KLE Society's
KLE Technological University, Hubballi.

A Minor Project Report

on

Masked Face Recognition

submitted in partial fulfillment of the requirement for the degree of

Bachelor of Engineering

in

Computer Science and Engineering

Submitted by

Pooja Doddannavar	01FE18BCS141
Prasad V Patil	01FE18BCS148
Prasann Kshirasagar	01FE18BCS149
Prashant Kumar	01FE18BCS150
Abhay Ambekar	01FE18BCS006

Under the guidance of
Dr. Sujatha C

SCHOOL OF COMPUTER SCIENCE AND ENGINEERING

Hubballi – 580 031

2020 -21

KLE Society's
KLE Technological University, Hubballi.

2020 - 2021

SCHOOL OF COMPUTER SCIENCE AND ENGINEERING

CERTIFICATE

This is to certify that Minor Project titled masked face recognition is a bonafied work carried out by the student team comprising of Pooja Doddannavar (01FE18BCS141), Prasad V Patil (01FE18BCS148), Abhay Ambekar (01FE18BCS006), Prashant Kumar (01FE18BCS150), Prasann Kshirasagar (01FE18BCS149), for partial fulfillment of completion of sixth semester B.E. in Computer Science and Engineering during the academic year 2020-21.

Guide	Head, SoCSE
Dr. Sujatha C	Dr. Meena S. M

Viva -Voce:

Name of the Examiners	Signature with date
1.	
2.	

Acknowledgement

We would like to thank our faculty and management for their professional guidance towards the completion of the project work. We take this opportunity to thank Dr. Ashok Shettar, Vice-Chancellor, Dr. N.H Ayachit, Registrar, and Dr. P.G Tewari, Dean Academics, KLE Technological University, Hubballi, for their vision and support.

We also take this opportunity to thank Dr. Meena S. M, Professor and Head, SoCSE for having provided us direction and facilitated for enhancement of skills and academic growth.

We thank our guide Dr. Sujatha C, Associate Professor, SoCSE for the constant guidance during interaction and reviews.

We extend our acknowledgement to the reviewers for critical suggestions and inputs. We also thank Project Co-ordinator Dr. Sujatha C. and faculty in-charges for their support during the course of completion.

We express gratitude to our beloved parents for constant encouragement and support.

Pooja Doddannavar - 01FE18BCS006
Prasad Patil - 01FE18BCS148
Prasann Kshirasagar - 01FE18BCS149
Prashant Kumar - 01FE18BCS150
Abhay Ambekar - 01FE18BCS006

ABSTRACT

Face recognition is widely used across several biometric applications. In recent times, where Covid-19 has impacted manufacturing, travel, tourism, hospitality, and crippling the global economy, wearing a mask is mandatory in several institutions and public places for its widespread and safety of an individual. Under this scenario, person recognition for security purposes is been challenging for the current face recognition systems in which most of the facial features are covered. Our work relies on computer vision and deep learning models which intend to make an impact and solve the real-world problem of safety measures at some significant level.

In our work we propose a framework for the Masked Face Recognition using InceptionV3 and FaceNet architectures, which can be easily integrated into various embedded devices with limited computational capacity. We aim to detect face mask and recognize the person in images as well as in real-time videos. We demonstrate the results with an overall accuracy of 88% for masked face recognition within the defined scope.

Our work can be extended to authenticate people at public places such as airports, offices, hospitals, schools, etc., to ensure that the safety standards are maintained and people abide by the rules and regulations.

CONTENTS

LIST OF TABLES

LIST OF FIGURES

Chapter 1

INTRODUCTION

In this new era where we are experiencing a pandemic, people are advised to wear masks to protect themselves and to reduce the spread of the coronavirus. In this case, masked face recognition is a very difficult task since certain face parts are hidden. A primary focus is to bring up a rapid and efficient solution to address this problem. Since the majority of the systems are ineffective for this new constraint of face recognition with mask technique have been less effective and inaccurate in recognizing masked face recognition which is essential for face authorization or authentication for wide-region tracking for security and surveillance, facilitating secure payments, contact tracing of Covid-19 suspects, etc.

In our work, we propose a framework for an efficient masked face recognition, initially a face mask is detected and then the facial feature-based extraction is performed for identifying the masked person using his/her face. Using this framework, we can detect people wearing or not wearing masks in public places and recognize the person and authenticate accurately, this would be helpful in increasing the safety measures.

1.1 Motivation

Masked Face recognition has recently been a blooming attention for this new era where we experiencing a pandemic and people all around are advised to wear masks, which has caused a major security issue in general public in case of authorization. Which has more demand for useful security systems, masked face recognition system is limited when the face is covered and cannot fetch the facial features required.

To build these applications, reliable and robust automated masked facial recognition techniques and methods are required. However, these setup methods and techniques are not available easily or only available in high complex, expensive setups.

Masked face recognition is to help the security system of the public and authorize more accurately in order to prevent the unauthorized access during this pandemic crisis and also for further such scenarios applicable. It would be helpful to increase our safety and this would potential be contribution in fighting the pandemic and towards the public and welfare of society.

1.2 Literature Survey

Jiang, Mingjie [1] discussed about RetinaFaceMask and ResNet models. They have also included MobileNet as a backbone for comparison and to reduce computation and model size in deployment scenarios with limited computing resources. For extracting high-level semantic information, FPN is applied as neck in RetinaFaceMask model. Then the information is fused into previous layers feature map by adding operation with a coefficient. Here in their model, classifiers, predictors, estimators are considered as head, which achieve the final objectives of the network.

Cahyono, Ferry [2] proposed an appropriate method to be applied in a presence system using faces by comparing two deep learning architectural models, they are FaceNet and OpenFace. FaceNet model was introduced by Google researchers which has good accuracy in face recognition. While OpenFace is a model that was developed from FaceNet and is trained with smaller datasets and still attains an accuracy almost equal to FaceNet. The preprocessing of face follows the flow of detect, crop, and resize the face using MTCNN model. Then facial features were extracted into 128 dimensional vector embedding using the FaceNet along with OpenFace. To obtain better accuracy, SVM is used for facial feature classification. The accuracy obtained from FaceNet model was higher compared to OpenFace model.

Wang, Zhongyuan [3] proposed a feasible approach for detecting the facial regions. The occluded face detection problem has been approached using Multi-Task Cascaded Convolutional Neural Network (MTCNN). Then facial features were extracted using the Google FaceNet embedding model and finally the classification task was performed using Support Vector Machine (SVM). Experiments signify that the mentioned approach gives a remarkable performance on masked face recognition. Besides, its performance has been also evaluated within excessive facial masks and attractive outcomes were found.

Ge, Shiming [4] discussed about LLE-CNNs, which consists of three modules. In the proposed module, extraction and characterization of face candidates is performed by cascading two CNN's for proposal generation and feature extraction, respectively. In proposal generation, P-Net was adopted to build first four layers of CNN's, those are three convolutional layers and a softmax layer and VGGFace model is been applied on those proposals. In the end, for transferring the output into similarity based vectors, the embedding module is applied. Finally, the verification module picks up the output having the minimum value among all the similarity vectors and hence identifies the person. The model works well for face detection, but it lags when it comes to detecting faces with occlusions.

Aswal [5] proposed masked face detection and identification using two approaches, namely single-step process using pretrained YOLO-Face/trained YOLOv3 model on a dataset of known people and two-step process, using RetinaFace for localizing the masked faces and VGGFace2 for generating facial feature vectors. The dataset discussed in paper[5], consists of real-world video examples comprising of 7 individuals with various orientations, illuminations, and occlusions. Experimental results show that RetinaFace and VGGFace2 achieve state-of-the-art results of 92.7% on overall performance, 98.1% face detection, and 94.5% face verification accuracy respectively on their custom dataset.

1.3 Technical Challenges

- The model offers low speed for face recognition as well as accuracy (92.7 %).
- The model works well for face detection , but it lags when it comes to detecting faces with occlusions.

1.4 Problem Statement

We propose to develop a solution for recognising the person with mask in static/real-time scenarios. We detect the face of a person with mask using InceptionV3 model and recognize the person using FaceNet model.

