**A**

**Synopsis Report**

On

**Face Mask Detection**

For

fulfillment of award of the

**B. Tech Degree in Information Technology**

**Under the Supervision of**

**Mr. Ashish Tripathi**



**Submitted by:**

Atishay Jain (1901920130051)

Ankur Bindra (1901920130029)

Akshat Jain (1901920130022)

**Session: 2021-22**

**G. L. Bajaj Institute of Technology and Management,**

**Greater Noida**

**TABLE OF CONTENTS**

1. Introduction

2. Background

3. Literature Review

4. Proposed work

5. Methodology

6. Result

7. Conclusion

8. Future Scope

**1. INTRODUCTION**

The 209th report of the world health organization (WHO) published on 16th August 2020 reported that coronavirus disease (COVID-19) caused by acute respiratory syndrome (SARS-CoV2) has globally infected more than 6 million people and caused over 379,941 deaths worldwide. According to Carissa F. Etienne, Director, Pan American Health Organization (PAHO), the key to control COVID-19 pandemic is to maintain social distancing, improving surveillance and strengthening health systems. Recently, a study on understanding measures to tackle COVID-19 pandemic carried by the researchers at the University of Edinburgh reveals that wearing a face mask or other covering over the nose and mouth cuts the risk of Coronavirus spread by avoiding forward distance travelled by a person’s exhaled breath by more than 90%. Their results strongly recommend the use of the face masks in general public to curtail the spread of Coronavirus. Further, with the reopening of countries from COVID-19 lockdown, Government and Public health agencies are recommending face mask as essential measures to keep us safe when venturing into public. To mandate the use of facemask, it becomes essential to devise some technique that enforce individuals to apply a mask before exposure to public places.

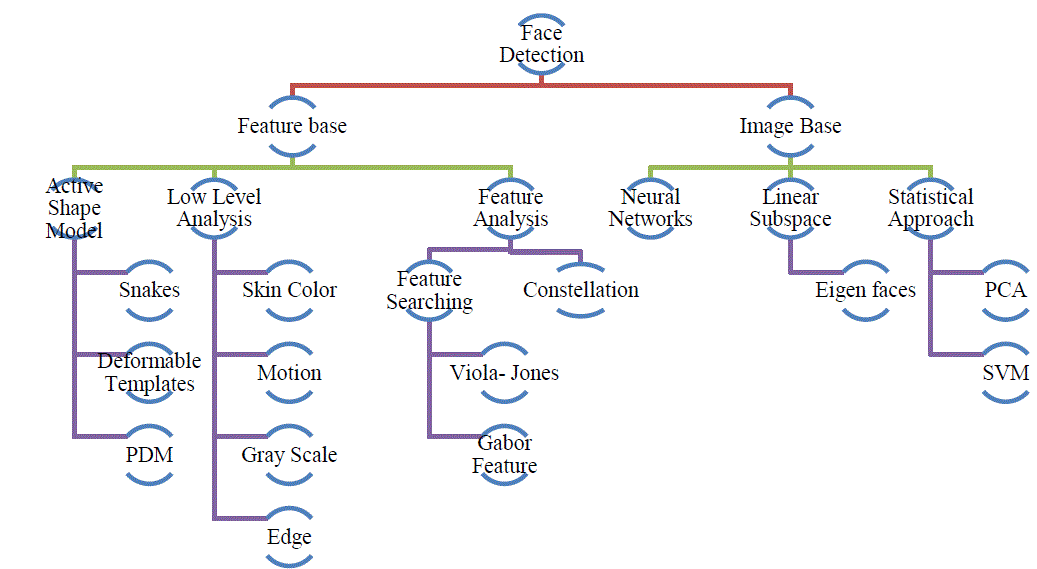
Face mask detection refers to detect whether a person is wearing a mask or not. In fact, the problem is reverse engineering of face detection where the face is detected using different machine learning algorithms for the purpose of security, authentication and surveillance. Face detection is a key area in the field of Computer Vision and Pattern Recognition. A significant body of research has contributed sophisticated to algorithms for face detection in past. The primary research on face detection was done in 2001 using the design of handcraft feature and application of traditional machine learning algorithms to train effective classifiers for detection and recognition. The problems encountered with this approach include high complexity in feature design and low detection accuracy. In recent years, face detection methods based on deep convolutional neural networks (CNN) have been widely developed.

