Face detections of a person wearing COVID mask using Image Processing

A Dissertation Submitted to the University of Hyderabad
in Partial Fulfillment of the Degree of

Master of Technology
in
Artificial Intelligence

By
Vamshi Bukya
19MCMI06

School of Computer and Information Sciences
University of Hyderabad
Gachibowli, Hyderabad - 500 046
Telangana, India

July , 2021

CERTIFICATE

This is to certify that the Thesis entitled "**Face detections of a person wearing COVID mask using Image Processing**" submitted by **Vamshi Bukya**, bearing Reg. No. 19MCMI06, in partial fulfillment of the requirements for the award of Master of Technology in Artificial Intelligence is a bonafide work carried out by him under my supervision and guidance.

The Thesis has not been submitted previously in part or in full to this or any other University or Institution for the award of any degree or diploma.

Professor Arun Agarwal
Professor Raghavendra Rao C
Professor Rajeev Wankar
Supervisor
SCIS
University of Hyderabad

Prof. Chakravarthy Bhagvati
Dean
School of CIS
University of Hyderabad

DECLARATION

I, Vamshi Bukya hereby declare that this Thesis entitled "**Face detections of a person wearing COVID mask using Image Processing** " submitted by me under the guidance and supervision of **Professor Arun Agarwal,Professor Raghavendra Rao C,Professor Rajeev Wankar** is a bonafide work. I also declare that it has not been submitted previously in part or in full to this University or other University or Institution for the award of any degree or diploma. I do not agree to have my Thesis deposited in Shodganga/INFLIBNET untill the work has been published.

A report on plagiarism statistics from the University Librarian is enclosed.

Name: Vamshi Bukya

Date:

Signature of the student

Reg. No: 19MCMI06

//Countersigned//

Signature of the Supervisor(s)

Professor Arun Agarwal	Professor Raghavendra Rao C	Professor Rajeev Wankar
Supervisor	Supervisor	Supervisor
SCIS	SCIS	SCIS
University of Hyderabad	University of Hyderabad	University of Hyderabad

ACKNOWLEDGEMENTS

I would like to thank my research guides Professor Arun Agarwal,Professor Raghavendra Rao C and Professor Rajeev Wankar, who gave me the opportunity and resources to pursue this line of research. I would also like to thank my friends and family for their support throughout the period of research. Without whom this research project would not have been possible on multiple levels.

Vamshi Bukya

Abstract

According to a report, over one percent of the world population infected with COVID19 in the last 365 days. To avoid the spread of the infection, WHO(World Health Organization) suggested individuals wear a face mask and maintain social distance. The use of face masks raised concerns about the accuracy of the facial recognition system used for office attendance, unlocking phones, etc. Masked faces make it difficult to recognize the individual.

The objective of this study is to figure out who is wearing a face mask. There are two steps in recognizing a person's identity they are face detection and face recognition. The faces are detected using Multi-Task Cascaded Convolutional Neural Networks (MTCNN) trained on the WIDER FACE dataset. Face descriptors were obtained from MTCNN's suggested Region of Interest using the model VGGFACE2. The MS-Celeb-1M dataset was used to train the VGGFACE2. Using facial descriptors, it is possible to identify a person. A cosine distance criterion is utilized to verify individuals, and The K-Nearest Neighboring classifier is used to determine a person's identity. This research provided better results in identifying a person wearing a face mask.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 The Problem

Individuals can transmit the COVID-19 virus to others even if they show no signs or symptoms, according to the World Health Organization (WHO). According to a paper's mathematical model, 40–80 percent of transmission begins with people who have no symptoms. The WHO advises people to look after someone who has COVID-19, and those who experience symptoms such as coughing and sniffling should use a face mask. The WHO (World Health Organization) suggested that everyone use a face mask to prevent the COVID 19 virus from spreading. As a result, people must devise a way to work during an epidemic by employing precautions such as social distancing and the use of face masks.Hospitals and other healthcare institutions, as well as workplaces, grocery stores, and pharmacies, may be among these sites.

1.2 Problem Definition

People used face masks during the covid epidemic, which created significant issues in face recognition. Is it feasible to build a model that uses both masked and unmasked faces as input and predicts the identification of unknowns?

1.3 Current Solution

Faces may be detected in a variety of ways. The Viola-Jones face detection model is well-known and frequently utilized. Face recognition may be achieved with several pre-trained models and approaches including Face matching, Face similarity, and Face Transformation. The techniques described above might lead to the problems outlined below.
Viola-Jones creates a sliding window and searches the candidate window for a face. It is looking for characteristics that are similar to Haar-Like features. But Viola-Jones can detect the front part of a person's face but not when the person poses in the upward, downward, or sideways direction and in few cases, it becomes difficult to detect the person's face when the face is covered with a face mask.
Face Recognition can be achieved using the various pre-trained model and other approaches, but most of the pre-trained models are trained on low-resolution images so when a person wears a facemask it becomes difficult to extract the facial features which

lead to wrongly predicting the Individual and also there is no reliable dataset for face Recognition of masked faces.

1.4 Approach

Face Recognition of a person wearing a face mask is the subject of this study. The input is collected from the camera and processed through a face detection model, which detects faces from the image and extracts a region of interest, which is then passed through a model that extracts facial features called face Descriptors, which are then used to identify the individual.

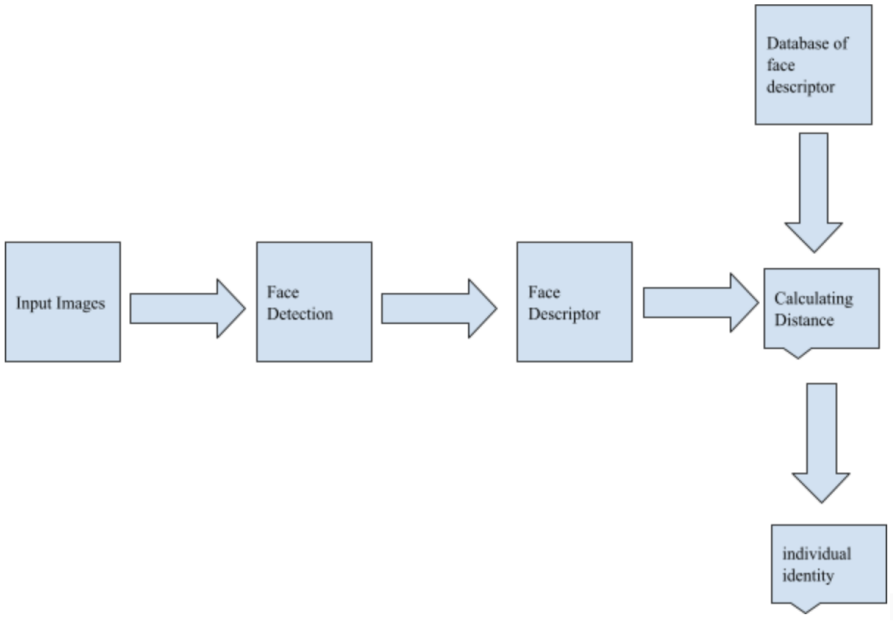

Figure 1.1. Work-Flow

1.5 Organization of Dissertation

The second chapter is a survey of the literature on several approaches to face detection and recognition. Convolutional Neural Networks and a few Machine Learning Algorithms are also discussed in the survey.