1.5 Applications

This project can be used at schools, offices – manufacturers, retail, other SMEs and corporate giants, hospitals/healthcare organizations, airports and railway stations, sports venues and many more public places to ensure that the safety standards are maintained and people are abiding by the rules and regulations and to authenticate only is a person is wearing mask and is recognised correctly.

1.6 Objectives and Scope of the project

1.6.1 Objectives

- To generate a non-masked and masked face dataset for face recognition.
- To build a learning model for face mask detection.
- To build a learning model for masked face recognition.

1.6.2 Scope

- The orientation of the face during the acquisition must be between 45-60 degrees for accurate recognition of the person.
- Image input must be in form of png, jpg, jpeg formats.
- Video input must be in form of mkv, avi, mp4 video formats.

Chapter 2

REQUIREMENT ANALYSIS

Requirements analysis focuses on the tasks that determine the requirements or conditions to satisfy the new or altered product or project, taking account of the possibly conflicting requirements of the various participants, analyzing, documenting, validating, and managing software or system requirements.

2.1 Functional Requirements

Functional requirements define the fundamental system behaviour for the Masked Face Recognition System. It defines a function of the system or its component, where a function is described as a specification of behaviour between inputs and outputs. The plan for implementing functional requirements is detailed within the system design.

- The system shall be able to have an unbiased, over 2500+ images with and without mask dataset which unique images in train and test phases.
- The system shall be able to detect faces in images and video streams and extract each face's Region of Interest (ROI).
- The system shall be able to detect the mask on the detected face.
- The system shall be able to recognize the person from the extracted region correctly within the defined scope.
- The system shall be able to assign a label as "with_mask" along with person name or only "without_mask" label.

2.2 Non Functional Requirements

A non-functional requirement (NFR) is a requirement that specifies criteria that can be used to judge the operation of a system, rather than specific behaviours. The plan for implementing non-functional requirements is detailed in the system architecture.

- The system should be portable and can be applied to embedded devices with limited computational capacity.
- The system should be user friendly and self-descriptive for maintenance purposes.
- The system response time should be less than 5 seconds per input image.
- The background should not be too bright or too dark while detecting the face and mask.

2.3 Hardware Requirements

- Processor: 1.6 GHz Intel Core i5/Pentium IV 2.4 GHz.
- RAM: at least 2 GB RAM..
- Speed: 500 MHz.
- Hard Disk: 80 GB minimum.
- Accessories: A high quality wireless/webcam camera, LCD/LED .
- Monitor, Keyboard, Mouse.

2.4 Software Requirements

- Operating System:
  - Windows 8 or later
  - Mac OS 10.13.6 or later (preferable)
  - Ubuntu 16.04 or later (64-bit) (preferable)
  - PyCharm/ VSCode editor/ TensorFlow GPU (optional)
- Programming Language:
  - Python (3.7.6)

- Open Libraries:
  - openCV (Intel's Computer Vision Open-Source Library) (4.2.0)
  - tensorFlow (1.14.0)
  - keras (2.3.1) (TensorFlow backend)
  - sklearn (0.22.1)
  - imutils (0.5.3)
  - numpy (1.18.2)
  - matplotlib (3.1.3)
  - argparse (1.1)

Chapter 3

SYSTEM DESIGN

In this chapter we discuss the framework for masked face detection using InceptionV3 model and masked face recognition using FaceNet model. The proposed high level design and its workflow is explained along with dataset generation and dataset pre-processing.

3.1 Framework for the masked face recognition

The architecture of the proposed model is mainly based on 2 modules. Those are Face detection along with mask detection and face recognition. This is attained using Deep CNN-based models namely InceptionV3 and FaceNet. In the proposed system MTCNN (Multi-Task Cascaded Convolutional Neural Network) is used for face detection along with InceptionV3 for mask detection. The inceptionV3 was used for mask detection as it is good for training small-sized datasets with higher accuracy. The major module in the proposed architecture is the Face Recognition module. Face recognition can be done using various models namely FaceNet, VGG, and Deep Face models. In this system, FaceNet was used in the Face Recognition module.

Advantages of FaceNet :

1. FaceNet can identify similar faces having different facial expressions and angles(facial expressions).

2. FaceNet can handle various lighting conditions, occlusions and transformations efficiently.

3. This model can identify the same faces despite the person have markings on their face.

4. The model can recognize the individual correctly when half of the face was cut off.

5. Action sequences where the person face is highly different can still be matched in the FaceNet model.

The InceptionV3 and FaceNet Models were then combined on the constraint that only the masked faces must be recognized.

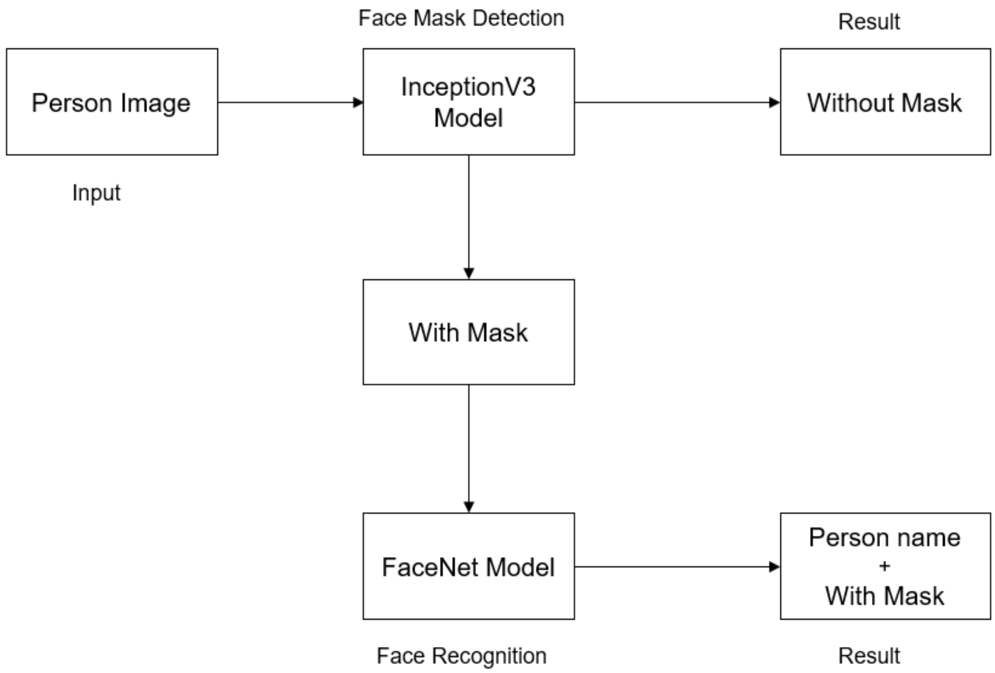


Figure 3.1: High level design for masked face recognition

3.2 Workflow of the proposed system

The user will give input by uploading the image on the website/web interface. If the input image is properly loaded then general pre-processing will be done like resizing and re-scaling after that, if the face of the person is present in the input image then the image passes through the MTCNN model where the model detects the face of the person. Then the image is sent to the InceptionV3 model for face mask detection, if a mask is present in the right place on the face of the person then the InceptionV3 model passes this image to the FaceNet model where the person face recognition is done based on the comparison between vector embeddings taken from this input image with vector embeddings present within the database. The workflow of the proposed system is shown in Figure 3.2.

