MAULANA AZAD
NATIONAL INSTITUTE OF TECHNOLOGY
BHOPAL, INDIA, 462003

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

REALTIME FACE MASK DETECTOR

Minor Project Report
Semester VI

Submitted by:

SHUBHAM JHALANI	181112077
SANDEEP KUMAR	181112084
MANSI BADOLE	181112082
YOGESH KUMAR AHIRWAR	181112089

Under the Guidance of
Prof. Sanyam Shukla
DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
Session: 2021-2021

MAULANA AZAD

NATIONAL INSTITUTE OF TECHNOLOGY

BHOPAL, INDIA, 462003

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

CERTIFICATE

This is to certify that the project report carried out on **“REALTIME FACE MASK DETECTOR”** by the 3rd year students:

SHUBHAM JHALANI	181112077
SANDEEP KUMAR	181112084
MANSI BADOLE	181112082
YOGESH KUMAR AHIRWAR	181112089

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

Prof. Sanyam Shukla
(Minor Project Mentor)

DECLARATION

We, hereby declare that the following report which is being presented in the Minor Project Documentation Entitled as **“REALTIME FACE MASK DETECTOR”** is an authentic documentation of our own original work and to best of our knowledge. The following 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.

SHUBHAM JHALANI	181112077
SANDEEP KUMAR	181112084
MANSI BADOLE	181112082
YOGESH KUMAR AHIRWAR	181112089

ACKNOWLEDGEMENT

With due respect, we express our deep sense of gratitude to our respected guide and coordinator **Prof. Sanyam Shukla** for his valuable help and guidance. We are thankful for the encouragement that he has given us in completing this project successfully.

It is imperative for us to mention the fact that the report of minor project could not have been accomplished without the periodic suggestions and advice of our project guide **Prof. Sanyam Shukla** 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 utilize all the necessary 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

The end of 2019 witnessed the outbreak of Coronavirus Disease 2019 (COVID-19), which has continued to be the cause of plight for millions of lives and businesses even in 2020. As the world recovers from the pandemic and plans to return to a state of normalcy, there is a wave of anxiety among all individuals, especially those who intend to resume in person activity. Studies have proved that wearing a **face mask** significantly reduces the risk of viral transmission as well as provides a sense of protection. However, it is not feasible to manually track the implementation of this policy. Technology holds the key here.

We introduce a Deep Learning based system that can detect instances where face masks are not used properly. Our system consists of a dual stage **Convolutional Neural Network (CNN)** architecture capable of detecting masked and unmasked faces with extra functionality that it also recognises the faces if a person is not wearing a mask and sends an email to them and adds them into a CSV file and it can be integrated with pre-installed CCTV cameras. This will help track safety violations, promote the use of face masks, and ensure a safe working environment.

Keywords: Artificial Neural Network; Convolutional Neural Network; Object Detection; Object Tracking; Face Recognition; COVID-19; Face Masks; Safety Improvement

TABLE OF CONTENTS

INTRODUCTION

The year 2020 has shown mankind some mind-boggling series of events amongst which the COVID- 19 pandemic is the most life-changing event which has startled the world since the year began. Affecting the health and lives of masses, COVID-19 has called for strict measures to be followed in order to prevent the spread of disease. From the very basic hygiene standards to the treatments in the hospitals, people are doing all they can for their own and the society's safety; face masks are one of the personal protective equipment. People wear face masks once they step out of their homes and authorities strictly ensure that people are wearing face masks while they are in groups and public places.

To monitor that people are following this basic safety principle, a strategy should be developed. A face mask detector system can be implemented to check this. Face mask detection means to identify whether a person is wearing a mask or not. The first step to recognize the presence of a mask on the face is to detect the face, which makes the strategy divided into two parts: to detect faces and to detect masks on those faces. Face detection is one of the applications of object detection and can be used in many areas like security, biometrics, law enforcement and more. There are many detector systems developed around the world and being implemented. However, all this science needs optimization; a better, more precise detector, because the world cannot afford any more increase in corona cases.

In this project, we will be developing a face mask detector that is able to distinguish between faces with masks and faces with no masks. If a person is not wearing a mask then the model will recognize its face and send an email to them and add an entry of them with time in a CSV file. In this report, we have proposed a detector which does face detection and a neural network to detect the presence of a face mask. The implementation of the algorithm is on images, videos and live video streams.

DEEP LEARNING

During recent years, deep learning has become somewhat of a buzzword in the tech community. We always seem to hear about it in news regarding AI, and yet most people don't actually know what it is! In this article, I'll be demystifying the buzzword that is deep learning, and providing an intuition of how it works.