Chapter 3 explains how to construct a dataset utilizing a couple of Google's prominent platforms, such as the Google programable search engine and the Google Images Search API.

Chapter 4 explains how to face detection works, as well as the workflow and performance of the face detection model.

In Chapter 5, we learn how face recognition works, how this technique works, what types of deep learning models are utilized, and how these models function.

The conclusion of the entire study is presented in Chapter 6, as well as the research line.

1.6 Summary

Due to an increase in COVID-19 cases throughout the world, Most individuals are having trouble working. In the workplace, it is getting difficult to recognize people. The present techniques and their disadvantages were reviewed in this chapter, including the fact that face detection is difficult in some scenarios, such as posing in upward and sideways orientations or when a face is obscured. Furthermore, the majority of face recognition algorithms are trained on low-resolution pictures, which may reduce accuracy. As a result, this chapter lays forth a plan for dealing with these problems.

Chapter 2

Literature Review

The prerequisites for completing the project have been detailed in this chapter. The study was completed using a variety of research papers, books, and online material.

2.1 Face Detection

Face detection is a term that refers to computer technology that can detect the presence of people's faces in digital images. The Face detection model later used in the application like face Recognition

2.1.1 Face detection techniques

There are various methods for detecting faces, but the Viola-Jones technique is the most often used. The following is a comprehensive summary of Viola-Jones.
One of the most commonly used approaches for detecting faces is Viola-Jones. **Viola-Jones** employs Haar-Like features, Integral Images, AdaBoost, and the Cascade Classifier to produce quick and accurate results.

Haar-like Features

Viola-Jones has the Haar-like features among them the most popular features are Line Features, Edge Features, Four-sided Features, and a few others are the most often utilized. These features have black and white regions. These Haar-like features slide over the image to find the facial features. The difference between the total of black pixels and white pixels is used to determine the value of the characteristics of the face.

Below figure has Haar-Like features

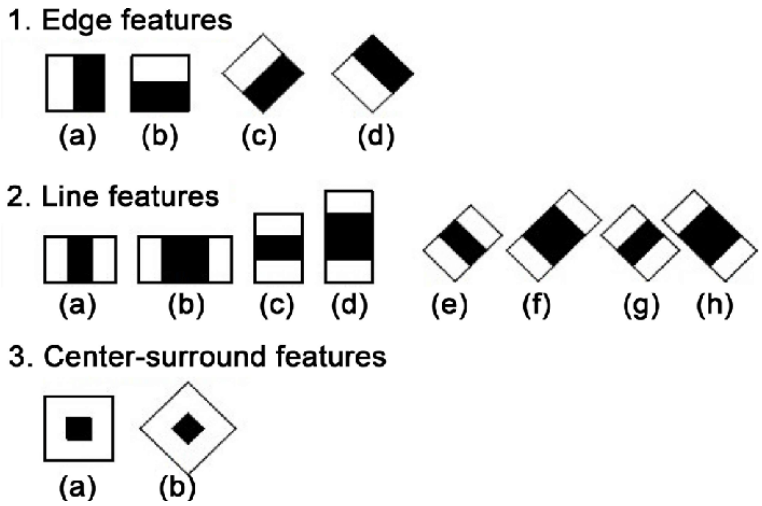

Figure 2.1. Haar-Like Features

Integral Images

Integral Images is a software program that allows you to rapidly do complex computations. The total of all pixels above and to the left, including the target pixel, is calculated in integral pictures to get each point.

1	2	3
1	2	3
1	2	3

Image

1	3	6
2	6	12
3	9	18

Integral Image

Figure 2.2. Integral Images

Adaboost

AdaBoost was the first boosting method used in binary classification. Adaboost is a machine learning technique that uses a large number of weak classifiers to produce strong classifiers. In a 24x24 candidate window, there are 1,60,000 features, which are given as input to Adaboost, which then determines the strong classifier and the weak classifier.

Cascading Classifiers

It returns the finest characteristics after executing AdaBoost. Then we put those characteristics into a cascade classifier, which quickly removes features that aren't linked to the face. We divided the cascade process into many phases, with each stage attempting to locate the face's characteristics. For instance, the first step identifies the nose, the second stage the mouth, and the third stage the eyes, and so on.

below Images provide the the work flow of Viola Jones face detection model

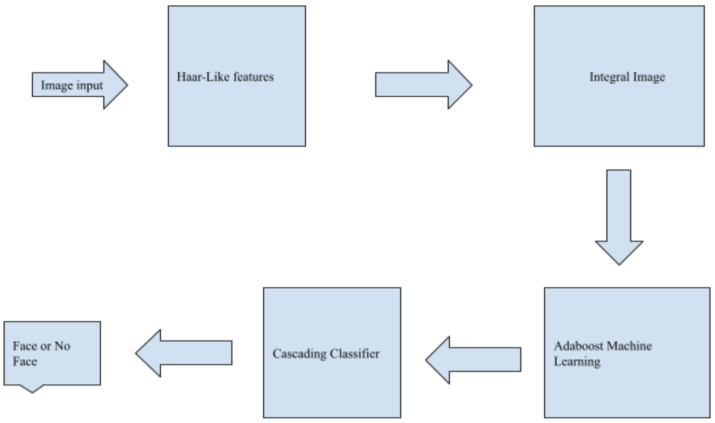

Figure 2.3. Work-Flow of Viola-Jones

2.2 Face Recognition

Face recognition is a method of verifying and identifying a person's identity through the use of a digital picture.

A face-to-database mapping of a particular face is known as **Face Identification**

A perfect match between a face and a known identity is called **Face Verification**

Facial recognition has a wide range of applications, including mobile phone unlocking, workplace attendance more recently, drones utilizing this technology to locate lost children.

2.2.1 Face recognition techniques

Face recognition techniques come in a variety of forms.