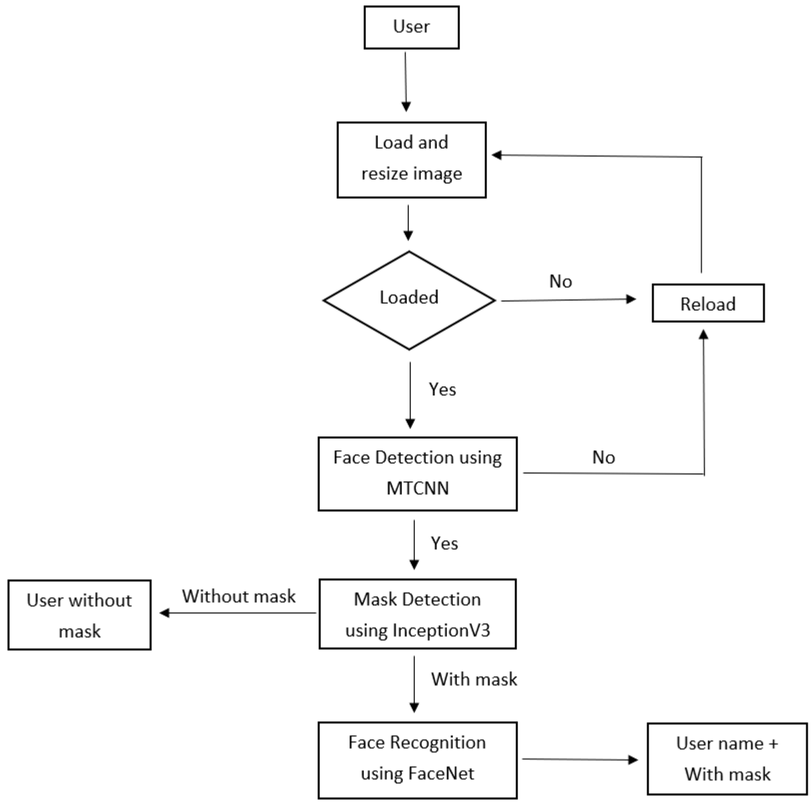


Figure 3.2: Flowchart for masked face recognition

3.3 Dataset Generation and Pre-processing

Image dataset generation is a process of collecting images of various individuals with different angles and postures. Dataset generation is carried out to train a built model which when implemented gives the needed result. The dataset available on the web are not so relevant to train the face recognition model in the best way, so we have generated our own dataset according to our model needs, which is further used for model training and testing. Since the data was not sufficient enough to train the model we have used data augmentation as a pre-processing technique to meet the requirements.

3.3.1 Dataset Generation

The dataset is generated consists of 2520 images both with and without mask.

The dataset consists of 2520 images of 42 people -

1. with mask : 1260 images

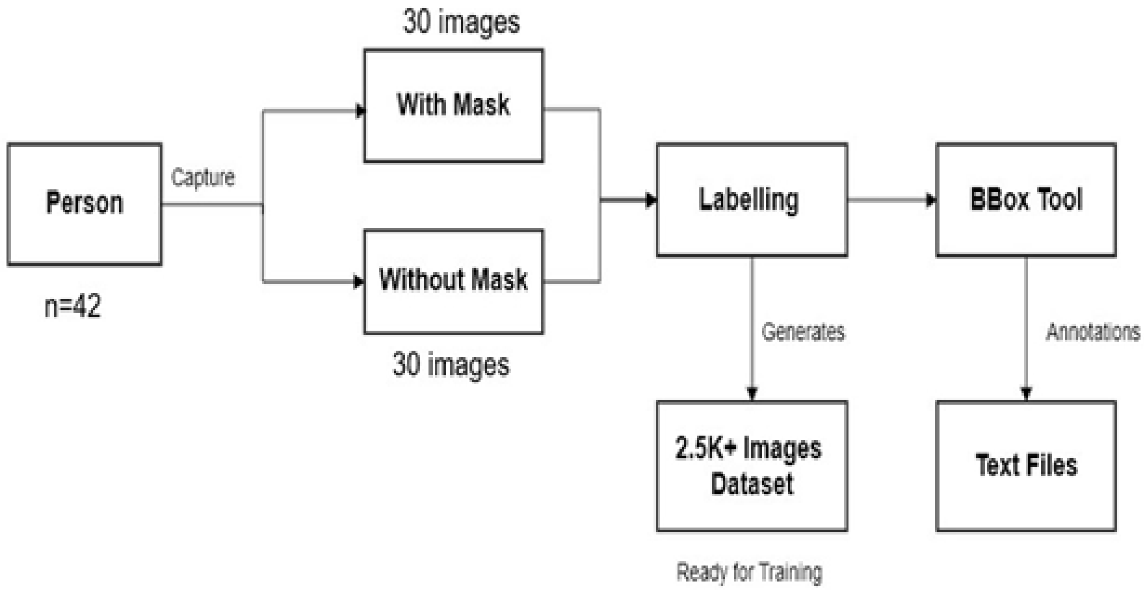


Figure 3.3: Pipeline for dataset generation

2. without mask : 1260 images

Each person's 30 images with and without mask have been captured with different angles and postures.

3.3.2 Dataset Pre-Processing

The bounding box annotations are put for every image using bbox tool. Then using a converting tool class of every image class added .Finally the text files for every image is generated consisting of the coordinates of the bounding box .Resizing the images to respective architecture. For InceptionV3 it is 299x299x3. Re scaling of each pixels into 0-1 and making it floating point.Data augmentation to generate images of different angles.

3.3.3 Dataset Augmentation

Image data augmentation is a technique that's accustomed artificially expand the training dataset by creating modified versions of the pictures present within the dataset. Data augmentation helped us to increase the size of the dataset, improve the performance of the model by reducing the over-fitting problem, introduce variability in the dataset without actually collecting new data. The dataset was augmented to generate more images for training the InceptionV3 model. For each epoch, augmentation of the whole dataset was done, so for 20 epochs and a total dataset of 2520 images, a total of 50400 augmented images were generated for training purposes. The augmentation process is followed by keeping the rotation range to

20 i.e. each image in the dataset is augmented to generate 20 varied images, zoom range to 0.15, width shift range to 0.2, height shift range to 0.2, shear range to 0.15, and horizontal flip was set to true Boolean for every image.

Figure 3.4: Image before augmentation

Figure 3.5: Images generated after augmentation

Chapter 4

IMPLEMENTATION

This chapter gives a brief description about the implementation details of the system like how the masked face detection and recognition is performed and which CNN architecture is applied. The workflow of our proposed methodology can be seen in Figure 3.1. According to which this chapter is divided into three modules, face mask detection, face recognition with mask and finally masked face recognition.

4.1 Face Mask Detection

Face mask detection is a process in which we can detect whether the person is wearing mask or not by using facial landmarks which consists of face, eyebrows, eyes, mouth and jawline. And to get the facial regions we select the region of interest(ROI). we extract the features of nose, eyes and face from it and compare with the image in the dataset to show the accurate output.

Authors in paper[6] have mentioned about MobileNetV2 as the base version for face mask detection, what occurs right here is the images are transformed in an NumPy array. further they provide the input images to MobileNetV2 model to test for the correct result. In another paper which we reffered[7] they have used faster-RCNN model to detect the face mask, first off they resize the photo to the specified length for RCNN model, which is 600x800x3. And the output function map of 37x50x256 dimension. RPN is attached to a convolutional layer with 3×3 filters, 1 padding, 512 output channels. The output is attached to 2 1×1 convolutional layer for class and box-regression. Here, each anchor has 3×3 corresponding boxes in the original image, which means there are $37 \times 50 \times 9 = 16{,}650$ boxes in the original photo. Further, a non-max suppression is carried out to make certain that there is no overlapping for the proposed regions. Now, Fast R-CNN, ROI pooling is used for those proposed regions (ROIs). The output is $7 \times 7 \times 512$. After that, they have got flatten this layer with a few fully connected layers. The very last step is a soft-max function for classification and linear regression to restore the boxes' location

