Maulana Azad National Institute of Technology

BHOPAL, INDIA, 462003

Department of Computer Science and Engineering

Face mask Detection Using CNN

Minor Project Report

Semester - V & VI

Submitted by:

Preetam Patel	181112247
Anubhav Goyal	181112248
Parth Sourav	181112305
Vatsal Rastogi	181112239

Under the guidance of

Dr. R K Pateriya

Department of Computer Science and Engineering

Session : 2020 - 2021

CERTIFICATE

This is to certify that the minor project report carried out on "Face Mask Detection Using CNN" by the 3rd year students:

Preetam Patel	181112247
Anubhav Goyal	181112248
Parth Sourav	181112305
Vatsal Rastogi	181112239

Have successfully completed their minor project in partial fulfilment of their Degree in Bachelor of Technology in Computer Science and Engineering.

Dr. R K Pateriya

(Minor Project Mentor)

Declaration

We, hereby declare that the following report which is being presented in the Minor Project Documentation Entitled as **"Face Mask Detection Using CNN"** is an authentic documentation of our own original work and to the best of our knowledge .The following minor project and its report, in part or whole, has not been presented or submitted by us for any purpose in any other institute or organization. Any contribution made to the research by others, with whom we have worked at Maulana Azad National Institute of Technology, Bhopal or elsewhere, is explicitly acknowledged in the report.

Preetam Patel	181112247
Anubhav Goyal	181112248
Parth Sourav	181112305
Vatsal Rastogi	181112239

ACKNOWLEDGEMENT

With due respect, we express our deep sense of gratitude to our respected guide and coordinator Dr. R K Pateriya Sir for his valuable help and guidance. We are thankful for the encouragement that he has given us in completing this minor project successfully.

It is imperative for us to mention the fact that the report of the minor project could not have been accomplished without the periodic suggestions and advice of our project guide Dr. R K Pateriya Sir and project coordinators Dr. Dhirendra Pratap Singh and Dr. Jaytrilok Choudhary.

We are also grateful to our respected director Dr. N. S. Raghuwanshi for permitting us to use all the facilities of the college.We are also thankful to all the other faculty, staff members and laboratory attendants of our department for their kind cooperation and help.Last but certainly not the least; we would like to express our deep appreciation towards our family members and batch mates for providing the much needed support and encouragement

Abstract

COVID-19 pandemic has rapidly affected our day-to-day life disrupting world trade and movements. Wearing a protective face mask has become a new normal. In the near future, many public service providers will ask the customers to wear masks correctly to avail of their services. Therefore, face mask detection has become a crucial task to help global society.

This work presents a simplified approach to achieve this purpose using some basic Machine Learning packages like Keras, OpenCV and Scikit-Learn.

The proposed method detects the face from the image correctly and then identifies if it has a mask on it or not.

As a surveillance task performer, it can also detect a face along with a mask in motion. The method attains avg accuracy up to 94.77% respectively on dataset. We explore optimized values of parameters using the Sequential Convolutional Neural Network model to detect the presence of masks correctly without causing over-fitting.

Keywords:

Computer Vision , Face Detection, Image;
Tracking; COVID-19; Face Masks; Safety

Table of Contents

Convolutional Neural Network(CNN)

CNN is a feed forward neural network that is generally used to analyze visual images by processing data with a grid like topology.

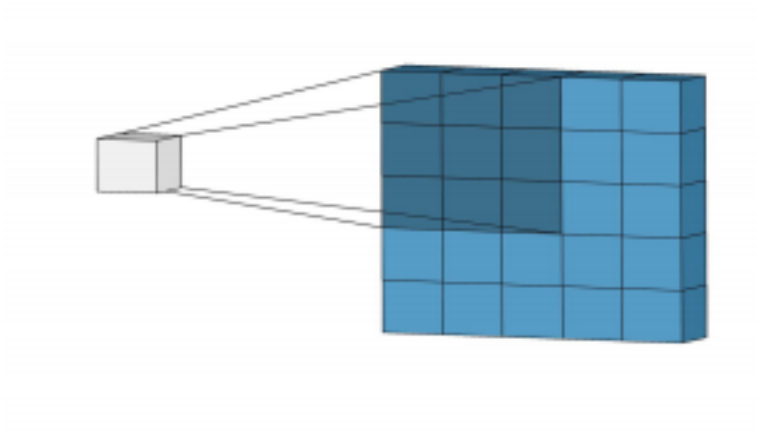
CNN are among the most popular neural network frameworks that are used in complex applications like Deep Learning models for Computer Vision(CV).