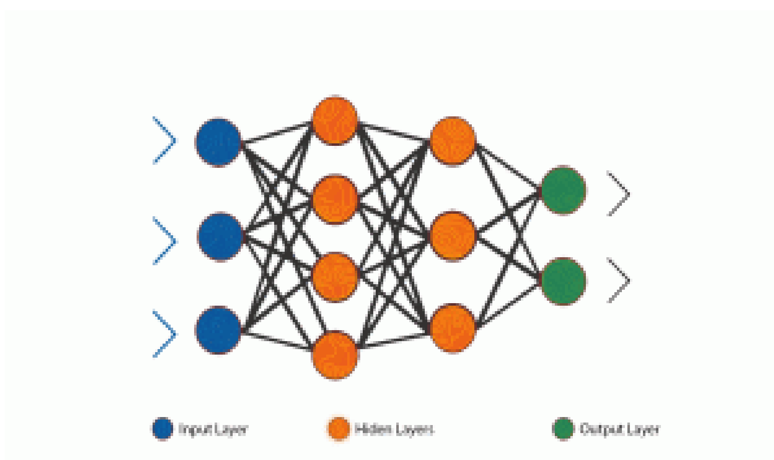
Building the Intuition

Generally speaking, deep learning is a machine learning method that takes in an input X, and uses it to predict an output of Y. As an example, given the stock prices of the past week as input, my deep learning algorithm will try to predict the stock price of the next day.

Given a large dataset of input and output pairs, a deep learning algorithm will try to minimize the difference between its prediction and expected output. By doing this, it tries to learn the association/pattern between given inputs and outputs — this in turn allows a deep learning model to generalize to inputs that it hasn't seen before.

Artificial Neural Network (ANN) – What is an ANN and why should you use it?

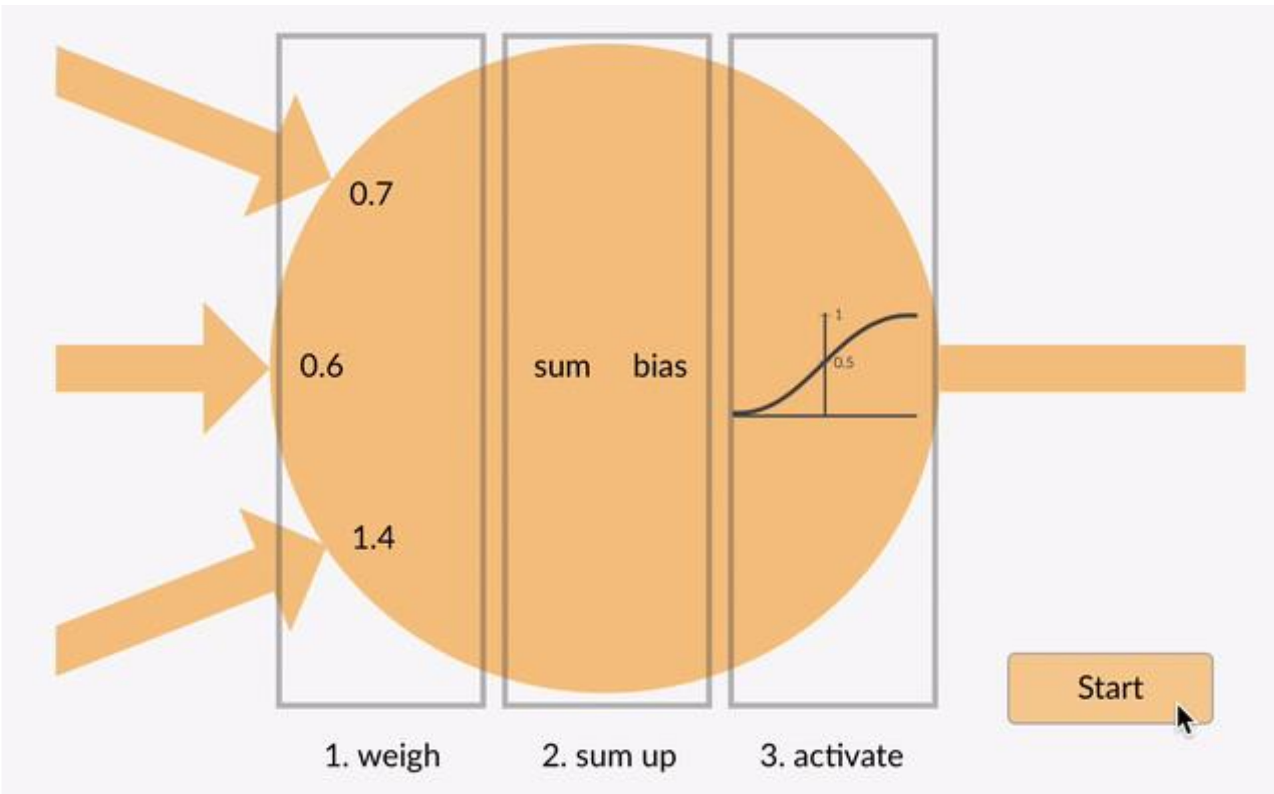
A single perceptron (or neuron) can be imagined as a Logistic Regression. Artificial Neural Network, or ANN, is a group of multiple perceptrons/ neurons at each layer. ANN is also known as a Feed-Forward Neural network because inputs are processed only in the forward direction



As you can see here, ANN consists of 3 layers – Input, Hidden and Output. The input layer accepts the inputs, the hidden layer processes the inputs, and the output layer produces the result. Essentially, each layer tries to learn certain weights. ANN can be used to solve problems related to: Tabular data, Image data, Text data.