1. Holistic Matching
2. Feature Based
3. Model Based

4. Hybrid methods

Holistic matching

Holistic matching is a method that compares the similarity of two Eigenvectors, one of them is input images and the others from a database. If the Eigen vector's similarity is near to 0, then the Individual is similar, otherwise, it is dissimilar.
The process of holistic matching as follows. Eigenfaces are the characteristic feature of faces. PCA which stands for principal component analysis is a mathematical method that is used to extract features from faces. The Training dataset is supplied as input in the holistic system, and Eigen's faces are extracted from images and stored in a database. Obtain the eigenfaces from the input image and compared them with Eigenfaces in the database, and then Individual is predicted.

Feature Based

In this process, the local features of the face get extracted like the nose, eyes, and mouth. The structural classifier receives these extracted features. The feature-based main problem is restoration, which means it tries to extract face characteristics that are undetectable with substantial changes, such as when the head is positioned in a different direction.

Model Based

The model-based method aims to create a face model. The parameters are fed into the model, which then attempts to recognize the individual.

Hybrid methods

The term "hybrid techniques" refers to a method that combines holistic face recognition with facial feature extraction. The facial features, such as the contour of the chin and forehead, are retrieved from 3D pictures. It is feasible to extract the majority of face characteristics using these 3D pictures.

Detection, Position, Representation, and Matching are all performed by the 3D model. Detection is the process of taking real-time pictures of a person. The process of finding facial characteristics such as the nose, eyes, mouth, chin curve, and forehead is known as position. The process of turning facial characteristics into a numerical representation of the face is known as representation. Matching is the process of comparing an input image to a database image.

Face recognition using seven hu invariant moments

The other approach of face Recognition is using seven hu invariant moments. To know more about this first, understand what is Image moment? Images moment is nothing but the weighted average of pixel intensity. the above-mentioned seven hu invariant moments are used for shape matching. Calculate the invariant moment to translation, rotation, and scale for a given image. These extracted invariants are fed to a Deep Learning model for Training. Later an input image is given as input and extracts the seven invariant moments to predict the individual.

2.3 Convolution neural network

A Convolutional Neural Network (CNN) is a Deep Learning approach for processing images. CNN has a three-step procedure.

1. Convolution
2. Pooling
3. Fully Connected Layer

Convolution

The initial layer of a neural network is convolutional. Convolutional is a method of utilizing filters to preserve the relationship between pixels. One of the matrices considered as an image matrix, while the other is the kernel matrix. To construct a feature matrix, perform mathematical operations on both the image and kernel matrices. the most commonly used filter are edge detection, horizontal line detection, image blurring, image sharpening, etc. This procedure is carried out with a stride 1. Stride is simply a shift in the number of pixels across the input images.

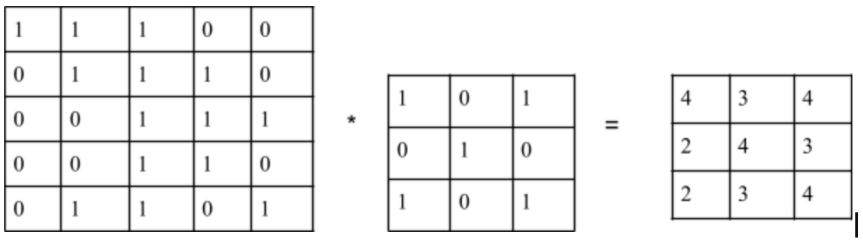

Figure 2.4. Convolution

Pooling

Pooling is the technique of downsampling a feature map from higher dimensions to lower dimensions without losing the image's feature. There are three types of pooling: maximum pooling, average pooling, and sum pooling. In max pooling, it selects the large number within the kernel. In average pooling, the average of all elements computed within the kernel. While in sum pooling, the sum of all elements computed within the kernel. During the pooling process, a stride of 1 or 2 is usually considered.

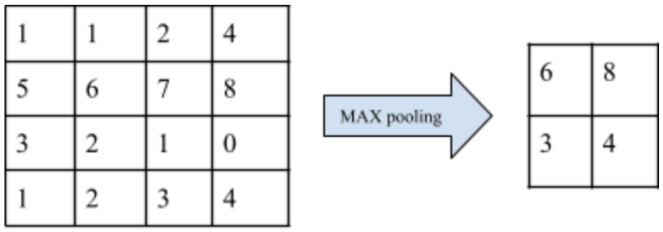

Figure 2.5. Max Pooling

fully connected layer

A fully connected layer (FC layer) is a term used to describe the last layer that is fully linked. The last layer of a convolutional neural network is the fully connected layer, which flattens all of the vectors and feeds them to the neuron.

The graphic below shows how Convolutional Neural Networks function in detail.

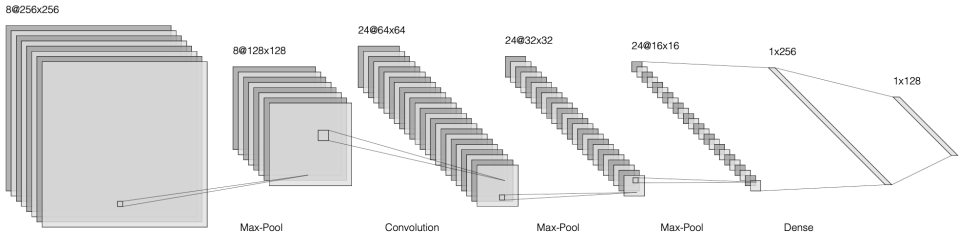

Figure 2.6. Deep Neural Network

The aforementioned Convolutional Neural Network can assist in comprehending future subjects connected to face detection and recognition.

2.4 Machine learning Models

K-Nearest Neighbouring

The K-Nearest Neighboring algorithm is a supervised machine learning method. The most common applications of KNN are classification and regression. When the data point is supplied as input, the input data point to search for the most common neighbor and predict the output. here the data point refers to an Individual.

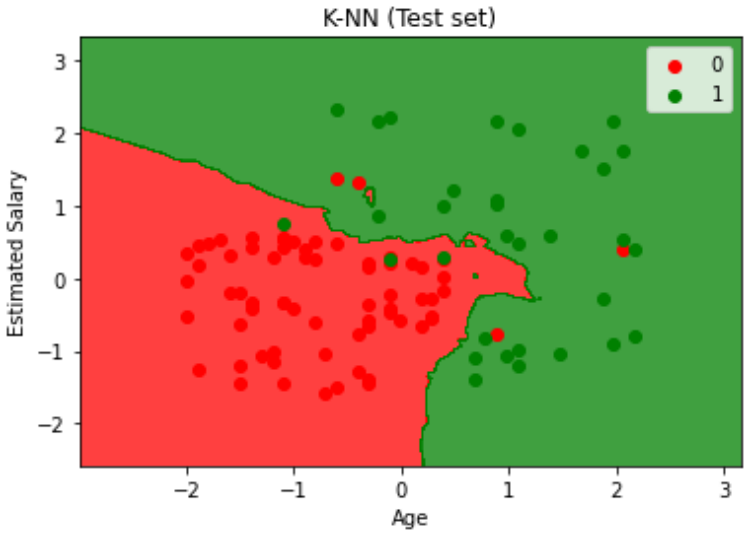

Figure 2.7. KNN Machine Learning Algorithm

2.5 Datasets

A dataset is a collection of data that is used to train the model and then use the trained model to predict the outcome. Textual data, images, and video may all be included in the dataset.