CNN is also known as ConvNet.

What are these convolutions?

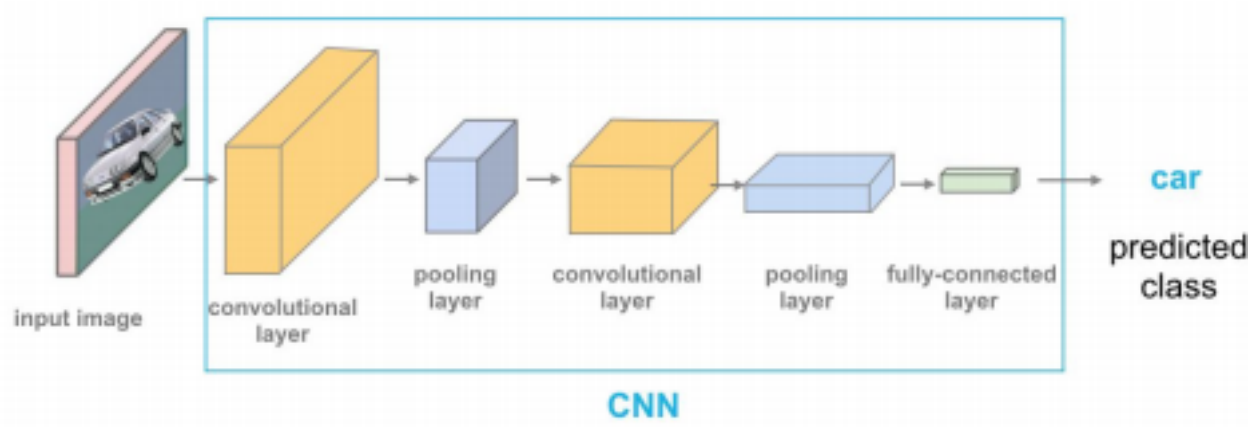
Convolutions are the building block for deep learning models in CV and many other applications.

- Convolutions are also called filters .
- It is a small tensor that can be multiplied over little sections of the main page.
- Different convolution captures different aspects of the original image.