Advantages of Artificial Neural Network (ANN) –

The Artificial Neural Network is capable of learning any nonlinear function. Hence, these networks are popularly known as Universal Function Approximators. ANNs have the capacity to learn weights that map any input to the output. One of the main reasons behind universal approximation is the activation function. Activation functions introduce nonlinear properties to the network. This helps the network learn any complex relationship between input and output.



Relationship between input and output.

As you can see here, the output at each neuron is the activation of a weighted sum of inputs. But wait – what happens if there is no activation function? The network only learns the linear function and can never learn complex relationships. That's why:

An activation function is a powerhouse of ANN!

BACKPROPAGATION

Backpropagation algorithm is probably the most fundamental building block in a neural network.

The algorithm is used to effectively train a neural network through a method called chain rule. In simple terms, after each forward pass through a network, backpropagation performs a backward pass while adjusting the model's parameters (weights and biases).

In other words, **backpropagation aims to minimize the cost function by adjusting the network's weights and biases.** The level of adjustment is determined by the gradients of the cost function with respect to those parameters.

Gradient of a function C(x_1, x_2, ..., x_m) in point x is a vector of the partial derivatives of C in x.

$$\frac{\partial C}{\partial x} = [\frac{\partial C}{\partial x_1}, \frac{\partial C}{\partial x_2}, \ldots, \frac{\partial C}{\partial x_m}]$$

The derivative of a function C measures the sensitivity to change of the function value (output value) with respect to a change in its argument x (input value). In other words, the derivative tells us the direction C is going. The gradient shows how much the parameter x needs to change (in positive or negative direction) to minimize C. Computing those gradients happens using a technique called chain rule.

$$\delta_j^l = \frac{\partial C}{\partial z_j^l} \qquad local\ gradient$$

The *"local gradient"* can easily be determined using the chain rule.

For a single weight *(w_jk)^l,* the gradient is:

$$\frac{\partial C}{\partial w_{jk}^{l}} = \frac{\partial C}{\partial z_j^{l}} \frac{\partial z_j^{l}}{\partial w_{jk}^{l}} \qquad \textit{chain rule}$$

$$z_j^{l} = \sum_{k=1}^{m} w_{jk}^{l} a_k^{l-1} + b_j^{l} \qquad \textit{by definition}$$

$$m \; - \; \textit{number of neurons in } l-1 \textit{ layer}$$

$$\frac{\partial z_j^{l}}{\partial w_{jk}^{l}} = a_k^{l-1} \qquad \textit{by differentiation (calculating derivative)}$$

$$\frac{\partial C}{\partial w_{jk}^{l}} = \frac{\partial C}{\partial z_j^{l}} a_k^{l-1} \qquad \textit{final value}$$

The gradients allow us to optimize the model's parameters:

$$\textit{while (termination condition not met)}$$

$$w := w - \epsilon \frac{\partial C}{\partial w}$$

$$b := b - \epsilon \frac{\partial C}{\partial b}$$

$$\textit{end}$$

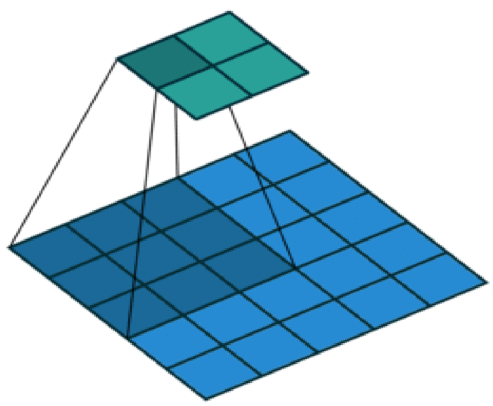
Initial values of w and b are randomly chosen.

- Epsilon (e) is the learning rate. It determines the gradient's influence.
- w and b are matrix representations of the weights and biases. Derivative of C in w or b can be calculated using partial derivatives of C in the individual weights or biases.
- Termination condition is met once the cost function is minimized.

Convolution Neural Network (CNN) – What is a CNN and Why Should you use it?

Convolutional neural networks (CNN) are all the rage in the deep learning community right now. These CNN models are being used across different applications and domains, and they're especially prevalent in image and video processing projects.

The building blocks of CNNs are filters a.k.a. kernels. Kernels are used to extract the relevant features from the input using the convolution operation. Let's try to grasp the importance of filters using images as input data. Convolving an image with filters results in a feature map:



Advantages of Convolution Neural Network (CNN)

- CNN learns the filters automatically without mentioning it explicitly. These filters help in extracting the right and relevant features from the input data

- CNN captures the spatial features from an image. Spatial features refer to the arrangement of pixels and the relationship between them in an image. They help us in identifying the object accurately, the location of an object, as well as its relation with other objects in an image
- CNN also follows the concept of parameter sharing. A single filter is applied across different parts of an input to produce a feature map:

Layers in CNN

1. **Convolutional Layer**:

This layer is the first layer that is used to extract the various features from the input images. In this layer, the mathematical operation of convolution is performed between the input image and a filter of a particular size MxM. By sliding the filter over the input image, the dot product is taken between the filter and the parts of the input image with respect to the size of the filter (MxM).

The output is termed as the Feature map which gives us information about the image such as the corners and edges. Later, this feature map is fed to other layers to learn several other features of the input image.