WIDERFACES

Face detection and identification datasets come in a variety of forms. WIDERFACES is the most popular face detection dataset, with 32,203 pictures and 393,703 labels. This dataset includes faces in various poses, as well as obstacles to the face such as glasses, hats, and other items.

LFW

Labeled Faces in the Wild is one of the most used datasets for face recognition. There are 13233 pictures in this collection, 5749 individuals, and 1680 people with two or more photographs.

Synthetic dataset

All of the preceding datasets were used to test unmasked faces, however, the goal of this research is to recognize masked faces. To achieve this, The mask overlayed on the LFW dataset to generate a synthetic dataset. the generation of masked laying is explained in the chapter Dataset

2.6 Summary

The essential distinction between face detection and face recognition is explained in this chapter. For face detection, Viola-Jones is used, which extracts the Haar-like

feature from the face, generates integral images, builds strong classifiers from weak classifiers, and ultimately cascades classifiers. Face recognition may be done using a variety of methods, they are Holistic Matching, Feature-Based, Model-Based, and Hybrid Approaches. when talking about datasets LFW, WIDER FACES, and synthetic datasets are the datasets used for face recognition. Finally, the CNN operation is outlined. It gave a full understanding of the functions of Convolutional, Pooling, and Fully Connected. In addition to this, it provides information on the KNN machine learning algorithm

Chapter 3

Dataset

3.1 Overview

This project's dataset should include both masked and unmasked images of a single subject.

This research uses two types of datasets:

1. Synthatic dataset
2. Real world daces dataset

Synthetic dataset

The synthetic dataset is a dataset that has been made intentionally rather than using data from the actual world. Dlib machine learning model is used to create the synthetic dataset, and it is utilized to discover the facial coordinates. Using those coordinates masked is overlaid on the faces

Real-World Masked Faces

The term "real-world dataset" refers to images that were obtained from real-world data rather than manufactured data. This dataset contains both masked and unmasked faces. A few brighter regions on the face, such as the nose area, may be seen in the real-world masked faces dataset, but no such areas can be seen when employing a synthetic mask. Face overlay isn't even suitable in some circumstances.

3.2 Prerequisites

Google Programable Search Engine

Google Programmable Search Engine is a Google platform that allows web developers to use Google Search to add particular information in online searches, filter and categorize queries, and create bespoke search engines. The author may instruct search engines to hunt for material on specific topics. Furthermore, developers have the option of embedding their search engine into any blog or website.

Google image API

Google image API The Google Custom Search API lets you access and displays search results in websites and apps using code. You may use RESTful calls to receive image search results in Atom or JSON format with this API.

3.3 Preparation of the dataset

Real-world dataset Preparation

Images from Google are obtained using the Google programmable search engine and the Google Image Search API. The program receives a list containing subject names. The application will use this list to retrieve photographs from Google Images that include both masked and unmasked faces. It will download 20 images for each category, therefore masked images will have 20 images, and unmasked will have 20 images per subject.
Below images are downloaded using Google programmable search engine and google image search API
After preprocessing The size of the dataset 4776 images and 100 subjects. Images with the mask are 1652 images and without a mask are 3124.

Figure 3.1. Without Mask

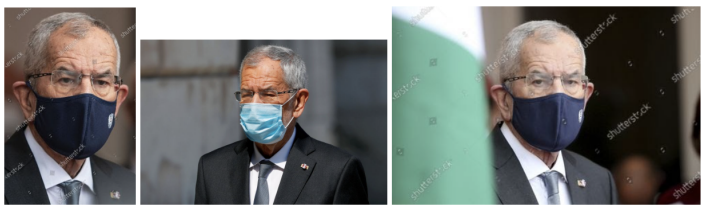

Figure 3.2. With Mask

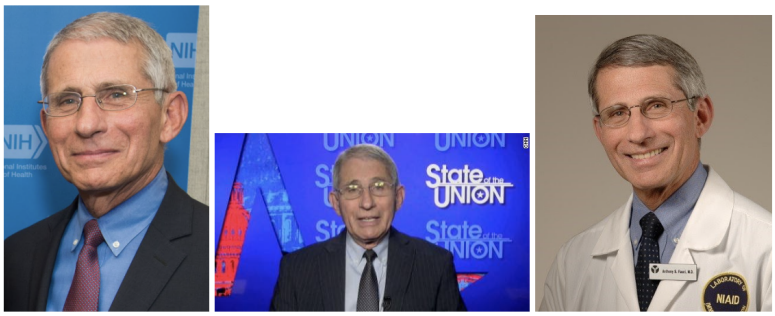

Figure 3.3. Without Mask

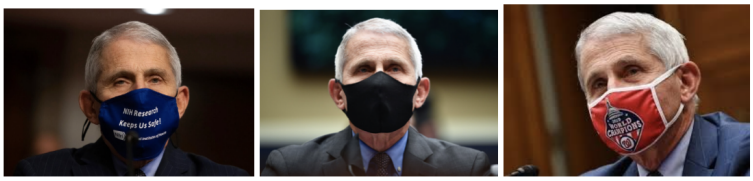

Figure 3.4. With Mask

Synthetic dataset Preparation

Dlib is a machine learning and data analysis program written in C++, according to the lib GitHub page. Dlib is capable of mapping a face using 68 coordinates. Using the

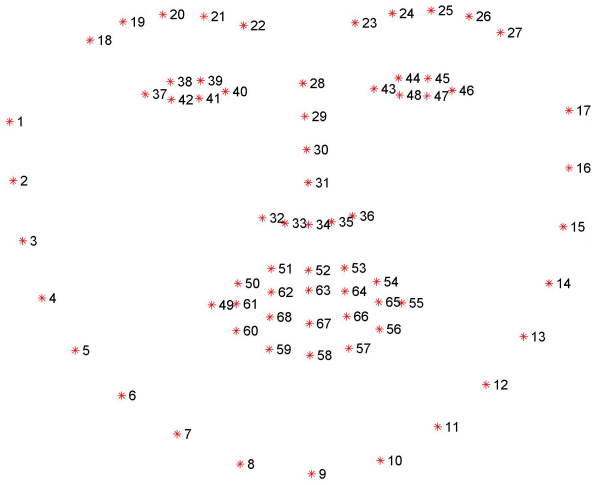

Figure 3.5. DLib landmark detection

coordinates supplied by Dlib, it possible to overlay the many types of face masks. given below the images are the inputs and outputs. The synthetic mask might pose certain issues, which will be discussed.