**2. BACKGROUND**

# In 2021, the largest pandemic in recent history spread through the world: COVID-19. there have already been 152 million cases and 3 million deaths around the world. In many regions, those numbers are considerably under-counted. Beyond that, many parts of the world have slowed or stopped due to the human, economic, and social impacts of distancing and protection measures. For the purpose of the ongoing pandemic and predictions for future pandemics, this project seeks to create a mask detection system that is capable of recognizing whether people wearing their masks or not. Projects with similar intent have been quite popular due to the ongoing pandemic. In the paper by Adnane Cabani and his colleagues from Universite de Haute-Alsace, a method was proposed to utilize haar-cascade based feature detectors to individually determine the presence of nose and mouth from a detected face. Their logic follows that no mask is worn if we can successfully detect a mouth from the face, mask is worn incorrectly if we can detect a nose by not a mouth, and mask is worn correctly if we can detect neither a nose nor a mouth. This approach is efficient and intuitive.

**3. LITERATURE REVIEW**

Face detection is a computer technology that determines the location and size of human face in arbitrary (digital) image. The facial features are detected and any other objects like trees, buildings and bodies, etc. are ignored from the digital image. Basically, there are two types of approaches to detect facial part in the given image i.e., feature base and image base approach. Feature base approach tries to extract features of the image and match it against the knowledge of the face features. While image base approach tries to get best match between training and testing images.



# **Fig: Detection Methods**

# **3.1. FEATURE-BASED APPROACH**

# The method of finding image displacements which is easiest to understand is the feature-based approach. This finds features (for example, image edges, corners, and other structures well localized in two dimensions) and tracks these as they move from frame to frame. This involves two stages. Firstly, the features are found in two or more consecutive images. The act of feature extraction, if done well, will both reduce the amount of information to be processed (and so reduce the workload), and also go some way towards obtaining a higher level of understanding of the scene, by its very nature of eliminating the unimportant parts. Secondly, these features are matched between the frames. In the simplest and commonest case, two frames are used and two sets of features are matched to give a single set of motion vectors. Alternatively, the features in one frame can be used as seed points at which to use other methods (for example, gradient-based methods to find the flow. The two stages of feature-based flow estimation each have their own problems. The feature detection stage requires features to be located accurately and reliably. This has proved to be a non-trivial task, and much work has been carried out on feature detectors. If a human is shown, instead of the original image sequence, a sequence of the detected features (drawn onto an empty image), then a smoothly moving set of features should be observable, with little feature flicker. The feature matching stage has the well-known correspondence problem of ambiguous potential matches occurring; unless image displacement is known to be smaller than the distance between features, some method must be found to choose between different potential matches. Finding optic flow using edges has the advantage (over using two dimensional features) that edge detection theory is well advanced, compared with that of two-dimensional feature detection. It has the advantage over approaches which attempt to find flow everywhere in the image, such as the method developed by Horn and Schunk.

# **3.2. IMAGE-BASED APPROACH**

# The appearance-based method depends on a set of delegate training face images to find out face models. The appearance-based approach is better than other ways of performance. In general appearance-based method rely on techniques from statistical analysis and machine learning to find the relevant characteristics of face images. This method also used in feature extraction for face recognition.

# The appearance-based model further divided into sub-methods for the use of face detection which are as follows-

# 3.2.1. Eigenface-Based: -

# Eigenface based algorithm used for Face Recognition, and it is a method for efficiently representing faces using Principal Component Analysis.

# 3.2.2. Distribution-Based: -

# The algorithms like PCA and Fisher’s Discriminant can be used to define the subspace representing facial patterns. There is a trained classifier, which correctly identifies instances of the target pattern class from the background image patterns.

# 3.2.3. Neural-Networks: -

# Many detection problems like object detection, face detection, emotion detection, and face recognition, etc. have been faced successfully by Neural Networks.

# 3.2.4. Support Vector Machine: -

# Support Vector Machines are linear classifiers that maximize the margin between the decision hyperplane and the examples in the training set. Osuna et al. first applied this classifier to face detection.

# 3.2.5. Sparse Network of Winnows: -

# They defined a sparse network of two linear units or target nodes; one represents face patterns and other for the non-face patterns. It is less time consuming and efficient.