2. Pooling Layer: The primary aim of this layer is to decrease the size of the convolved feature map to reduce the computational costs. This is performed by decreasing the connections between layers and independently operates on each feature map.

Depending upon the method used, there are several types of Pooling operations.

- In **Max Pooling**, the largest element is taken from the feature map.
- **Average Pooling** calculates the average of the elements in a predefined size Image section.
- The total sum of the elements in the predefined section is computed in **Sum Pooling**.

3. Fully Connected Layer -

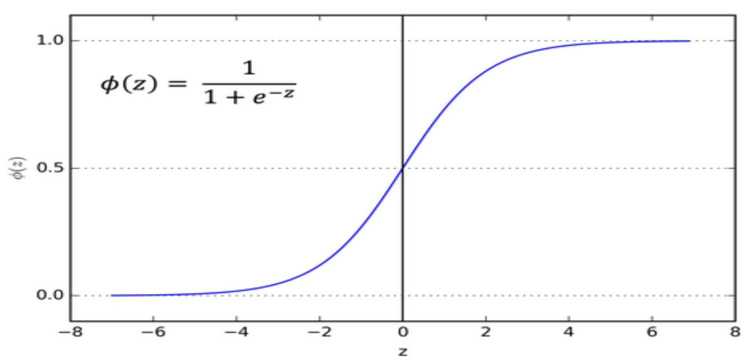
The Fully Connected (FC) layer consists of the weights and biases along with the neurons and is used to connect the neurons between two different layers. These layers are usually placed before the output layer and form the last few layers of a CNN Architecture.

3.1 Activation Functions -

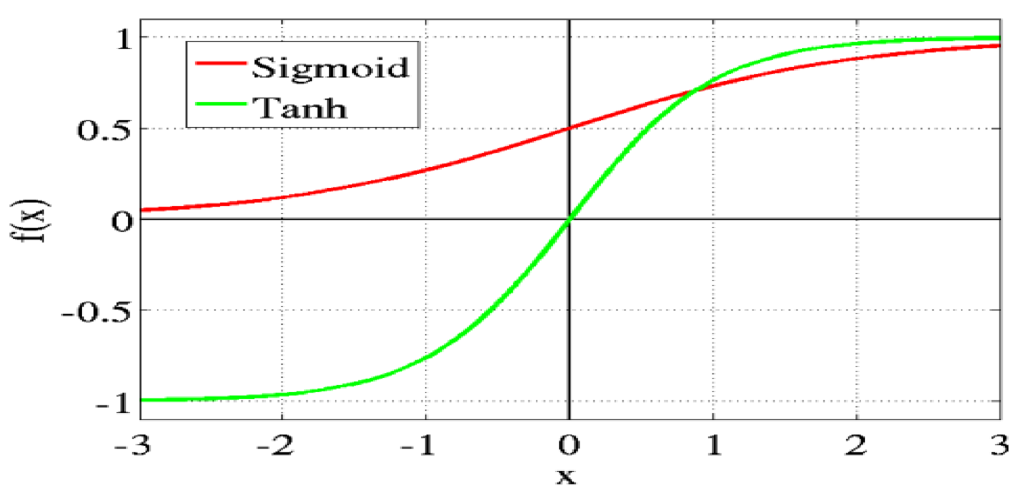
Finally, one of the most important parameters of the CNN model is the activation function. They are used to learn and approximate any kind of continuous and complex relationship between variables of the network. In simple words, it decides which information of the model should fire in the forward direction and which ones should not at the end of the network.

It adds non-linearity to the network. There are several commonly used activation functions such as the *ReLU, Softmax, tanH* and the *Sigmoid* functions. Each of these functions have a specific usage. For a binary classification CNN model, sigmoid and *SoftMax* functions are preferred and for a multi-class classification, generally *SoftMax* is used.

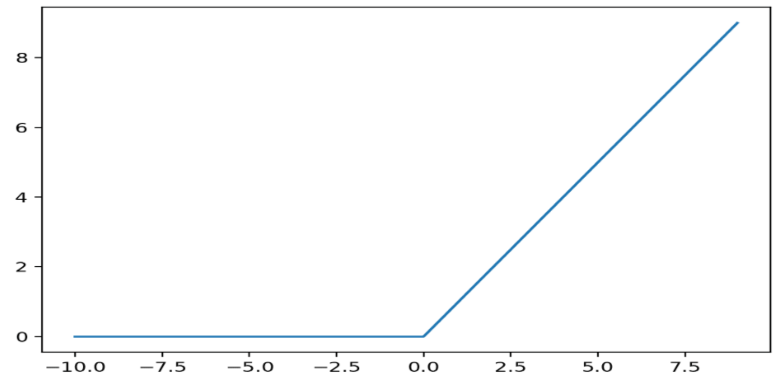
3.1.1. Sigmoid or Logistic Activation Function