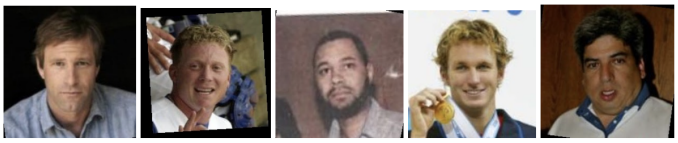

Figure 3.6. LFW Dataset

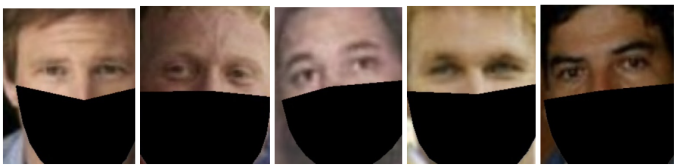

Figure 3.7. Individual with synthetic Mask

Comparison of Synthetic dataset and Real world dataset

Due to the COVID19 pandemic, it becomes difficult to collect a dataset with a masked face. so the only approach is to create a synthetic dataset in which masked are overlaid on already available standard datasets like LFW, but there are certain disadvantages like low-resolution images. It becomes difficult to extract the facial features.

But due to the improvement in technology it has become easy to collect datasets, in this project real-world dataset is downloaded from the internet using google programmable search engine and google images API with high resolution. During the process of obtaining only the masked faces, a few uncovered faces were also downloaded. This was the only issue faced when downloading the images.

3.4 Summary

The differences between synthetic and real-world datasets, as well as the advantages of the real-world dataset over the synthetic, are discussed in this chapter. This chapter describes how to use Dlib to create a synthetic dataset. I also provide information on how to download images from the internet that include both masked and unmasked face images. It explains how to use Google's programmable search engine and Google Image API to retrieve images from the internet.

Chapter 4

Multi-task Cascaded Convolutional Networks

4.1 Overview

This chapter covers the architecture, work flow, and inputs and outputs of MTCNN (Multi-task Cascaded Convolutional Networks), which is used for face detection.

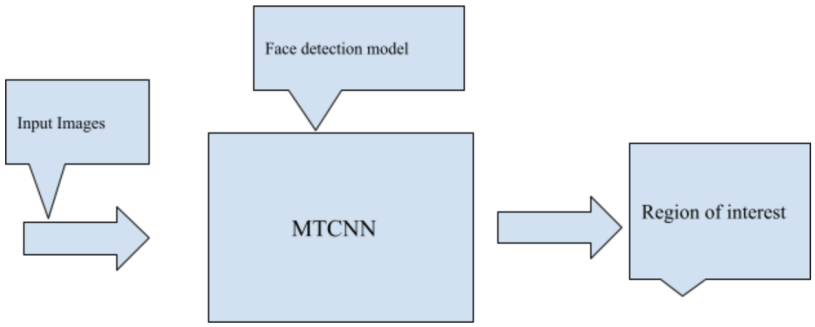

Figure 4.1. Work Flow

4.2 Prerequisites

To comprehend the principles of 'Multi-task Cascaded Convolutional Networks' you must first grasp how Convolutional neural networks operate.This subject is well-documented in the literature survey.

4.3 Face Detection

Face detection is a term that refers to computer technology that can detect the presence of people's faces in digital images.
Here in this project to overcome the drawback of Viola-jones, the MTCNN face detection model is used.MTCNN is trained on the WIDERFACES dataset.

4.3.1 Image Pyramid

An image pyramid is a series of pictures that all originate from a single source picture and are down-sampling until they reach a predetermined stopping point. The upper level is made possible by removing successive rows and columns from the lower level image. An MN picture becomes an M/2N/2 image as a result of this. As a result, the region is reduced to one-fourth of its original size. An image pyramid is the first step of MTCNN

Image pyramids are commonly used to locate the face in a photograph. for example, In a picture, there are two faces, one large face and the other has a small face. so, it is not possible by the model to detect the persons at the same level of the image As a result, when the images pyramid is built, the face with the small size is discovered at the same level as the face with the huge size can be detected on another level.

Figure 4.2. Image Pyramid

4.3.2 P-Net, R-Net and O-Net

The MTCNN architecture is seen in the figure below.

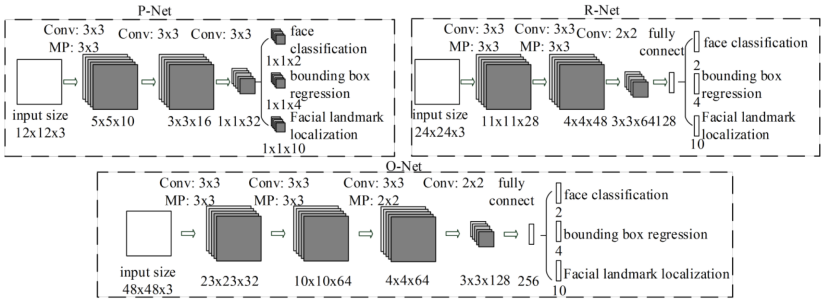

Figure 4.3. MTCNN Architecture

P-Net

The proposal network, often known as the P-Net, has a 12x12 kernel with a stride of 2. P-Net receives an image pyramid as input. If a face is discovered, the bound box's coordinates are returned. The bounding box coordinates are rescaled to match the original picture, and NMS is used to eliminate any extra bounding boxes that P-Net predicts. P-Net may also anticipate face landmarks such as two eyes, a nose, and two mouth endpoints.

R-Net

R-Net is also called a refined network. R-Net receives all of P-Net's anticipated bounding boxes. The bounding box coordinates of the P-Net network are resized to a 24x24 kernel size. R-Net rejects the P-Net predicted false bounding box and uses NMS to remove the overlapping bounding box. Face landmarks such as two eyes, a nose, and two mouth endpoints are also predicted by R-Net.

O-Net

O-Net is also called an output network. O-Net receives all of R-Net's anticipated bounding boxes. The bounding box coordinates of the R-Net are adjusted to a kernel size of 48x48. Further, O-Net rejects the R-Net predicted false bounding box and uses NMS to remove the overlapping bounding box. O-net generates a single bounding box per face, as well as five facial markers such as two eyes, a nose, and two mouth ends.