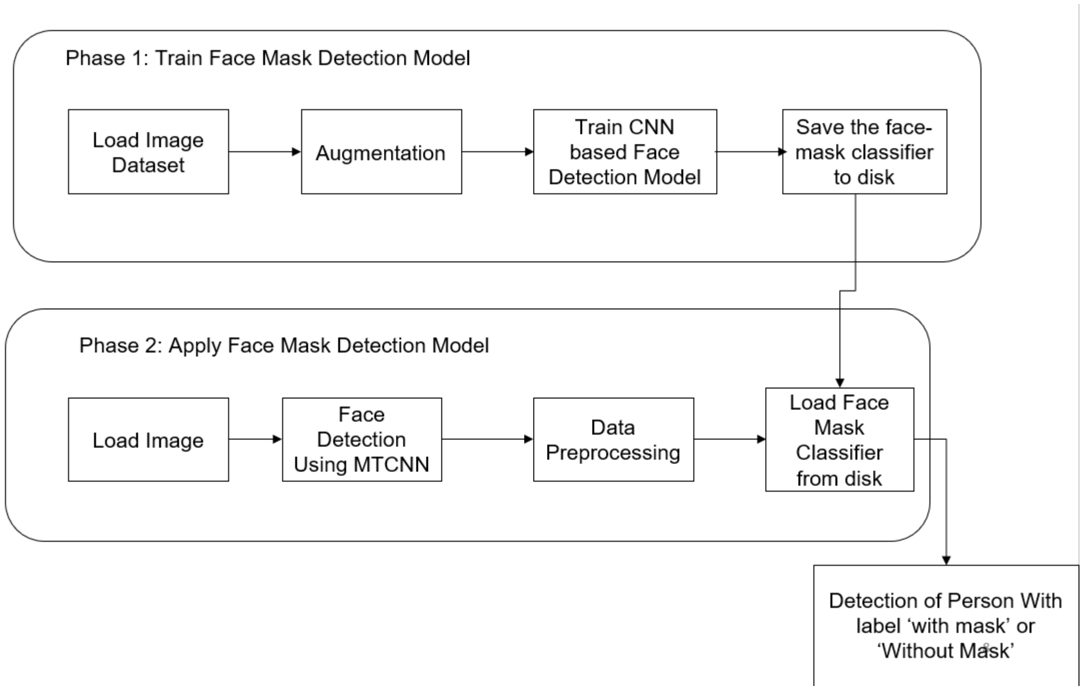


Figure 4.1: Proposed framework for face mask detection using InceptionV3

From Figure 4.1 we observe that, for face mask detection, there are 2 phases present. Phase1: train face mask detection model and phase2: apply face mask detection model.

Phase 1 - Train Face Mask Detection Model

Figure 4.1 depicts that, in first phase we will load the image dataset and then we will do the augmentation. Here we are using, Augmentation as an image generator and also we are doing the acquisition, in that we are removing the blur images, dark images and also we are resizing the images according to the CNN model used. Then we will train CNN based face mask detection model in which we are dividing our dataset into two parts training and testing and we are dividing in the ratio of 20:80. And here we are using the pre-trained weight of different CNN and we have used dropout in between the layers and we have taken dropout value as 0.5. After that, we save the face mask classifier model to the disk.

Phase 2 - Apply Face Mask Detection Model

In Figure 4.1 phase two, we are first loading the images after that we are doing face detection in which we have used the MTCNN model. And for face detection, we are extracting the Region of interest(ROI) like nose, mouth, eyes, eyebrow jawline. Then we are again doing data pre-processing for the images, then we are loading the saved face mask classification model from the disk. After that, the output of the image will be given with the label "With_mask" or "Without_mask".

CNN models for face mask detection

In CNN first, we have tried with MobileNetV2, there we got an accuracy of about 96%. After that we have tried the YoloV4 model in that CSP darknet 53 is used as a backbone that can enhance the learning capability of CNN. and here we got an accuracy of about 97%.And at last, we tried with the Inception V3 model in which we got an accuracy of 98%. We have tested almost 200 images manually and in all the cases we got the accurate result and in some of the test cases MobileNetV2 fails and on some test cases, YoloV4 fails but the InceptionV3 model has given accurate results in all the test cases.

Test cases used:
1. Mouth Covered with hands.
2. Applied Facepack on the face.
3. Partially wearing the mask.
4. Colour of mask and face are the same.

MobileNetV2

Figure 4.2 depicts that, the framework for MobileNetV2 model, there is a convolution neural community structure that seeks to carry out nicely on cellular devices. As a whole, the structure of MobileNetV2 carries the preliminary completely convolution layer with 32 filters, observed through 19 residual bottleneck layers. In MobileNetV2, there are varieties of blocks. One is the residual block with a stride of 1. Another one is blocked with the stride of two for downsizing. There are three layers for each varieties of blocks. This time, the primary layer is 1×1 convolution with ReLU6.

The 2nd layer is the depth-sensible convolution. The 0.33 layer is every other 1×1 convolution however with none non-linearity. It is alleged that if ReLU is used again, the deep networks handiest have the energy of a linear classifier at the non-0 extent a part of the output domain.And there's an growth issue t. And t=6 for all primary experiments.If the enter were given sixty four channels, the inner output could get sixty four×t=sixty four×6=384 channels.

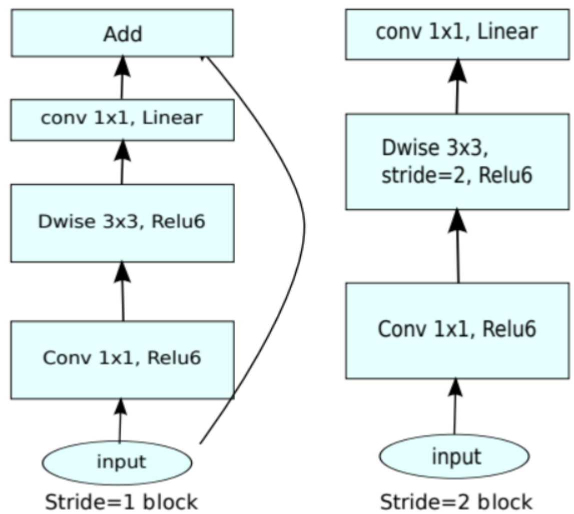


Figure 4.2: Architecture of MobileNetV2

InceptionV3

Figure 4.3 depicts that, the InceptionV3 model is a 48-layer deep convolution neural community structure from the Inception own circle of relatives that carries Label Smoothing, Factorized 7 x 7 convolutions, and using an auxiliary classifier to transmit label records deeper down the community (along side using batch normalisation for layers withinside the facet head). The community's picture enter length is 299 via way of means of 299 pixels. The version is made of symmetric and uneven constructing blocks which incorporates convolutions, common pooling, max pooling, concats, dropouts, and absolutely linked layers. Batch norm is carried out to activation inputs and is used substantially in the course of the version. Softmax is used to calculate loss.

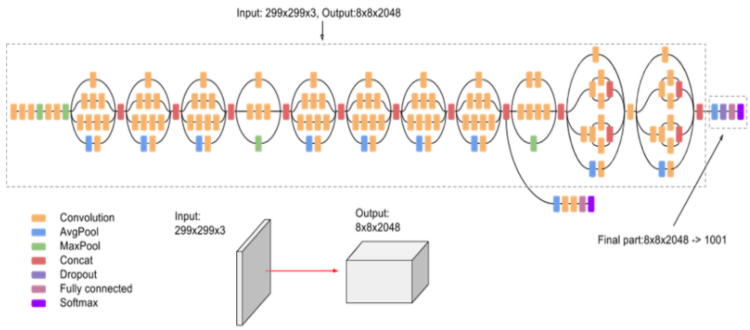


Figure 4.3: Architecture of InceptionV3

YOLOv4

Figure 4.4 depicts that the YOLOv4 model became a real-time item detection version posted in April 2020 that carried out contemporary overall performance at the COCO dataset. It works with the aid of using breaking the item detection assignment into pieces, regression to perceive item positioning through bounding packing containers and type to decide the item's class. This implementation of YOLOv4 makes use of the Darknet framework.

There are a few computerized methods to enhance the scope and length of your education set thru augmentation. The YOLOv4 education pipeline does augmentations automatically.

The enter decision determines the range of pixels with the intention to be surpassed into the version to examine and are expecting from. A massive pixel decision improves accuracy however trades off with slower education and inference time. A large pixel decision can assist your version locate small objects. YOLO version pixel decision should be a more than one of 32. A popular decision length to pick is 416x416.

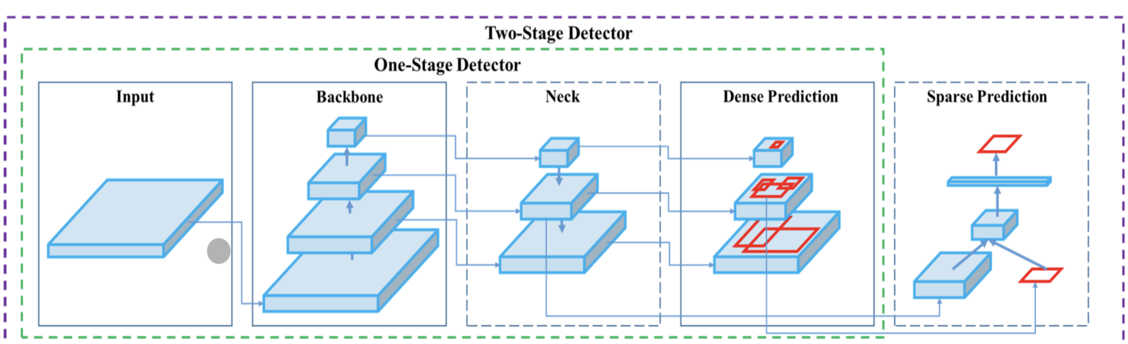


Figure 4.4: Architecture of YOLOv4

4.2 Face Recognition with mask

Face recognition with mask is a process of identifying or verifying the identity of a person based on their face from a digital image. It is used to recognise people from images, videos or in real-time. Face recognition with mask can be divided mainly into three categories [14] namely face detection, feature extraction and lastly feature matching. And the output of each method will be the input to the next method.