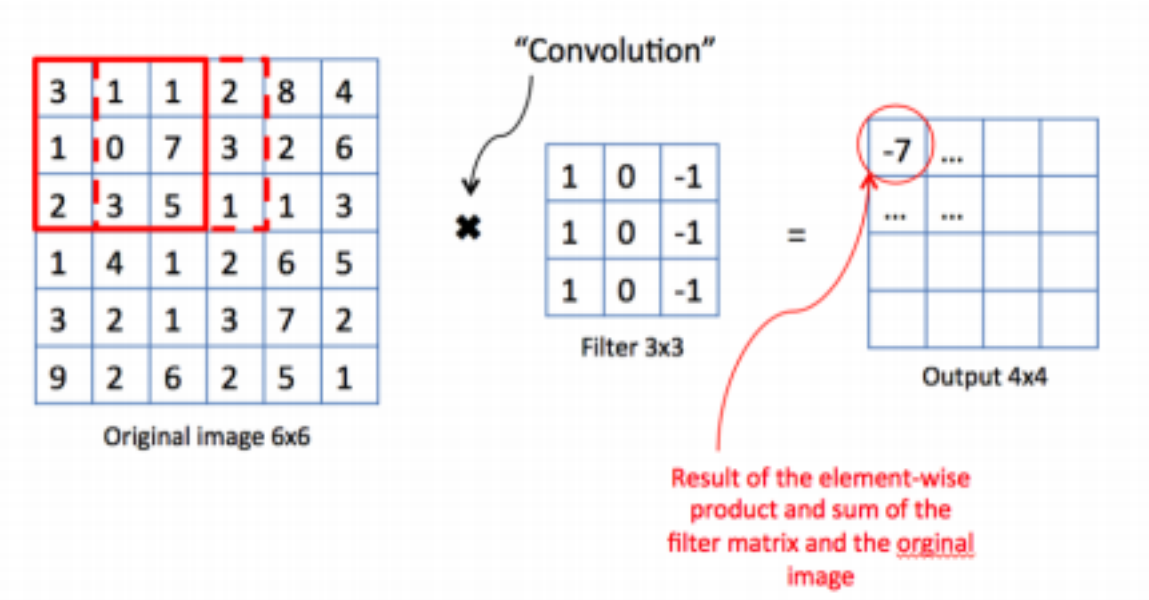
Layers in CNN

A CNN contains different layers as shown below.



Convolutional Layer

It has several filters that perform the convolution operation. The convolution operation is a dot product of original pixel values with weights defined in the filter. The results are summed up to one number that represents all the pixels that filter observed.



Hence, the feature maps are extracted by edge detection through this layer.

ReLu Layer

This layer:

➔ Performs element wise operation

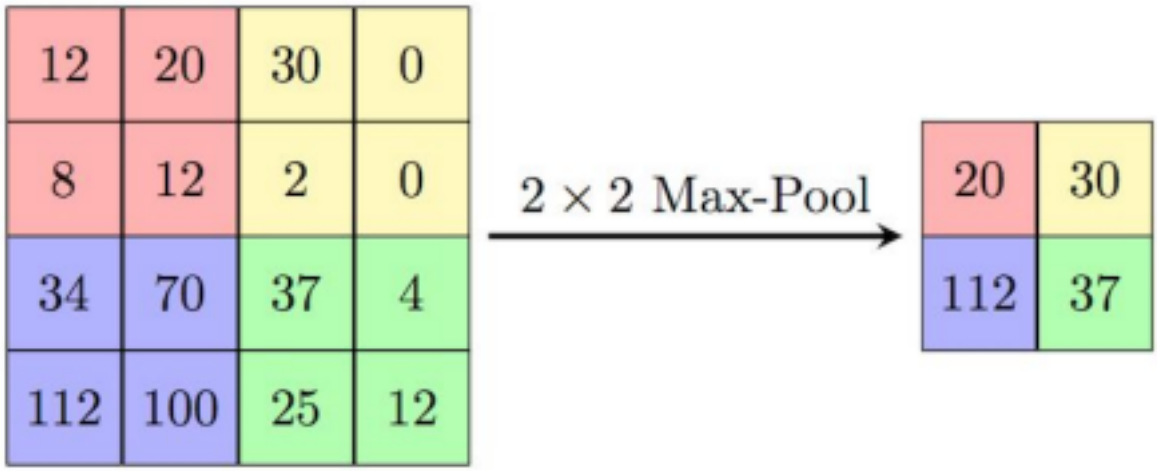
➔ Sets all -ve to 0

➔ Introduce non-linearity to the network

Hence, we get a rectified feature map through this layer.

Pooling Layer

Pooling is a down- sampling operation that reduces the dimensionality of the rectified feature map.



After applying the max-pooling we do flattening.

Flattening

In this we convert all the resultant 2D arrays from pooled vector map into a single long continuous linear vector.

Fully Connected Layer

The flattened matrix from the pooling layer is fed as input to this layer to classify the image.

Why Convolutions?

- Convolutional layers reduce the number of parameters speed up the training of the model significantly
- Parameter sharing
- Sparsity of connections

What is OpenCV ?

OpenCV is the huge open-source library for the computer vision, machine learning, and image processing and now it plays a major role in real -time operation which is very important in today's systems.By using it, one can process images and videos to identify objects, faces, or even handwriting of a human.