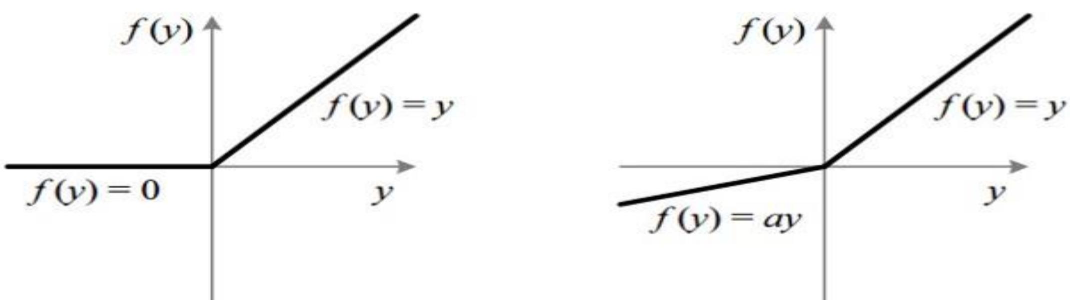
3.1.2. Tanh or hyperbolic tangent Activation Function

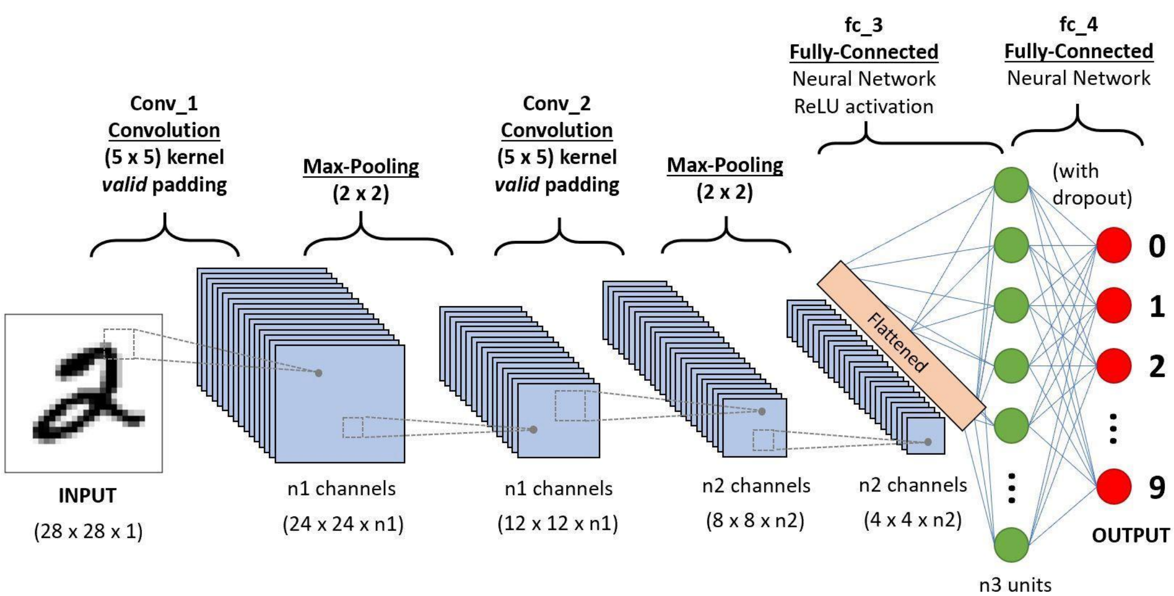


3.1.3. ReLU (Rectified Linear Unit) Activation Function



3.1.4. Leaky ReLU





Computer Vision

Computer Vision can be defined as a discipline that explains how to reconstruct, interrupt, and understand a 3D scene from its 2D images, in terms of the properties of the structure present in the scene. It deals with modeling and replicating human vision using computer software and hardware.

Computer Vision overlaps significantly with the following fields –

- **Image Processing** – It focuses on image manipulation.
- **Pattern Recognition** – It explains various techniques to classify patterns.
- **Photogrammetry** – It is concerned with obtaining accurate measurements from images

Computer Vision Vs Image Processing

Image processing deals with image-to-image transformation. The input and output of image processing are both images.

Computer vision is the construction of explicit, meaningful descriptions of physical objects from their image. The output of computer vision is a description or an interpretation of structures in a 3D scene.

OpenCV

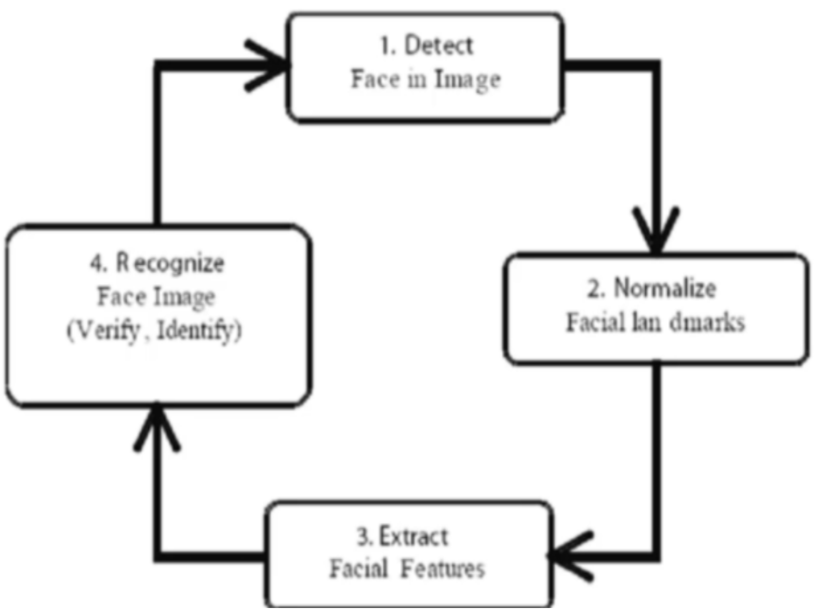
OpenCV is a cross-platform library using which we can develop real-time computer vision applications. It mainly focuses on image processing, video capture and analysis including features like face detection and object detection.