Face recognition with mask is a complex task which relies majorly on eigenvectors. The traditional or early methods [8] for implementing face recognition model used PCA for feature extraction and Feed Forward back propagation neural networks for recognition. Latter on for small training set examples and for large categories the Similarity metrics [9] was being used for recognition and verification. Now there are modern techniques or methods for implementing face recognition such as DeepFace [10], which employ a four-stage pipeline that follows the flow of, detect, align, represent and classify. This model has an advanced approach since it is combined with 3-D face modelling and piecewise affine transformation to achieve the task of face recognition. VGGFace [11] is another neural network model that can be used for face recognition with mask model, where a 2048 vector dimensional descriptor face embedding is produced. Then these are L2 normalized [12] and using the cosine distance the similarity between the faces is calculated. FaceNet [13] maps the face images extracted to the Euclidean space that in turn will generate 128-Dimensional vector embeddings as a measure of similarity. To increase the efficiency and achieve good results, a novel online mining triplet technique was introduced. FaceNet has attained good accuracy with more reliable results on face recognition benchmark datasets. Therefore, after going through various academic literature papers, publications and blogs, the FaceNet model is chosen for implementing the Masked Face Recognition part.

FaceNet [15] is a face recognition model which is introduced by Google researches by integrating machine learning in processing face recognition. It is also known as one-shot learning model. FaceNet gives unified embedding for face verification, recognition and clustering tasks. It directly trains the face by mapping each face image into the Euclidean space where the distance consists of facial model similarities. FaceNet training method uses triplet loss that helps in minimizing the gap of anchor and positive, and maximizes the gap of anchor and negative image. FaceNet uses deep convolutional networks to optimize its embedding, and fetches 128 vector embeddings as a feature extractor. FaceNet is trained using training face thumbnails ranging from 100M to 200M, which consist of 8M different identities with varying input size from 96x96 pixels to 224x224 pixels. It is widely used in cases where there is scarcity of dataset, and still it attains a good accuracy.

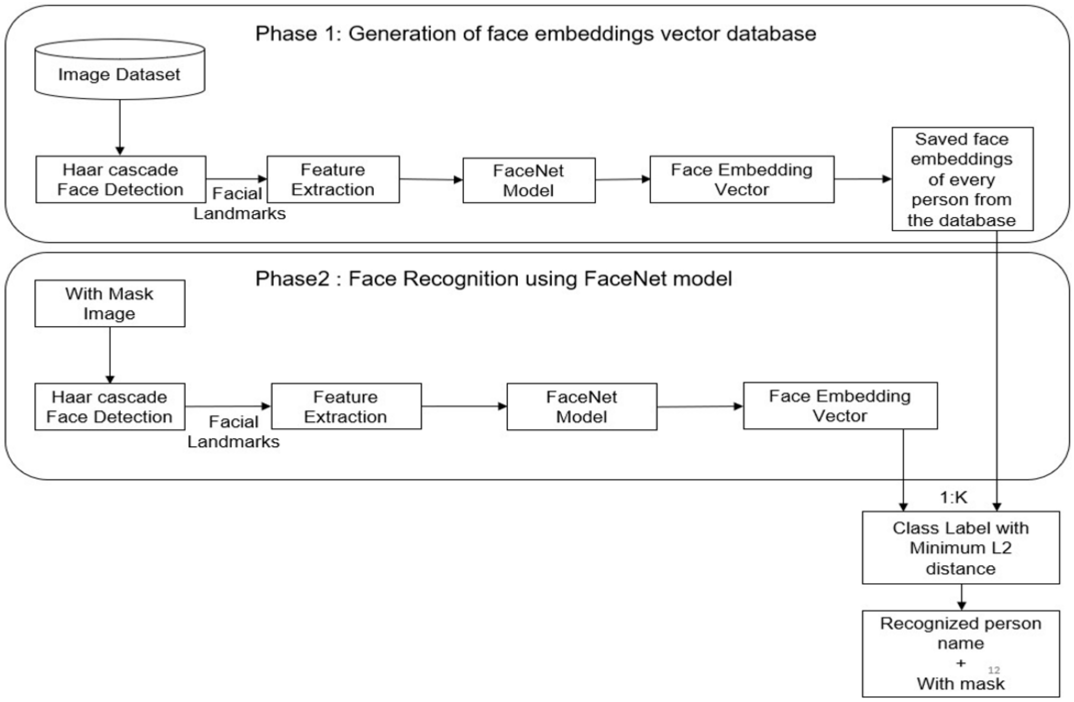


Figure 4.5: Proposed framework for face recognition with mask using FaceNet

From Figure 4.5 we can observe that, shows the framework of how the face recognition takes place for masked faces using the FaceNet model. This process has two major phases, in first phase generation of 128-Dimensional vector embeddings of the masked face from database takes place. And the face recognition for the masked faces using the FaceNet model is performed in the second phase.

Phase 1: Generation of face vector embeddings database.

From Figure 4.5 we observe that, in the first phase we will consider only one masked face image of each person from the dataset which we have developed, and this will be our final image dataset(42 images of different people in our case) used for face recognition model. Then these images are passed to the Haar cascade classifier for face detection which helps in extraction of facial landmarks and features. These extracted features are passed to the pre-trained FaceNet model for the generation of 128-D vector embeddings. And finally, the face embeddings of every person in the database is saved.

Phase 2: Recognition of the face using FaceNet model.

Now, after the completion of the phase 1, we have the corpus of 128-dimensional embeddings corresponding to the names of the person. In the second phase, the input image will be the masked face image which is the output of the InceptionV3 models output/module1's output. This image is then passed through Haar cascade classifier for feature extraction and then through pre-trained network of FaceNet model for generating 128-D vector embeddings. These generated 128-D vector embeddings are in turn compared to the stored embeddings using Euclidian (L2) distance. If the lowest distance between the captured embedding (i.e. the embedding obtained from input image) and the stored embeddings (i.e. the embedding of the person which is already stored on the disk and is the output of the first phase) is less than a threshold value, then the system will recognise the person with his name corresponding to that lowest distant embedding (Example "Prasad").