How to load an Image ?

```
import cv2 as cv

img = cv.imread(cv.samples.findFile("starry_night.jpg"))

if img is None:
        sys.exit("Could not read the image.")

cv.imshow("Display window", img)
k = cv.waitKey(0)

if k == ord("s"):
         cv.imwrite("starry_night.png", img)
```

How to load video or connect to a web camera ?

```
import numpy as np
import cv2

cap = cv2.VideoCapture(0)

while(True):

        # Capture frame-by-frame
         ret, frame = cap.read()

         # Our operations on the frame come here
         gray = cv2.cvtColor(frame, cv2.COLOR_BGR2GRAY)

         # Display the resulting frame
         cv2.imshow('frame',gray) #To close window press 'q'
         if cv2.waitKey(1) & 0xFF == ord('q'):
                 break

#When everything done, release the capture

cap.release()
cv2.destroyAllWindows()
```

Related Work

In face detection method, a face is detected from an image that has several attributes in it.

For face detection we require expression recognition, face tracking, and pose estimation. Given a solitary image, the challenge is to identify the face from the picture. Face detection is a difficult process because the faces change in size, shape, color, etc and they are not immutable.

It becomes a laborious job for opaque images implemented by some other thing not confronting the camera and so forth. To get the conclusive face we have to solve two major challenges :

1) Unavailability of sizably voluminous datasets containing both masked and unmasked faces,

2) Exclusion of facial expression in the covered area.

Utilizing the locally linear embedding(LLE) algorithm and the dictionaries trained on an immensely colossal pool of masked faces, synthesized mundane faces, several mis laid expressions can be recuperated and the ascendancy of facial cues can be mitigated to great extent. According to the work reported, convolutional neural networks(CNNs) in computer vision come with a strict constraint regarding the size of the input image.

Here the main challenge of the task is to detect the face from the image correctly and then identify if it has a mask on it or not. In order to perform surveillance tasks, the proposed method should also detect a face along with a mask in motion.

Dataset

Datasets that have been used for training CNN model contains two categories of images:

1. Face with mask
2. Face without mask

LITERATURE SURVEY

MAMATA S. KALAS,
REAL TIME FACE DETECTION AND TRACKING USING OPENCV,

International Journal of Soft Computing and Artificial Intelligence, ISSN: 2321-404X, Volume-2, Issue-1, May- 2014

Imported Package

A.TensorFlow

TensorFlow is an end-to-end open source platform for machine learning. It has a comprehensive, flexible ecosystem of tools, libraries and community resources that lets researchers push the state-of-the-art in ML and developers easily build and deploy ML powered applications

(i)Easy model building

Build and train ML models easily using intuitive high-level APIs like Keras with eager execution, which makes for immediate model iteration and easy debugging.

(ii)Robust ML production anywhere

Easily train and deploy models in the cloud, on-prem, in the browser, or on-device no matter what language you use.

(iii)Powerful experimentation for research

A simple and flexible architecture to take new ideas from concept to code, to state-of-the-art models, and to publication faster.

B. Keras

Keras gives fundamental reflections and building units for creation and transportation of ML arrangements with high iteration velocity. It takes full advantage of the scalability and cross-platform capabilities of TensorFlow. All the layers used in the CNN model are implemented using Keras. Along with the conversion of the class vector to the binary class matrix in data processing, it helps to compile the overall model.

Why is keras used in python ?

Keras is a neural networks library written in **Python** that is high-level in nature – which makes it extremely simple and intuitive to **use**. It works as a wrapper to low-level libraries like TensorFlow or Theano high-level neural networks library, written in **Python** that works as a wrapper to TensorFlow or Theano

C. OpenCV

OpenCV (Open Source Computer Vision Library), an open-source computer vision and ML software library, is utilized to differentiate and recognize faces, recognize objects, group movements in recordings, trace progressive modules, follow eye gesture, track camera actions, expel red eyes from pictures taken utilizing flash, find comparative pictures from an image database, perceive landscape and set up markers to overlay it with increased reality and so forth. The proposed method makes use of these features of OpenCV in resizing and color conversion of data images.

Proposed System

The proposed system focuses on how to identify the person on an image/video stream wearing a face mask with the help of computer vision and deep learning algorithm by using the OpenCV, Tensor flow , Keras and other python libraries.

Approach :

1. Train Deep learning model (MobileNetV2)
2. Apply mask detector over images / live video stream

The majority of the images were augmented by OpenCV. The set of images were already labeled mask and no mask.