4.3.3 Non-Maximum Supression

Non-maximum suppression is another name for NMS. Generally, NMS is used to eliminate the overlapping bounding box. NMS removes the bounding boxes with the lowest confidence score and creates a new list for the remaining bounding boxes. After that, sort them in decreasing order. IOU on the bounding box is performed. If the overlapping score, IOU (intersection over union), is more than 0.5, the bounding box with the lower confidence score is removed.

Figure 4.4. Non-Maximum Supression

4.3.4 Experiments

This experiment uses a dataset that includes both masked and unmasked faces. The dataset has 4776 pictures and 100 subjects in size. There are 1652 pictures with masks and 3124 images without masks.

Figure 4.5. Input Images

Figure 4.6. figure 21:Output Images

The face detection is performed on both masked and unmasked faces and the accuracy score of the masked face is 0.94 where are accuracy score of the unmasked face is 0.98

Comparison of viola jones and MTCNN

As previously stated, viola jones is unable to identify faces when they are posed in downward, upward, or sideways orientations, or when they are covered by any obstacle. MTCNN, which was trained on the WIDERFACES dataset, tackles these issues. such as the mask-covered face, eyeglasses, hat, and other items are included in the dataset.

4.4 Summary

Face detection using the MTCNN model is the main topic of this chapter. It explains MTCNN's design, which includes three internal networks: P-Net, R-Net, and O-Net. We construct an image pyramid before passing the images to the model, which aids the MTCNN model in detecting large faces in an image, and then it is passed to P-Net. P-Net job is to detect faces, landmarks, and construct a bounding box. After that, the outputs are sent to the R-Net. The R-Net job is to improve the inputs before passing them on to the O-Net. The O-Net is the last step of this MTCNN model, and it supplies the Bounding Box as well as the facial landmarks. during this process, NMS is used to

eliminate bounding boxes that overlap. This method provided a better result in Different scenarios.

Chapter 5

Face Recognition

5.1 Overview

Face recognition is a method of identifying or authenticating a person's identity by looking at their face in a digital photograph. A person's facial characteristics are compared to see how similar they are. The similarity metric is utilized to recognize a person in this project.

5.2 Prerequisites

To understand the full structure of face recognition, you need to be familiar with a few deep learning models such as ResNet50, Vgg19, and SqueezeNet. The basic structure of face recognition algorithms is represented by these deep learning models.

5.2.1 Cosine Distance

The cosine distance is a metric that compares the distance between two vectors. These vectors can be facial descriptors. To understand cosine distance We must understand what cosine similarity is?

Mathematical operation of cosine similarity is provided in the equation. Generally, cosine similarity is the angle between two vectors.

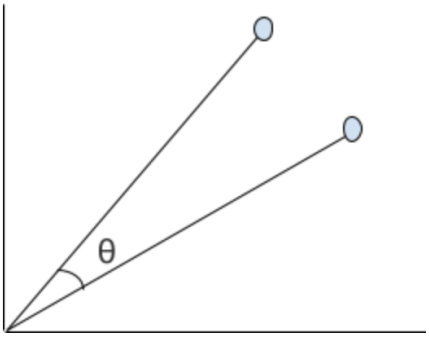

Figure 5.1. Cosine similarity

$$Cosine\_Similarity = [A.B]/[||A|| * ||B||]$$

$$Cosine\_Distance = 1 - Cosine\_similarity$$

5.2.2 ResNet50

ResNet-50 is a 50-layer neural network. The residual neural network is another name for ResNet50. ResNet50 is a neural network that has been trained on over a million photos from the ImageNet collection. Around 1000 topics, such as pen, pencil, puppy, and so on, maybe classified using this methodology. The input picture is 224x224 pixels in size.

ResNet50
50 Layer
7x7,64,stride 2
3x3,maxpool,stride 2
1x1,64 3x3,64 X3 1x1,256
1x1,128 3x3,128 X4 1x1,512
1x1,256 3x3,256 X6 1x1,1024
1x1,512 3x3,512 X3 1x1,2048
Average pooling,1000-d Fully Connected,softmax
$3.8x10^9$

Figure 5.2. ResNet50

5.2.3 VGG19

Visual Geometry Group from Oxford is also known as Vgg19. The Vgg19 is a 19-layer convolutional neural network. Vgg19 has been trained on over a million photos from the ImageNet collection. Around 1000 topics, such as pen, pencil, puppy, and so on, maybe classified using this methodology. The input picture is 224x224 pixels in size.
The information below gives you a full picture of VGG19's architecture.

VGG19
19-Layers
Input (224x224 RGB image)
Conv3-64 Conv3-64
Maxpool
Conv3-128 Conv3-128
Maxpool
Conv3-256 Conv3-256 Conv3-256 Conv3-256
Maxpool
Conv3-512 Conv3-512 Conv3-512 Conv3-512
Maxpool
Conv3-512 Conv3-512 Conv3-512 Conv3-512
Maxpool
FC-4096
FC-4096
FC-1000
Softmax

Figure 5.3. VGG19

5.2.4 SqueezeNet

SqueezeNet is a 50-layer convolutional neural network. SqueezeNet was trained using the ImageNet database, which contains over 1 million pictures. Around 1000 topics, such as pen, pencil, puppy, and so on, maybe classified using this methodology. The input picture is 224x224 pixels in size.

SE-ResNet50
50 Layer
7x7,64,stride 2
3x3,maxpool,stride 2
1x1,64 3x3,64 X3 1x1,256 fc,[16,256]
1x1,128 3x3,128 X4 1x1,512 fc,[32,512]
1x1,256 3x3,256 X6 1x1,1024 fc,[64,1024]
1x1,512 3x3,512 X3 1x1,2048 fc,[128,2048]
Average pooling,1000-d Fully Connected,softmax

Figure 5.4. SqueezeNet

5.3 VGGFACE2

VGGFACE2 has been utilized in this project. Oxford University has created VGGFACE2, a facial recognition model. The model was trained using 3.31 million photos from 9131 people, with each subject receiving an average of 362.6 photos. Images with

a variety of poses, ages, lighting, ethnicity, and professions are downloaded for Google image search. ResNet50, Vgg19, and SqueezeNet are the three networks in this model.

The ROI (region of interest) is sent to VGGFACE2 once MTCNN detects it. VGGFACE2 is a program that extracts facial characteristics known as face descriptors. The length of the face descriptor is 2048.