Face Recognition

A facial recognition system is a technology capable of matching a human face from a digital image or a video frame against a database of faces, typically employed to authenticate users through ID verification services, works by pinpointing and measuring facial features from a given image

Face Recognition basically involves four steps:

1. Face Detection
2. Face Alignment
3. Feature Extraction
4. Face Recognition



1. Face Detection

Face detection is the process of automatically locating human faces in visual media (digital images or video).

Two popular types of Face Detectors image recognition library supports:

1. HOG Face Detector
2. CNN based Face Detector included in Dlib

HOG Face Detector

- A HOG (Histogram of Oriented Gradients) is generally used for object detection.
- A HOG relies on distribution of intensity gradients or edge directions.

Pros

- Faster while using in CPU, Very light weight
- Works under small occlusion

Cons

- Min size of face should be 80x80 px.
- Does not work for side faces or high extremes of non-frontal faces, like looking down or up.
- Doesn't work under heavy occlusion (sunglasses, hat/cap, scarf, beard etc).

CNN Face Detector in Dlib

- This method uses an Object Detector with CNN based features.
- The training process is simple. No need for a large amount of training data.

Pros

- Detect Multiple Face orientations
- Works with medium occlusion
- Fast on GPU

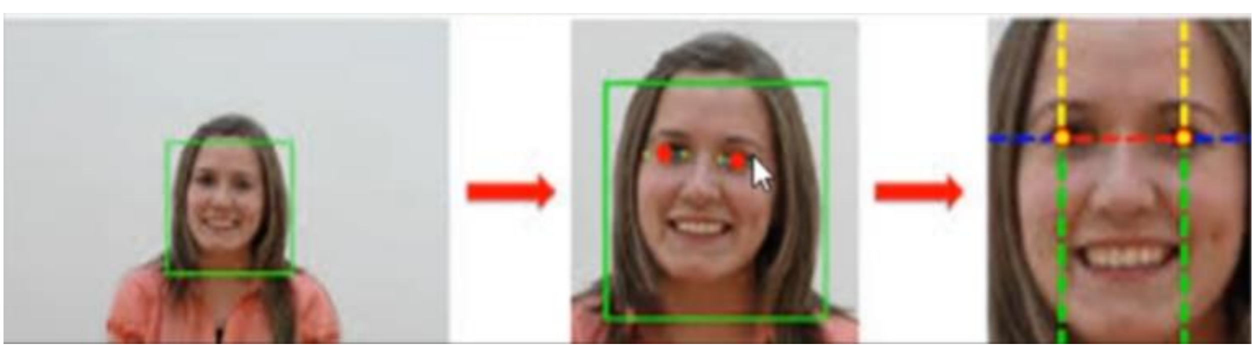
Cons

- Very slow on the CPU.

2. Face Alignment

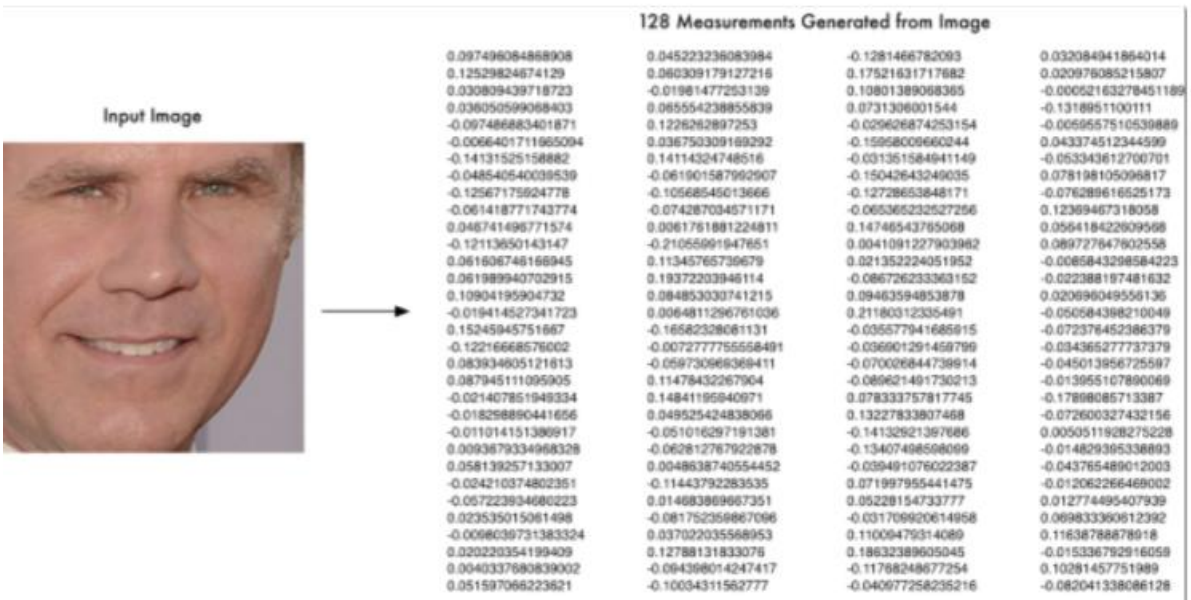
Face Alignment plays an important role in face recognition, because it is often used in pre-processing steps.