4.3 Masked Face Recognition

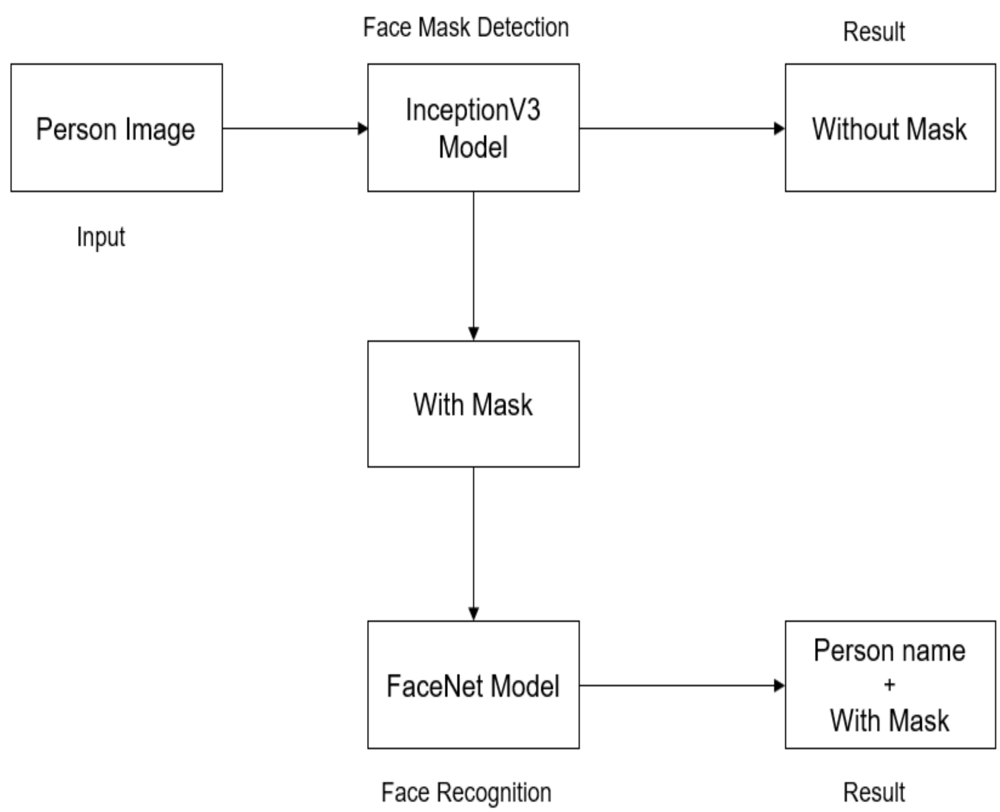


Figure 4.6: Pipeline for masked face recognition

Figure 4.6 depicts that, the masked face recognition model, we are taking an input image and that input image is sent to face mask detection model in which first we have tried with MobileNetV2 and after that, we have tried with YoloV4 and lastly, we have tried with InceptionV3 model.

among all the three models, in the case of InceptionV3 we get the best result so for the face mask detection we have proceeded with InceptionV3 based model and in this face detection part if the output of the face mask detection is "Without_mask' then it will simply give the result with the label "Without_mask' on the image.

And, if the face mask detection result is "With_mask' then we will further proceed for the face recognition part and in this, we have tried with two models VGGFace2 and FaceNet and in the case of FaceNet we got the best result so for the recognition part we have proceeded with the FaceNet model.

So if the person is With_mask' then we do the recognition also and the final output will be with the label "Person name +With_mask".

Chapter 5

RESULTS AND DISCUSSIONS

In this chapter, we will be briefing about the results of mask detection based on InceptionV3, face recognition based on FaceNet and masked face recognition based on the combined model.

5.1 Dataset details

The dataset generated contains 2520 images captured from 42 people having both, masks (1260 images) and without a mask (1260 images). Each person's 30 images with and without a mask have been captured with different angles and postures.

5.1.1 Data augmentation

Here, augmentation was used as a image generator.Augmentation of whole dataset was done, so for 20 epochs and total dataset of 2520 images, a total of 50400 augmented images were generated for training purpose.

Figure 5.1: Sample image from the dataset before augmentation

Figure 5.2: Images generated after augmentation

5.1.2 With Mask Image Dataset:

A total of 1260 images were generated, each person's 30 images with masks have been captured with different angles and postures.

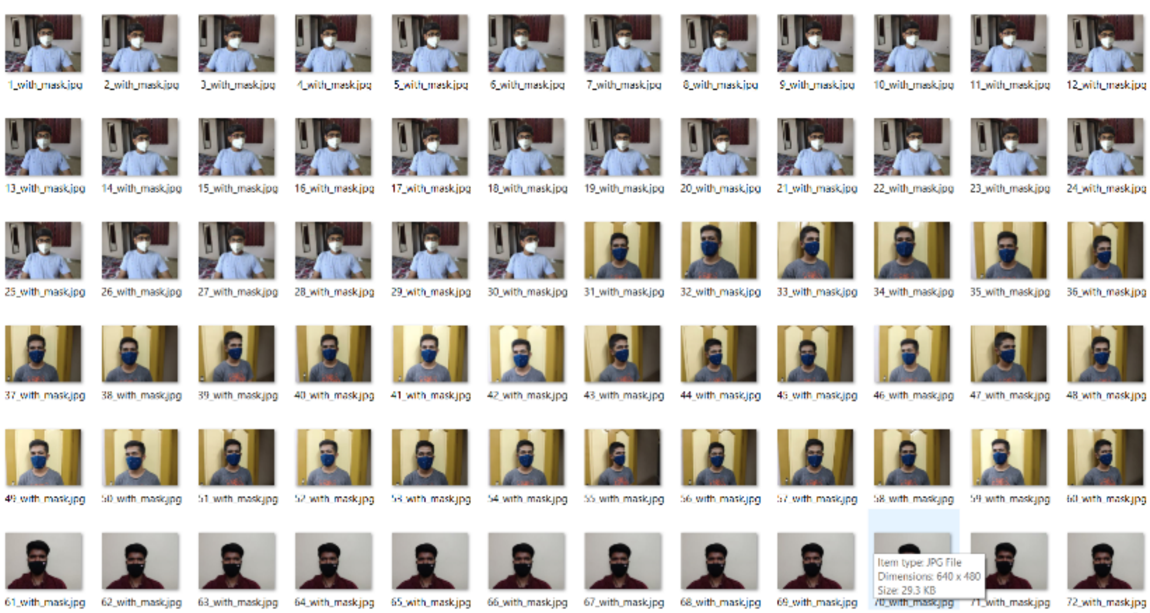


Figure 5.3: Masked images dataset

5.1.3 Without Mask Image Dataset:

Each person's 30 images without a mask have been captured with different angles and postures making it a total of 1260 images.

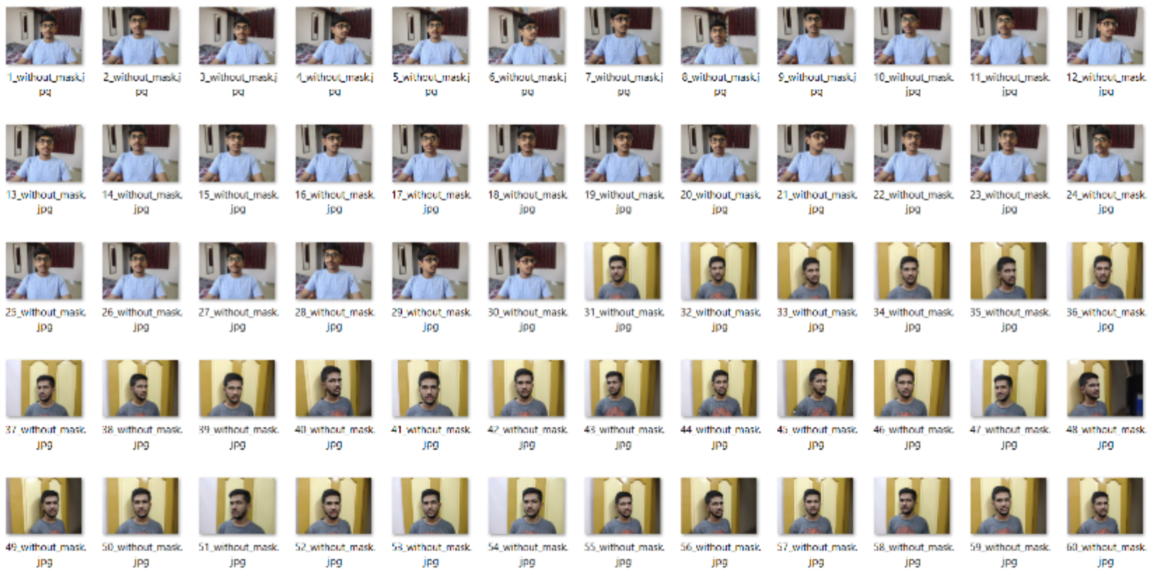


Figure 5.4: Without mask images dataset

5.2 Evaluation metrics

In a classification task, the precision for a class is the number of true positives divided by the total number of elements labelled as belonging to the positive class. Recall in this context is defined as the number of true positives divided by the total number of elements that belong to the positive class (i.e., the sum of true positives and false negatives, which are items that were not labelled as belonging to the positive class but should have been).