The images that were present were of different sizes and resolutions, probably extracted from different sources or from machines (cameras) of different resolutions.

Face Mask Detection in webcam stream.

- The flow to identify the person in the webcam wearing the face mask or not. The process is two-fold.
    1. To identify the faces in the webcam.
    2. Classify the faces based on the mask.
       Identify the Face in the Webcam: To identify the faces a pre- trained model provided by the OpenCV framework was used. The model was trained using web images. OpenCV provides 2 models for this face detector:
        1. Floating-point 16 version of the original Caffe implementation.
        2. 8 bit quantized version using Tensor flow

The objective of developing biometric applications, such as facial recognition, has recently

become important in smart cities. Besides, many scientists and engineers around the world have focused on establishing increasingly robust and accurate algorithms and methods for these types of systems and their application in everyday life.

All types of security systems must protect all personal data. The most commonly used type for recognition is the password. However, through the development of information technologies and security algorithms, many systems are beginning to use many biometric factors for the recognition task.

These biometric factors make it possible to identify people's identities by their physiological or behavioral characteristics. They also provide several advantages, for example, the presence of a person in front of the sensor is sufficient, and there is no more need to remember several passwords or confidential codes anymore.

FACE RECOGNITION

Three basic steps are used to develop a robust face recognition system:

The face recognition system begins first with the localization of the human faces in a particular image. The purpose of this step is to determine if the input image contains human faces or not. The variations of illumination and facial expression can prevent proper face detection. To facilitate the design of a further face recognition system and make it more robust, pre- processing steps are performed. Many techniques are used to detect and locate the human face image, for example, Viola- Jones detector, histogram of oriented gradient (HOG), and principal component analysis (PCA). Also, the face detection step can be

used for video and image classification, object detection, region-of-interest detection, and so on.

1.Feature Extraction:

The main function of this step is to extract the features of the face images detected in the detection step. This step

represents a face with a set of features vectors called a signature that describes the prominent features of the face image such as mouth, nose, and eyes with their geometry distribution . Each face is characterized by its structure, size, and shape, which allow it to be identified. Several techniques involve extracting the shape of the mouth, eyes, or nose to identify the face using the size and distance . HOG, Eigen face , independent component analysis , linear discriminant analysis (LDA) , scale invariant feature transform (SIFT) , Gabor filter, local phase quantization (LPQ) , Haar wavelets, Fourier transforms , and local binary pattern (LBP) techniques are widely used to extract the face features.

2.Face Recognition:

This step considers the features extracted from the background during the feature extraction step and compares it with known faces stored in a specific database. There are two general applications of face recognition, one is called identification and another one is called verification. During the identification step, a test face is compared with a set of faces aiming to find the most likely match. During the identification step, a test face is compared with a known face in the database in order to make the acceptance or rejection decision. Correlation filters (CFs) , convolutional neural network (CNN) , and also k-nearest neighbor (K-NN) are known to effectively address this task.

The classification of the images is done by training the model in 2 phases:

Phase 1: Face mask dataset is loaded into the system. Different classifiers like MobileNetV2, ResNet50, and VGG16 are used to generate a trained model.

Phase2: Load the face mask classifier model.

Detect faces in the images/video stream. Apply the classifier to each face RoI. Classify the images to be With Mask and Without Mask with Confidence.

This system may then be interfaced with

Case 1: Existing access control system so that violators can be restricted.

Case 2: There could be some scenarios in workplaces where people may forget or just put off the mask when it becomes uneasy for them to get accustomed to the new face masks. In such cases, alarm by the system may be disturbing other workers. Hence the concerned

authorities can take proper measures to alert the user so that they can wear the mask again.