The automatic detection of face and eye region and align face based on translation, scale and rotation.



3. Feature Extraction

Pass the pre-processed and detected face into a pre-trained face recognizer that is capable of producing 128 embeddings.



4. Face Recognition

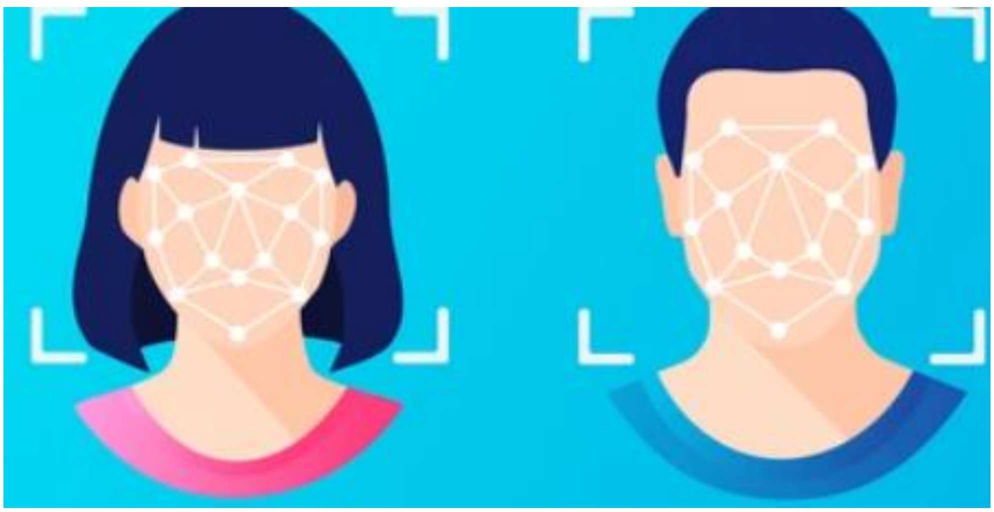
Find the person in our database of known people who has the closest measurement of the test image.

Real time Face Recognition

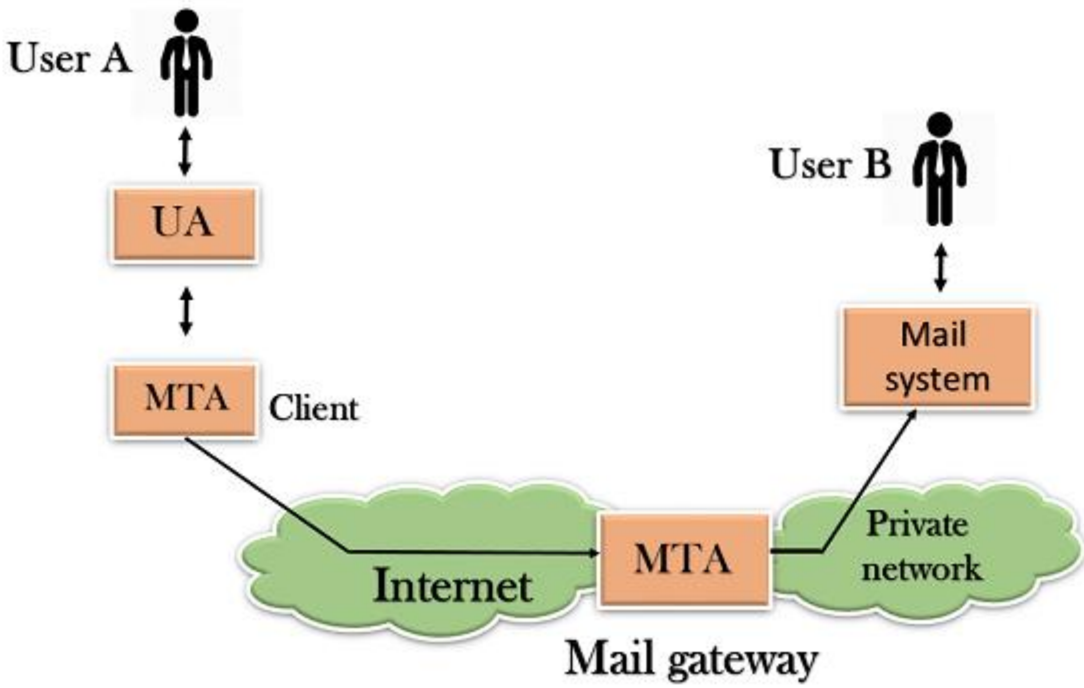
A Real time face recognition system can identify a face from a video frame.