Precision Rate- It is given by the ratio of correctly predicted positive observations to the total predicted positive observations. Low false positive rate relates to high precision.

$$\text{Precision} = \frac{\text{True Positive}}{\text{True Positive} + \text{False Positive}}$$

$$= \frac{\text{True Positive}}{\text{Total Predicted Positive}}$$

Figure 5.5: Precision rate

Recall - It is given by the ratio of correctly predicted positive observations to the all observations in actual class.

$$\text{Recall} = \frac{True\ Positive}{True\ Positive + False\ Negative}$$

$$= \frac{True\ Positive}{Total\ Actual\ Positive}$$

Figure 5.6: Recall rate

F1-score - It is given by the weighted average of Precision and Recall. This score takes both false positives and false negatives into consideration.F1 is usually more useful than accuracy in case of uneven class distribution.

$$\text{F1} = 2 \times \frac{Precision * Recall}{Precision + Recall}$$

Figure 5.7: F1-score

Accuracy - Accuracy is calculated as the number of all correct predictions divided by the total number of the dataset. The best accuracy is 1.0, whereas the worst is 0.0.

$$\mathrm{ACC} = \frac{\mathrm{TP} + \mathrm{TN}}{\mathrm{TP} + \mathrm{TN} + \mathrm{FN} + \mathrm{FP}} = \frac{\mathrm{TP} + \mathrm{TN}}{\mathrm{P} + \mathrm{N}}$$

Figure 5.8: Accuracy

This is the combined model for the face mask detection and recognition so we cannot calculate the F1-score, recall rate and precision so we have calculated the accuracy of the model manually.

We have taken 100 images and stored them in a file and after that, we have run the whole file for the final output and after getting the final result we see that 88 out of 100 have given the correct result. So from that, we can say that our model is giving about 88% accuracy.

5.3 Results for face mask detection

For face mask detection, we are using InceptionV3 model and we got accuracy for this model as 98%. and the results obtained for various cases are shown below.

Input and Output of Face mask detection for various cases:

Table 5.1: Results for face mask detection

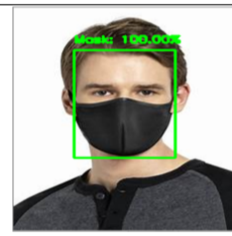
1. Positive case		
2. Negative case		

Table 5.2: Results for face mask detection on various test cases

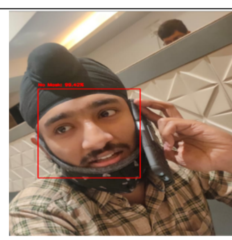
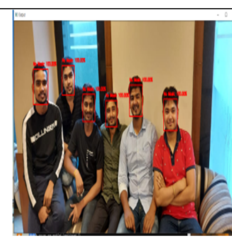
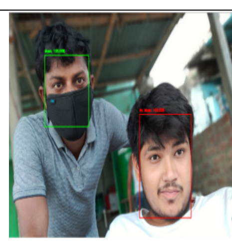
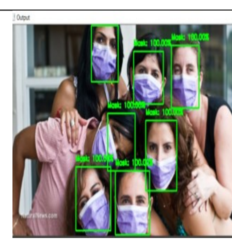
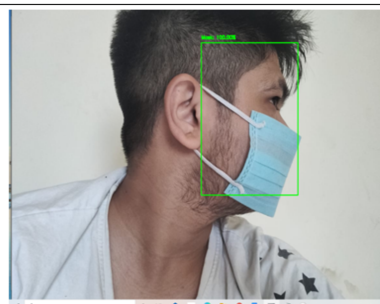
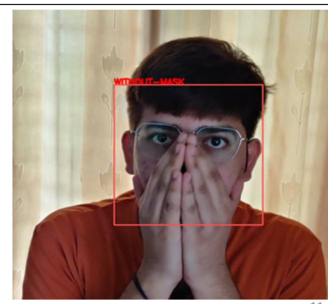
3. Negative case		
4. Mixed case		

Table 5.3: Results for face detection on various test cases

5.Test cases	
6.Test cases	 

- **Comparison of different models for face mask detection:**

  1. MobileNetV2
  2. YoloV4
  3. InceptionV3

Table 5.4: Comparison of different models for face mask detection

Content	Dataset images	Accuracy	F1-Score
MobileNetV2	2520	0.96	0.96
YoloV4	2520	0.97	–
InceptionV3	2520	0.98	0.96

From Table 5.4 we can observe that, MobileNetV2 gives 96% accuracy, YoloV4 gives 97% accuracy and Inception V3 gives 98% accuracy. So by this we can conclude that Inception V3 is giving better results than the remaining models which are used.

5.4 Results for face recognition :

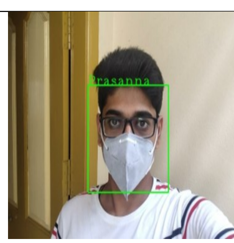
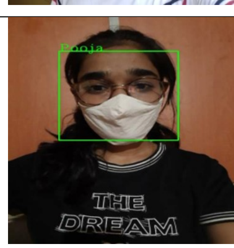
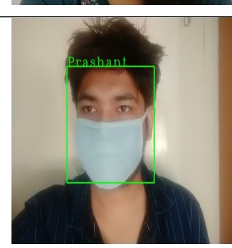
For face recognition, FaceNet is used to give us accurate results. The accuracy obtained through FaceNet model is better as compared to other models . And the results obtained for various cases are shown in the following tables.

Obtained Results for Face Recognition

Table 5.5: Results for face recognition

1. Person with mask		
2. Person With Mask		

Table 5.6: Results for face recognition on different test cases:

5.5 Results for masked face recognition

In masked face recognition, we combine both the inception V3 model for face mask detection and FaceNet model for face recognition. for the recognition of the person wearing mask.

Final Results:

Table 5.7: Results for masked face recognition

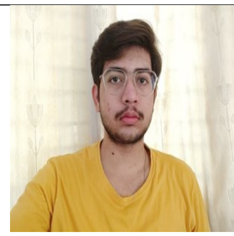
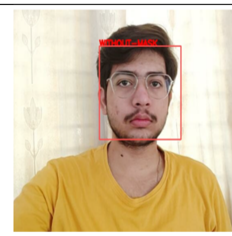
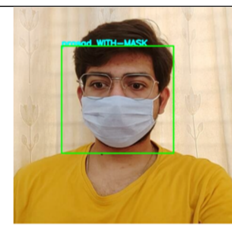
1. Person without mask		
2.Person with mask		

Table 5.8: Results for masked face recognition with occlusion

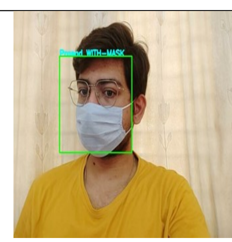
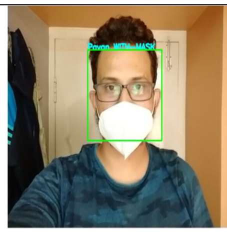
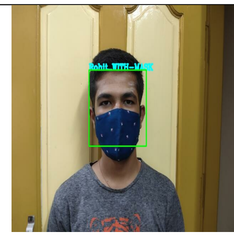
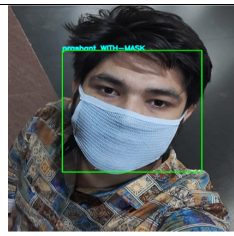
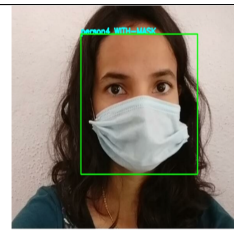
3. Person with mask		
4.Person with mask		

Table 5.9: Results for masked face recognition

Here when we give an input image, it goes to the face detection model and if the person in the image is not wearing mask then we the output as the "without_mask" in red bounding box and if the person is wearing mask then we get the output as "person name + without_mask" in green bounding box.

Chapter 6

CONCLUSION AND FUTURE SCOPE

We have proposed a framework for masked face recognition model. A dataset containing 2520 images of 42 individuals was generated. An InceptionV3 model is being used in the proposed architecture to obtain better results for face mask detection which when compared to other CNN models i.e. YOLOv4 and MobileNetV2 gave better accuracy. FaceNet model is used for face recognition with mask. As FaceNet is a one-shot learning model which is preferred majorly in case where there is scarcity of datasets and still the model obtains good accuracy. As the dataset generated contains few images and since those are masked faces, the extracted features are also less, so FaceNet model best suits to this condition. These models i.e. InceptionV3 and FaceNet were then combined to form the high level architecture of the system, based on the constrain that only masked faces must be considered for recognition.