Methodology	Computer Vision	Convolutional Neural Network
Approach	When a computer looks at an image with a specific goal, the irrelevant information is not taken into account. This helps reduce the types of bias that humans might introduce to a process, whether intentionally or unintentionally.	Accuracy in image recognition problems. This helps us to get the results accurate and differentiate between mask and no mask.
	When the device fails because of a virus or other software issues, it is highly probable that Computer Vision and image processing will fail.	CNN automatically detects the important features without any human supervision.
		If there is no good GPU they are quite slow to train (for complex tasks). They need a lot of training data.
	Time and error rate are reduced in the process of Computer Imagining. It reduces the cost of hiring and training special staff (human force) to do the activities that computers do.	It is computationally very expensive and time consuming to train with traditional CPUs.
		Once we train the system, the predictions are pretty fast.

Result And Analysis

The model is trained, validated and tested upon datasets. Corresponding to the dataset the method attains avg accuracy up to 95.77%. Fig.1 depicts how this optimized accuracy mitigates the cost of error. One of the main reasons behind achieving this accuracy lies in MaxPooling. It provides rudimentary translation invariance to the internal representation along with the reduction in the number of parameters the model has to learn. This sample-based discretization process down-samples the input representation consisting of image, by reducing its dimensionality. Number of neurons has an optimized value of 64 which is not too high. A much higher number and filter can lead to worse performance. The optimized filter values and pool_size help to filter out the main portion (face) of the image to detect the existence of the mask correctly without causing over-fitting.