# 3.2.6. Naive Bayes Classifiers: -

# They computed the probability of a face to be present in the picture by counting the frequency of occurrence of a series of the pattern over the training images. The classifier captured the joint statistics of local appearance and position of the faces.

# 3.2.7. Hidden Markov Model: -

# The states of the model would be the facial features, which usually described as strips of pixels. HMM’s commonly used along with other methods to build detection algorithms.

# 3.2.8. Information Theoretical Approach: -

# Markov Random Fields (MRF) can use for face pattern and correlated features. The Markov process maximizes the discrimination between classes using Kullback-Leibler divergence. Therefore, this method can be used in Face Detection.

# 3.2.9. Inductive Learning: -

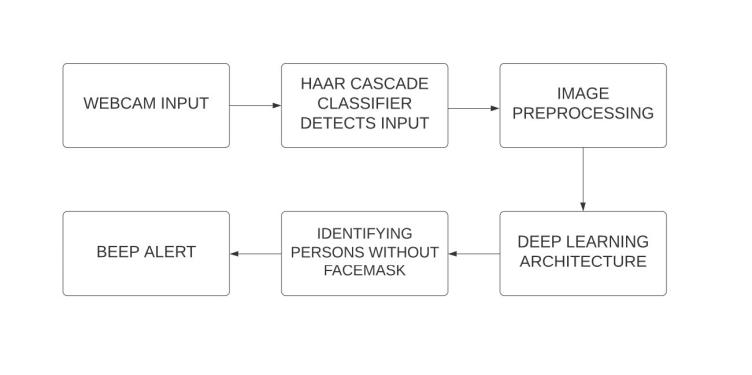
# This approach has been used to detect faces. Algorithms like Quinlan’s C4.5 or Mitchell’s FIND-S used for this purpose.

# **4. PROPOSED WORK**

We decided to build a very simple and basic Convolutional Neural Network (CNN) model using  
TensorFlow with Keras library and OpenCV to detect if you are wearing a face mask to protect  
yourself. All the aspects of our work are described below.

**5. METHODOLOGY**

We have devised a smart framework for detecting facemask in this paper. As the cases of covid-  
19 are decreasing maximum workplaces are opening with half or full employees. Even the  
education institutes are planning to be opened. For screening the people not wearing masks, this  
system can be installed in the entrances of enterprises, educational institutes, public and  
private offices. If the system detects a person’s face with no mask, it will generate a beep alerting  
them to wear mask.



**Fig: Proposed Method**

**A. Deep Learning Architecture**

The deep learning architecture learns various important nonlinear features from the given

samples. Then, this learned architecture is used to predict previously unseen samples.

**B. Image Processing**

Haar Cascade Classifier will detect the input from videocam. The images captured by the

system's webcam required pre-processing before going to the next step. In the pre-processing

step, the image is transformed into a grayscale image because the RGB color image contains so

much redundant information that is not necessary for face mask detection. Then, we resized the

images into (150x150) size to maintain uniformity of the input images to the architecture. Then,

the images are normalized and after normalization, the value of a pixel resides in the range from

0 to 1. Normalization helped the learning algorithm to learn faster and captured necessary

features from the images.

**C. Dataset Collection**

To train our deep learning architecture, we collected images. The architecture of the learning

technique highly depends on CNN. Data from source is collected for training and testing the

model. Dataset contains images of faces only. It consists of about 1,006 images in which 503

images containing people with face masks and 503 images containing people without face mask.

For training purposes, 60% images of each class are used, 10% images are utilized for testing

purposes and 30% images are used for cross validation.

**D. Architecture Development**

The learning model is based on CNN which is very useful for pattern recognition from images.

Neural Network need to see data from both the classes. The network comprises an input layer,

several hidden layers and an output layer. The hidden layers consist of multiple convolution

layers. The features extracted by CNN are used by multiple dense neural networks for

classification purposes. The architecture contains two pairs of convolution layers each followed

by one max pooling layer. The first convolution layer contains kernels with batch size 32 of

window size 3x3. Max pooling layer of window size 2x2. The second convolution layer contains kernels with batch size 64 of window size 3x3. Max pooling layer of window size 2x2. This

layer will be aggregating the results from the previous convolution layer and will be picking the max value in that 2x2 window. It decreases the spatial size of the representation and thereby

reduces the number of parameters. As a result, the computation is simplified for the network.