The model can be further fine tuned to improve the accuracy and recognize the multiple faces present in the image. Our work can be extended to authenticate people at public places such as airports, offices, hospitals, schools, etc., to ensure that the safety standards are maintained and people abide by the rules and regulations.

Bibliography

[1] Jiang, Mingjie, Xinqi Fan, and Hong Yan. "'Retinafacemask: A face mask detector." *Innovation and Technology Commission, and City University of Hong Kong.*, 2020.

[2] Cahyono, Ferry, Wirawan Wirawan, and Reza Fuad Rachmadi. "Face Recognition System using Facenet Algorithm for Employee Presence." *2020 4th International Conference on Vocational Education and Training (ICOVET),*pp. 57-62. IEEE, 2020.

[3] Wang, Zhongyuan, et al. "Masked face recognition dataset and application." , *arXiv preprint arXiv:2003.09093* , 2020.

[4] Ge, Shiming, et al. "Detecting masked faces in the wild with lle-cnns." ,*Proceedings of the IEEE conference on computer vision and pattern recognition.* ,2017.

[5] Aswal, Vivek, Omkar Tupe, Shifa Shaikh, and Nadir N. Charniya. "Single Camera Masked Face Identification." ,*2020 19th IEEE International Conference on Machine Learning and Applications (ICMLA), pp. 57-60.* IEEE, 2020.

[6] Rosebrock, Adrian."Covid-19: Face mask detector with opencv, keras/tensorflow, and deep learning." ,*Link: https://www. pyimagesearch. com/2020/05/04/covid-19-face-mask-detector-withopencv-keras-tensorflow-and-deeplearning* (2020)

[7] Ren, Shaoqing, et al."Faster r-cnn: Towards real-time object detection with region proposal networks." ,*Advances in neural information processing systems. 91-99.* 2015.

[8] M. A. Turk and A. P. Pentland,"Face recognition using eigenfaces" ,*IEEE Computer Society Conference on Computer Vision and Pattern Recognition,* 1991.

[9] S. Chopra, R. Hadsell and Y. LeCun, "Learning a similarity metric discriminatively with application to face verification" ,*IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)* 2005.

[10] Y. Taigman, M. Yang, M. Ranzato and L. Wolf,"DeepFace: Closing the gap to human-level performance in face verification" ,*", IEEE Conference on Computer Vision and Pattern Recognition,.* 2014.

[11] Parkhi, Omkar M., Andrea Vedaldi, and Andrew Zisserman,Deep face recognition." 2015.

[12] Aswal, Vivek, Omkar Tupe, Shifa Shaikh, and Nadir N. Charniya,"Single Camera Masked Face Identification." ,*"In 2020 19th IEEE International Conference on Machine Learning and Applications (ICMLA), pp. 57-60. ,.* IEEE, 2020.

[13] F. Schroff, D. Kalenichenko and J. Philbin,"FaceNet: A unified embedding for face recognition and clustering", *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*2015

[14] Brownlee, Jason. Deep learning for computer vision: image classification, object detection, and face recognition in python. Machine Learning Mastery, 2019.

Plagiarism Report

Project Title: Masked Face Recognition

Team No: D3

Project Domain: Image Processing

Masked Face Recognition

ORIGINALITY REPORT

22%	13%	15%	11%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	Submitted to B.V. B College of Engineering and Technology, Hubli Student Paper	2%
2	arxiv.org Internet Source	2%
3	www.coursehero.com Internet Source	2%
4	Vivek Aswal, Omkar Tupe, Shifa Shaikh, Nadir N. Charniya. "Single Camera Masked Face Identification", 2020 19th IEEE International Conference on Machine Learning and Applications (ICMLA), 2020 Publication	2%
5	Sunil Singh, Umang Ahuja, Munish Kumar, Krishan Kumar, Monika Sachdeva. "Face mask detection using YOLOv3 and faster R-CNN models: COVID-19 environment", Multimedia Tools and Applications, 2021 Publication	1%
6	dokumen.pub Internet Source	1%

7	ieeexplore.ieee.org Internet Source	1 %
8	www.tandfonline.com Internet Source	1 %
9	repository.its.ac.id Internet Source	1 %
10	Submitted to Higher Education Commission Pakistan Student Paper	1 %
11	Submitted to University of Cape Town Student Paper	1 %
12	wikimili.com Internet Source	1 %
13	Submitted to Bahcesehir University Student Paper	1 %
14	Submitted to Technological Institute of the Philippines Student Paper	1 %
15	Submitted to Multimedia University Student Paper	<1 %
16	Ivan William, De Rosal Ignatius Moses Setiadi, Eko Hari Rachmawanto, Heru Agus Santoso, Christy Atika Sari. "Face Recognition using FaceNet (Survey, Performance Test, and Comparison)", 2019 Fourth International	<1 %

Conference on Informatics and Computing (ICIC), 2019
Publication

17	Li Zhang, Liang Zhang, Keli Xiao, Qi Liu. "Forecasting price shocks with social attention and sentiment analysis", 2016 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM), 2016 Publication	<1 %
18	"Proceedings of International Conference on Trends in Computational and Cognitive Engineering", Springer Science and Business Media LLC, 2021 Publication	<1 %
19	Minghui Guo, Yongzhao Du. "Classification of Thyroid Ultrasound Standard Plane Images using ResNet-18 Networks", 2019 IEEE 13th International Conference on Anti-counterfeiting, Security, and Identification (ASID), 2019 Publication	<1 %
20	Isabel Amaya-Rodriguez, Isabel Amaya-Rodriguez, Javier Civit-Masot, Francisco Luna-Perejon et al. "Chapter 12 ResNet", Springer Science and Business Media LLC, 2021 Publication	<1 %
21	Submitted to Florida Institute of Technology	

Student Paper

<1 %

22	mobisoftinfotech.com Internet Source	<1 %
23	Bin Xue, Jianpeng Hu, Pengming Zhang. "Intelligent detection and recognition system for mask wearing based on improved RetinaFace algorithm", 2020 2nd International Conference on Machine Learning, Big Data and Business Intelligence (MLBDBI), 2020 Publication	<1 %
24	Bhushan Bhagwan Gawde. "A Fast, Automatic Risk Detector for COVID-19", 2020 IEEE Pune Section International Conference (PuneCon), 2020 Publication	<1 %
25	Submitted to AUT University Student Paper	<1 %
26	Submitted to Gaborone Institute of Professional Studies Student Paper	<1 %
27	Walid Hariri. "Efficient Masked Face Recognition Method during the COVID-19 Pandemic", Research Square, 2020 Publication	<1 %
28	Milan Tripathi. "Analysis of Convolutional Neural Network based Image Classification	<1 %

Techniques", Journal of Innovative Image Processing, 2021
Publication

29 Submitted to University of Maryland, University College <1%
Student Paper

30 Nzurumike Obianuju Lynda. "Systematic survey of convolutional neural network in satellite image classification for geological mapping", 2019 15th International Conference on Electronics, Computer and Computation (ICECCO), 2019 <1%
Publication

31 Dragos-Vasile Bratu, Sorin-Aurel Moraru, Ligia Georgeta Guseila. "A Performance Comparison between Deep Learning Network and Haar Cascade on an IoT Device", 2019 International Conference on Sensing and Instrumentation in IoT Era (ISSI), 2019 <1%
Publication

32 Mohamed Loey, Gunasekaran Manogaran, Mohamed Hamed N. Taha, Nour Eldeen M. Khalifa. "Fighting against COVID-19: A novel deep learning model based on YOLO-v2 with ResNet-50 for medical face mask detection", Sustainable Cities and Society, 2020 <1%
Publication

33 pyimagesearch6.rssing.com

	Internet Source	<1 %
34	"Advances in Visual Computing", Springer Science and Business Media LLC, 2020 Publication	<1 %
35	Yashraj Motwani, Saambhavi Seth, Devang Dixit, A. Bagubali, R. Rajesh. "Multifactor door locking systems: A review", Materials Today: Proceedings, 2021 Publication	<1 %
36	"Advanced Computing", Springer Science and Business Media LLC, 2021 Publication	<1 %

Exclude quotes	Off	Exclude matches	Off
Exclude bibliography	On		