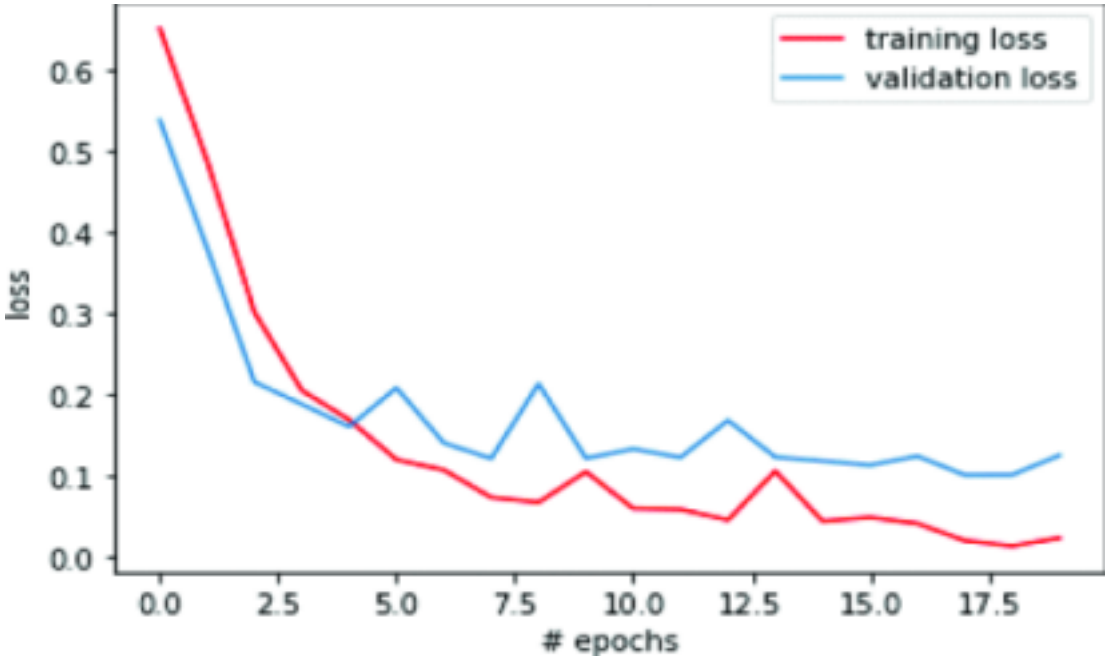


Fig 1 : epochs vs loss

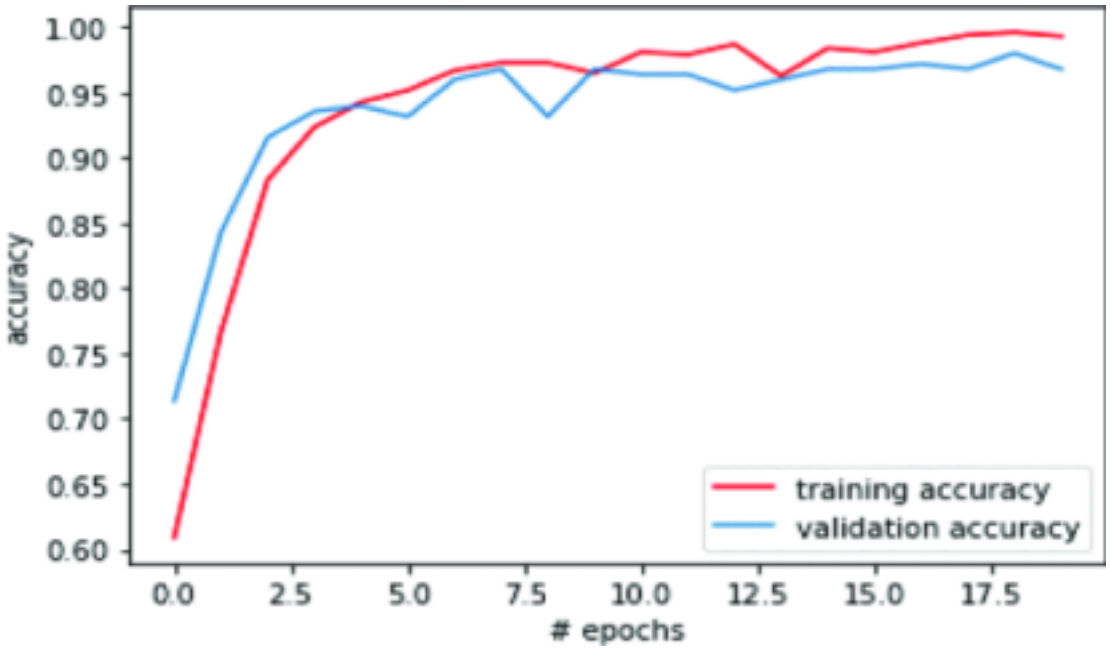


Fig 2: Epochs vs Accuracy

Conclusion

Different methods and approaches of face mask detection and recognition have been reviewed in this work. In comparison, Haar-like features are digital image features used in object recognition. They owe their name to their intuitive similarity with Haar wavelets and were used in the first real- time face detector. The key advantage of a Haar-like feature over most other features is its calculation speed. Adaboost can be less susceptible to the overfitting problem than most learning algorithms. Bad feature of adaptive boosting is its sensitivity to noisy data and outliers. In real-world scenarios human faces might be occluded by other objects such as facial masks. This makes the face recognition process a very challenging task. Deep learning-based method and quantization-based technique achieves a high recognition performance. MobileNetV2 is a very effective feature extractor for object detection and segmentation. MobileNetV2 provides a very efficient mobile-oriented model that can be used as a base for many visual recognition tasks. For the best of our knowledge, this work addresses the problem of masked face recognition and different approaches during COVID19 pandemic. It is worth stating that this study is not limited to this pandemic period since a lot of people are self-aware constantly, they take care of their health and wear masks to protect themselves against pollution and to reduce other pathogens transmission.

REFERENCES

1. Mamata s. Kalas, REAL TIME FACE DETECTION AND TRACKING USING OPENCV , International Journal of Soft Computing and Artificial Intelligence, ISSN: 2321-404X, Volume-2, Issue-1, May-2014
2. Walid Hariri, Efficient Masked Face Recognition Method during the COVID-19 Pandemic, pp.1-7, July 2020
3. Vinitha.V1, Velantina.V2, Covid-19 facemask detection with deep learning and computer vision International Research Journal of Engineering and Technology (IRJET), Volume: 07, pp.1-6, Aug 2020.
4. Yassin Kortli, Maher Jridi, Ayman Al Falou, and Mohamed Atri, face recognition systems: a survey, 20(2): 342, pp.1-10, 2020 Jan 7.
5. Vinitha.V1, Velantina.V2, COVID-19 FACEMASK DETECTION WITH DEEP LEARNING AND COMPUTER VISION International Research Journal of Engineering and Technology (IRJET), Volume: 07, pp.1-6, Aug 2020.

1.

Thank You