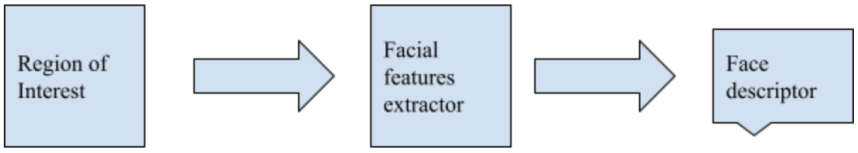

Figure 5.5. VGGFACE2 workflow

The similarity measure is used to determine the degree of similarity between the input face descriptor and the database face descriptor.

In this project, the cosine distance is employed to determine how similar the faces are. Faces are similar if the similarity is near zero. If it's near to 1, the faces aren't alike.

KNN is a Machine Learning algorithm that is used to determine Individual identification.

5.4 Experiment

To forecast the individual, one way is to use threshold values on cosine distance, while the other is to use the K-NN machine learning algorithm.

After numerous tests, it was discovered that a cosine distance of 0.7-0.8 works well, and it was also suggested in the FaceNet study. However, for every input image a lot of computation to be on the entire database, it takes a lot of work and time. KNN may be used to speed things up while also improving accuracy.

So, with accuracy in mind, we're considering the KNN machine learning method for the prediction of individuals.

below images are the outputs while performing the experiment

Figure 5.6. Face Recognition of image 1

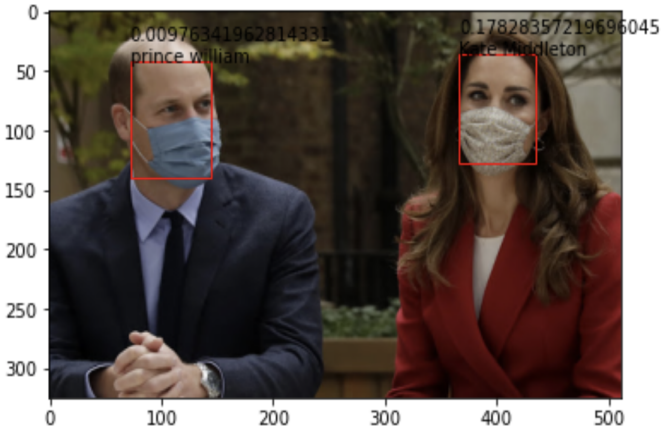

Figure 5.7. Face Recognition of image 2

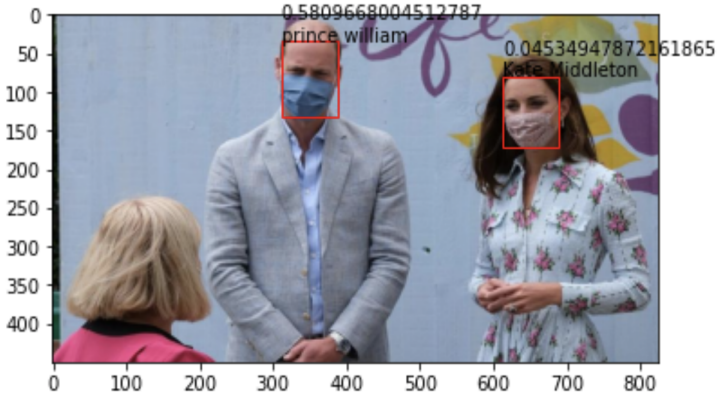

Figure 5.8. Face Recognition of image 3

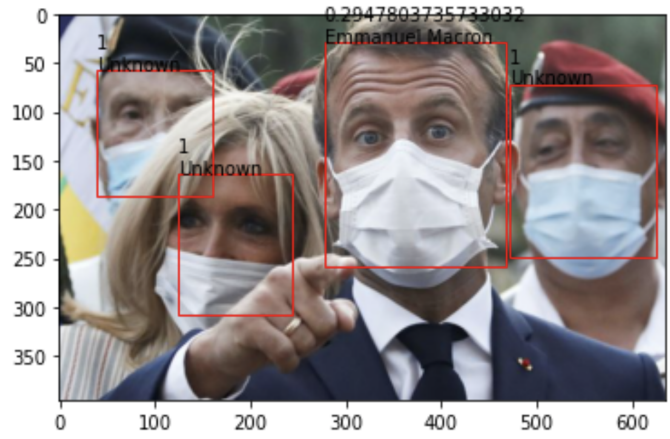

Figure 5.9. Face Recognition of image 4

Using KNN, a person is predicted based on the data points surrounding it. Is it feasible to anticipate the K values for a given dataset? This may be accomplished by carrying out several experiments.

With the dataset, we created many sets. The first set contains 100 subjects, each with 5 images, the second set contains 100 subjects, each with 10 images, the third set contains 100 subjects, each with 15 images, the fourth set contains 100 subjects, each with 20 images, the fifth set contains 100 subjects, each with 25 images, and the sixth set contains 100 subjects, each with 30 images.

The minimum and maximum number of images per individual are 20 and 30, respectively. In all trials, assume that the test dataset set contains 56 masked images.

Trained copies per subject	K	Accuracy Score
5	9	0.82
10	9-13-17-19	0.78
15	15-17-19-21-23-25-27-29-31	0.76
20	21-23-25-27-29	0.78
25	21-23-25-27-29-31-33-35	0.78
30	27-29	0.80

Table 5.1. Masked images as Input

when the model was trained on 100 different subjects, each with 5 images At K equal to 9, the greatest accuracy score attained is 0.82.

when the model was trained on 100 different subjects, each with 10 images At K equal to 9-13-17-19, the greatest accuracy score attained is 0.78.

when the model was trained on 100 different subjects, each with 15 images At K equal to 15-17-19-21-23-25-27-29-31, the greatest accuracy score attained is 0.76.

when the model was trained on 100 different subjects, each with 20 images At K equal to 21-23-25-27-29, the greatest accuracy score attained is 0.78.

when the model was trained on 100 different subjects, each with 25 images At K equal to 21-23-25-27-29-31-33-35, the greatest accuracy score attained is 0.78.

when the model was trained on 100 different subjects, each with 30 images At K equal to 27-29, the greatest accuracy score attained is 0.80.