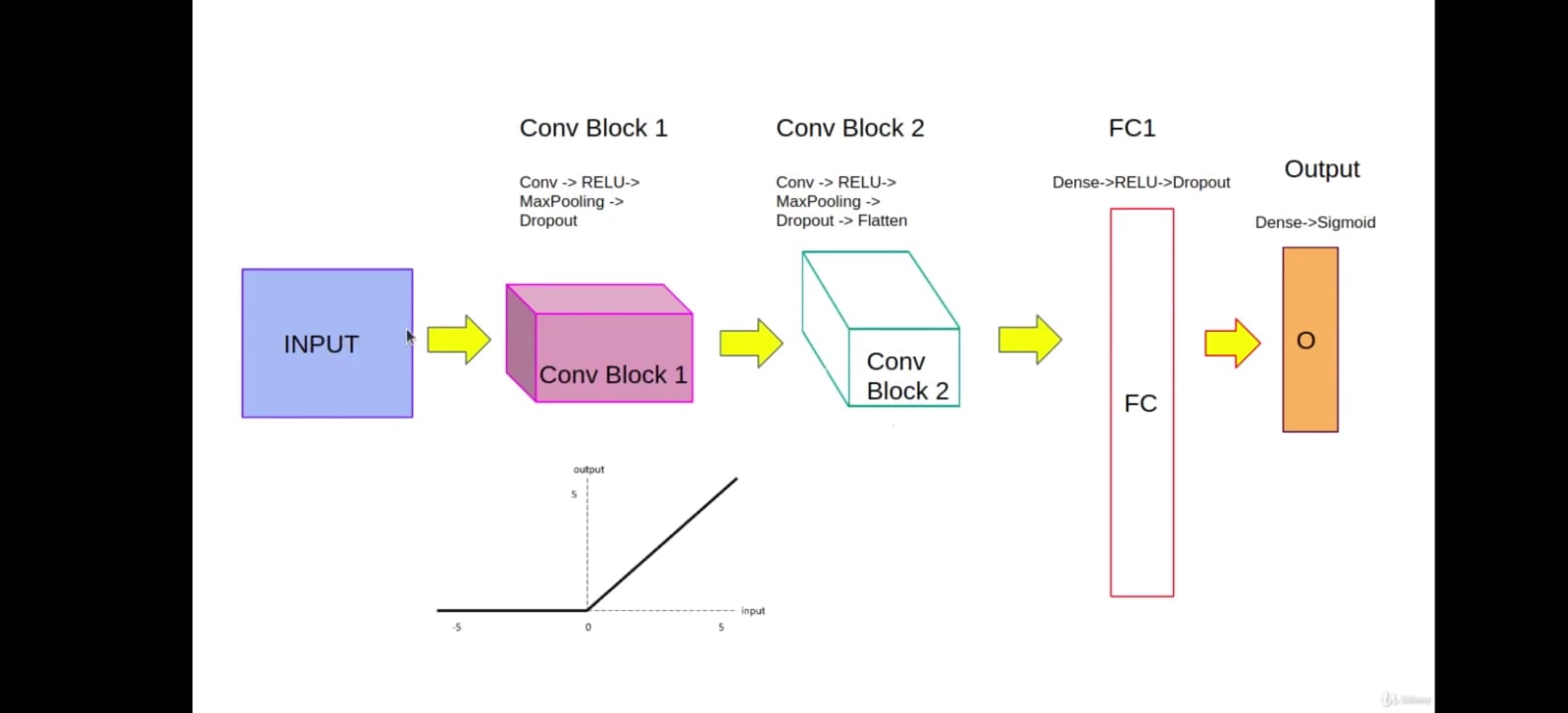
The output of the convolution layers will be flattened and will be converted into a 1-D array.

Then there is one dropout layer and two dense layers. The dropout layer prevents the network

from overfitting by dropping out units. The dense layer comprises a series of neurons each of

them learn nonlinear features. The flattened result this will be fed to the first dense layer of 256

nodes. Then finally second dense layer containing one node as there are two classes.



