENCES


[1]. Afiq Harith Ahamad, Norliza Zaini, Mohd Fuad Abdul Latip,” Person Detection for Social Distancing and Safety Violation Alert based on Segmented ROI”, Aug. 2020.

[2]. P. Bahl, C. Doolan, C. de Silva, A. A. Chughtai, L. Bourouiba, and C. R. MacIntyre, “Airborne or Droplet Precautions for Health Workers Treating Coronavirus Disease 2019?,” *J. Infect. Dis.*, no. XxXxxx, pp. 1–8, 2020, doi: 10.1093/infdis/jiaa189.

[3]. P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," *Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. CVPR 2001*, 2001, pp. I-I, doi: 10.1109/CVPR.2001.990517.

[4]. N. Yadav and U. Binay, "Comparative Study of Object Detection Algorithms", *Int. Res. J. Eng. Technol.*, pp. 586-591, 2017

[5]. T. Y. Lin et al., "Microsoft COCO: Common objects in context", *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 8693 LNCS, no. PART 5, pp. 740-755, 2014.

[6]. Mohammad Marufur Rahman, Md. Motaleb Hossen Manik, Md. Milon Islam, Saifuddin Mahmud, Jong-Hoon Kim, “An Automated System to Limit COVID-19 Using Facial Mask Detection in Smart City Network”, Sep.2020.

[7]. Siegfried, Ivan M. “Comparative Study of Deep Learning Methods in Detection Face Mask Utilization.” OSF Preprints, 14 Sept. 2020.

[8]. Adina Rahim,Ayesha Maqbool ,Tauseef Rana.”Monitoring social distancing under various low light conditions with deep learning and a single motionless time of flight camera”, 25 feb,2021. doi : 10.1371/journal.pone.0247440

[9]. S. Albawi, T. A. Mohammed and S. Al-Zawi, "Understanding of a convolutional neural network," *2017 International Conference on Engineering and Technology (ICET)*, 2017, pp. 1-6, doi: 10.1109/ICEngTechnol.2017.8308186.

# Project Report

ORIGINALITY REPORT

13 %

SIMILARITY INDEX

PRIMARY SOURCES

| | | |
|---|---|---|
| 1 | www.studymode.com<br>Internet | 121 words — 3% |
| 2 | en.m.wikipedia.org<br>Internet | 80 words — 2% |
| 3 | bitmesra.ac.in<br>Internet | 56 words — 1% |
| 4 | machinelearningmastery.com<br>Internet | 54 words — 1% |
| 5 | www.coursehero.com<br>Internet | 47 words — 1% |
| 6 | Eman Alajrami, Hani Tabash, Yassir Singer, M.-T. El Astal. "On using AI-Based Human Identification in Improving Surveillance System Efficiency", 2019 International Conference on Promising Electronic Technologies (ICPET), 2019<br>Crossref | 43 words — 1% |
| 7 | www.willberger.org<br>Internet | 37 words — 1% |
| 8 | Singh, Birender, Nikitha Kandru, and Mahesh Chandra. "Application control using eye motion", | 31 words — 1% |

Crossref

| | | |
|---|---|---|
| 9 | ijcrt.org<br>Internet | 20 words — 1% |

| | | | |
|---|---|---|---|
| EXCLUDE QUOTES | ON | EXCLUDE MATCHES | OFF |
| EXCLUDE BIBLIOGRAPHY | ON | | |