The Mechanism is very simple

- ➢ Get the Webcam video
- ➢ Loop through every single frame video image and do Face Recognition



SMTP (SIMPLE MAIL TRANSFER PROTOCOL): -



- **SMTP** is a set of communication guidelines that allows a software to transmit an e-mail over the internet using port number 465 and 587.
- It provides email exchange between users of the same or different computers.
- Its main purpose is to set up communication rules between servers. The servers have a way of identifying themselves and announcing what kind of communication they are trying to perform.

LITERATURE SURVEY

Face detection is a computer technology that determines the location and size of a human face in an arbitrary (digital) image. The facial features are detected and any other objects like trees, buildings and bodies etc. are ignored from the digital image. It can be regarded as a specific 'case of object-class detection, where the task is finding the location and sizes of all objects in an image that belong to a given class. Face detection can be regarded as a more general case of face localization. In face localization, the task is to find the locations and sizes of a known number of faces (usually one). Basically, there are two types of approaches to detect facial parts in the given image i.e. feature base and image-based approach. Feature base approach tries to extract features of the image and match it against the knowledge of the face features. While the image base approach tries to get the best match between training and testing images.

CONCLUSION

The computational models, which were implemented in this project, were chosen after extensive research, and the successful testing results confirm that the choices made by the researcher were reliable. The system with manual face detection and automatic face recognition did not have a recognition accuracy over 90%, due to the limited number of eigenfaces that were used for the PCA transform. This system was tested under very robust conditions in this experimental study and it is envisaged that real-world performance will be far more accurate. The fully automated frontal view face detection system displayed virtually perfect accuracy and in the researcher's opinion further work need not be conducted in this area. The fully automated face detection and recognition system was not robust enough to achieve a high recognition accuracy.

The only reason for this was the face recognition subsystem did not display even a slight degree of invariance to scale, rotate or shift errors of the segmented face image. This was one of the system requirements identified in section 2.3. However, if some sort of further processing, such as an eye detection technique, was implemented to further normalize the segmented face image, performance will increase to levels comparable to the manual face detection and recognition system. Implementing an eye detection technique would be a minor extension to the implemented system and would not require a great deal of additional research.

All other implemented systems displayed commendable results and reflect well on the deformable template and Principal Component Analysis strategies. The most suitable real-world applications for face detection and recognition systems are for mugshot matching and surveillance. There are better techniques such as iris or retina recognition and face recognition using the thermal spectrum for user access and user verification applications since these need a very high degree of accuracy. The real-time automated pose invariant face detection and recognition system proposed in chapter seven would be ideal for crowd surveillance applications. If such a system were widely implemented its potential for locating and tracking suspects for law enforcement agencies is immense.

Libraries Used

- NumPy : For Numerical Computation
- Pandas : For Importing Files
- TensorFlow: Used as Backend to run the program
- Keras : To Implement the CNN Libraries and Image Processing
    - .models : To initialize the CNN
    - .layers : To Implement all layers (Convolution, MaxPooling, Flatten, Fully Connected Layer)
    - .callbacks : To store the model result at epoch for further use(ModelCheckpoint)
    - .preprocessing : Used to Process the images, text etc.
        - . image : Used to Process images
- OpenCV : To be Used for Live interaction to user(Enable Webcam to detect mask)
- dlib : It's C++ toolkit use for facial and its landmark detection
- Face_recognition : Used to recognise face using dlib's face recognition feature.
- Pickle : Used to dump or extract variables in memory.
- Smtplib : Used to Send email to the person.
- csv : Used to read/write data in csv file
- os : Used for some os command directly in python

LANGUAGES AND SOFTWARE USED

Python : version(3.8) is used.

Spyder : version (4.1.4) to run the Python Program installed via Anaconda

References

- https://towardsdatascience.com/a-comprehensive-guide-to-convolutional-neural-networks-the-eli5-way-3bd2b1164a53
- https://www.udemy.com/course/computer-vision-face-recognition-quick-starter-in-python/
- https://opencv.org/
- https://www.tensorflow.org/learn
- https://keras.io/getting_started/
- https://pypi.org/project/face-recognition/

- Z. Zhu, Z. Wang, D. Li, Y. Zhu and W. Du, "**Geometric Structural Ensemble Learning for Imbalanced Problems**," in *IEEE Transactions on Cybernetics*, vol. 50, no. 4, pp. 1617-1629, April 2020, doi: 10.1109/TCYB.2018.2877663.

URL:https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=8528854&isnumber=9011772

- D. Huang, C. Wang and J. Lai, "**Locally Weighted Ensemble Clustering**," in IEEE Transactions on Cybernetics, vol. 48, no. 5, pp. 1460-1473, May 2018, doi: 10.1109/TCYB.2017.2702343.

URL:https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=7932479&isnumber=8330726

- S. Li, K. Jia, Y. Wen, T. Liu and D. Tao, "**Orthogonal Deep Neural Networks**," in IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 43, no. 4, pp. 1352-1368, 1 April 2021, doi: 10.1109/TPAMI.2019.2948352.

URL: https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=8877742&isnumber=9370031