**Fig: Architecture**

**E. Alert Generation**

The purpose of our system is to screen person not wearing face mask. The learning architecture

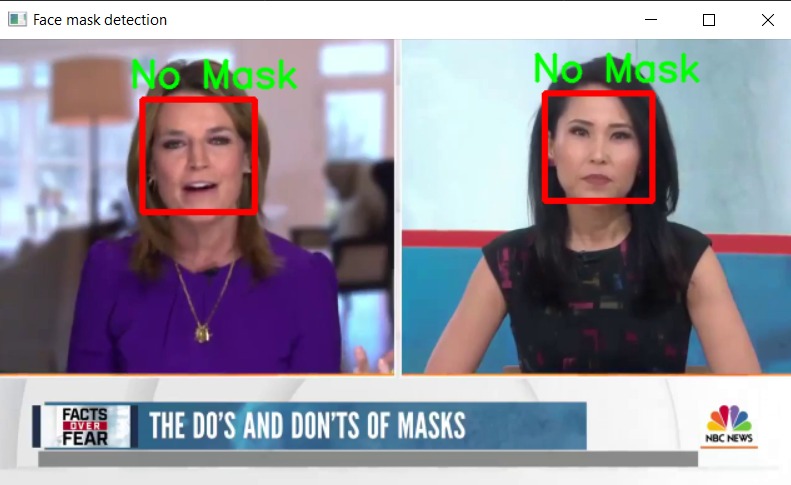
generates result on the input image, classifying the image into mask or no mask classes. If a

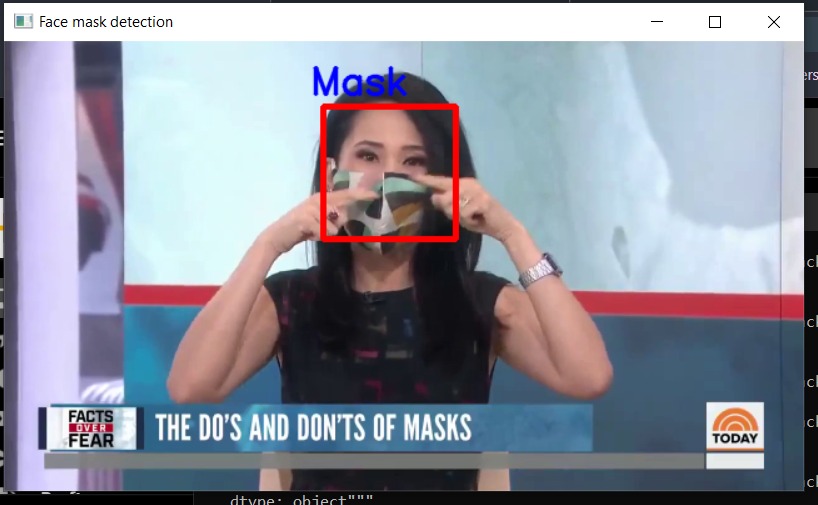
person is detected not wearing a mask then a beep alert will be generated until mask is put on.

And if everyone is wearing a mask then they will be safe from the virus. In this way our system

would help greatly to limit the growth of COVID-19.

**6. RESULT**





**7. CONCLUSION**

In this project, we first download dataset from Kaggle read all the images from the dataset and train the code on them. Then we use CNN to build a model and then compile it. Later, when we provide an image from the test data, it compares the features of that face to all the images from the dataset and returns that the image is with mask or without it. It returns the test image with a rectangle around the face with label of mask or no mask. It can work on laptop webcam or a video file provided.

The code is written using OpenCV, Tensorflow and keras. It runs on google Colab or jupyter notebook.

**8. FUTURE SCOPE**

More than ﬁfty countries around the world have recently initiated wearing face masks compulsory. People have to cover their faces in public, supermarkets, public transports, ofﬁces, and stores. Retail companies often use software to count the number of people entering their stores. They may also like to measure impressions on digital displays and promotional screens. We are planning to improve our Face Mask Detection tool and release it as an open-source project. Our software can be equated to any existing USB, IP cameras, and CCTV cameras to detect people without a mask. This detection live video feed can be implemented in web and desktop applications so that the operator can see notice messages. Software operators can also get an image in case someone is not wearing a mask. Furthermore, an alarm system can also be implemented to sound a beep when someone without a mask enters the area. This software can also be connected to the entrance gates and only people wearing face masks can come.