Trained copies per subject	K	Accuracy Score
5	1-3-5-7-9-11	0.96
10	7-9-11-13-15-17-19-21-23-25-27-29	0.98
15	3-5-7-9-11-13-15-17-19-21-25-27-29-31-35	1
20	1-3-5-7-9-11-13-15-17-19-21-25-27-29-31-35-37	1
25	1-3-5-7-9-11-13-15-17-19-21-25-27-29-31-35-37-39-41-43-45-47	1
30	1-3-5-7-9-11-13-15-17-19-21-25-27-29-31-35-37-39-41-45-47-49-51-53-55-57-59-61	1

Table 5.2. UnMasked images as Input

when the model was trained on 100 different subjects, each with 5 images At K equal to 1-3-5-7-9-11, the greatest accuracy score attained is 0.96.

when the model was trained on 100 different subjects, each with 10 images At K equal to 7-9-11-13-15-17-19-21-23-25-27-29, the greatest accuracy score attained is 0.98.

when the model was trained on 100 different subjects, each with 15 images At K equal to 3-5-7-9-11-13-15-17-19-21- 25-27-29-31-35, the greatest accuracy score attained is 1.

when the model was trained on 100 different subjects, each with 20 images At K equal to 1-3-5-7-9-11-13-15-17-19-21-25-27-29-31-35-37, the greatest accuracy score attained is 1.

when the model was trained on 100 different subjects, each with 25 images At K equal to 1-3-5-7-9-11-13-15-17-19-21-25-27-29-31-35-37-39-41- 43-45-47, the greatest accuracy score attained is 1.

when the model was trained on 100 different subjects, each with 30 images At K equal to 1-3-5-7-9-11-13-15-17-19-21-25-27-29-31-35-37-39-41-45-47-49-51-53-55-57-59-61, the greatest accuracy score attained is 1.

Observation

The average accuracy score for masked faces is 0.78. Unmasked faces, on the other hand, have an average accuracy score of 0.99.

As per the experiment K value should be the average number of images per subject.

The other approach of Face Recognition was tested using seven invariant moments, and the accuracy score was below 0.4.

I tested my hypothesis using the LFW synthetic dataset. It achieved the maximum accuracy of 0.44 at K=3. I've also discussed the drawbacks of synthetic datasets in the dataset chapter.

We began investigating why there was a drop in inaccuracy when we saw it. We gathered the results of the above experiment's mistake. The input data point is surrounded by the datapoints of other subjects when we applied the KNN method. The observation made from above is that there is a possibility of a masked face matching with another subject.

5.5 Summary

This chapter explains what face recognition is and how it works. VGGFACE2 is a well-known face recognition model that is utilized in this project. Vggface2 is a pre-trained Deep neural network model for face recognition that has been trained on 3.31million pictures with over 9131 subjects. VGG19, ResNet, and SqueezeNet are the three types of architecture utilized in VGGFACE2.VGGAFCE2 is used to extract the face descriptor from the region of interest. These facial descriptors are then utilized to identify individuals using the cosine Distance or KNN Algorithm. finally, the Analysis of Face Recognition is also provided which helps in further research.

Chapter 6

Conclusion and Future Scope

6.1 Conclusion

The primary focus of this project is face detection and recognition of a person wearing a face mask. Due to a rise in COVID19, it has become difficult to recognize the individual. In recent years improvements in Image processing Technology, it became possible to tackle the problem in the real world.
As of now, there is no reliable dataset for the Recognition of masked faces. So the only approach is to create a synthetic dataset. The synthetic dataset is created by using the Dlib machine learning tool which plots the coordinates on the face and these coordinates help in overlay the face mask. Due to the increase in technology, it is possible to create a real-world dataset from the Internet using google programmable search engine and google image API.
For face detection, the MTCNN model is used which internally has three networks they are Proposal network, Refine Network, and the Output Network which is trained on the WIDERFACES dataset.MTCNN takes the input as an images pyramid and draws a bounding box and also provides the facial landmarks. for face recognition, a well-known face recognition model called VGGAFCE2 is used which is trained over 3.31 million images over 9131 subjects.VGGFACE2 takes the input as the region of interest and provides a face descriptor as output that has a length of 2048. these face descriptors are later used for face Recognition. the face descriptor is passed through a metric called cosine distance which helps in Recognition individuals. When the cosine distance close to 0 then the person is similar or else dissimilar. This approach provides a better result in Recognizing the individual with facemask.

6.2 Future Scope

1. To create a large-scale dataset for masked face Recognition with higher resolution.

2. To create a lighter version of this project so it can run on lighter devices such as mobile and raspberry pi.

Refrences

[1] Kaipeng Zhang, Zhanpeng Zhang, Zhifeng Li, Senior Member, IEEE, and Yu Qiao, Senior Member, IEEE,"Joint Face Detection and Alignment using Multi-task Cascaded Convolutional Networks",2016,Cite as:arXiv:1604.02878 [cs.CV].

[2] Shuo Yang, Ping Luo, Chen Change Loy, Xiaoou Tang,"WIDER FACE: A Face Detection Benchmark",2015,Cite as: arXiv:1511.06523 [cs.CV].

[3] Aqeel Anwar, Arijit Raychowdhury,"Masked Face Recognition for Secure Authentication",2020,Cite as:arxiv:2008.11104

[4] Ziyuan Yang,Jing Li, Weidong Min and Qi Wang,"Real-Time Pre Identification and Cascaded Detection for Tiny Faces",2019.

[5] Qiong Cao, Li Shen, Weidi Xie, Omkar M. Parkhi,Andrew Zisserman,"VGGFace2: A dataset for recognising faces across pose and age",[v1]- 2017,[v2]-2018,Cite as:arXiv:1710.08092 [cs.CV].

[6] G.B.Huang, M.Ramesh, T.Berg, and E.Learned-Miller."Labeled faces in the wild: A database for studying face recognition in unconstrained environments. Technical report" 2007.

[7] Jie Hu,Li Shen,Samuel Albanie,Gang Sun,Enhua Wu."Squeeze-and-Excitation Networks",2017.

[8] Divyarajsinh N. Parmar, Brijesh B. Mehta,"Face Recognition Methods Applications",Cite as:arXiv:1403.0485. [cs.CV].

[9] Florian Schroff, Dmitry Kalenichenko, James Philbin, "FaceNet: A Unified Embedding for Face Recognition and Clustering",2015,Cite as:arXiv:1503.03832 [cs.CV]

[10] A. Nabatchian, E. Abdel-Raheem, M. Ahmadi,,"Human Face Recognition Using Different Moment Invariants: A Comparative Study",2008 Congress on Image and Signal Processing

Web Resources:

[11] https://opencv.org/

[12] https://medium.com/mlearning-ai/facial-mask-overlay-with-opencv-dlib-4d948964cc4d

[13] https://www.analyticsvidhya.com/blog/2021/04/simple-understanding-and-implementation-of-knn-algorithm/

[14] https://machinelearningmastery.com

[15] https://docs.opencv.org/3.4/d4/d1f/tutorial_pyramids.html

[16] https://www.pyimagesearch.com

[17] https://github.com/rcmalli/keras-vggface

[18] https://www.tensorflow.org

[19] https://learnopencv.com/shape-matching-using-hu-moments-c-python/

[20] http://alexlenail.me/NN-SVG/LeNet.html