A

Final Year Project Report

On

Masked Face Recognition Based Attendance System Using Deep Learning

Submitted in partial fulfillment of the requirements

of the degree of

Bachelor of Engineering

by

Mokashi Karishma Chandrakant	**(117IT1065B)**
Kodag Supriya Tanaji	**(117IT1429B)**
Pawar Saurabh Raghunath	**(118IT3326A)**

Under the guidance of

Prof. Yogesh Shahare

Department of Information Technology

Mahatma Gandhi Mission's College of Engineering & Technology

Kamothe, Navi Mumbai – 410 209

University of Mumbai

Academic Year: 2020-21

CERTIFICATE

This is to certify that the mini project entitled **“Masked Face Recognition Based Attendance System Using Deep Learning”** is a bonafide work of **Mokashi Karishma Chandrakant (117IT1065B), Kodag Supriya Tanaji (117IT1429B), Pawar Saurabh Raghunath (118IT3326A)** submitted to the University of Mumbai in partial fulfillment of the requirement for the award of the degree of **“Undergraduate”** in **“Information Technology”.**

Prof. Yogesh Shahare
Project Guide

External Examiner

Prof. Vijay Bhosale
Head of Department

Dr. Geeta S. Lathkar
Director

<u>DECLARATION</u>

We declare that this project report entitled **"Masked Face Recognition Based Attendance System Using Deep Learning"** represents our ideas in our own words and where others ideas or words have been included, we have adequately cited and referenced the original sources. We also declare that we have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact/source in my submission. We understand that any violation of the above will be cause for disciplinary action by the Institute and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been taken when needed.

Date:

Place:

(Signature)

TABLE OF CONTENTS

ACKNOWLEDGEMENT

The success and final outcome of this project required a lot of guidance and assistance from many people and we are extremely fortunate to have got this all along the completion of our project work. Whatever we have done is only due to such guidance and assistance and we would not forget to thank them.

It is matter of great pleasure for us to submit the project report on **“Masked Face Recognition Based Attendance System Using Deep Learning”**, as a part of our curriculum.

First and foremost, we would like to thank to our Director **Dr. Geeta Lathkar**, for giving us an opportunity to do the project work. We would like to thank our H.o.D Prof**. Vijay Bhosale** and subject incharge **Prof. Yogesh Shahare**, for the valuable guidance and advice. He inspired us greatly to work in this project. His willingness to motivate us contributed tremendously to our project.

And last but not the least a special thanks goes to my team members, who helped me to assemble the information and gave suggestions to complete our project.

ABSTRACT

A face recognition attendance system uses face recognition technology to spot and verify an individual using the person's countenance and automatically marks attendance. But with the spread of COVID-19, millions don masks across the planet, which affects the accuracy of the face identification system. The existing face recognition solutions are not any longer reliable when wearing a mask and removing masks for passing authentication will increase the danger of viral infection. To this end, this work proposes deep learning structure based on Convolutional Neural Networks (CNN), trained and tested on images of people's faces with and without masks. The system can recognize a person's face and the attendance will only be recorded if the person is wearing a mask. Supervised learning is applied to train the dataset which is collected from various sources. The software can be used for different groups of individuals such as employees, students, etc. The system records and stores the data in real-time with existing dataset.

Keywords: Facial recognition, Mask detection, Attendance, COVID-19, Coronavirus, Masked Face, Deep learning.

CHAPTER-1
INTRODUCTION

INTRODUCTION

In this pandemic situation, wearing a mask became essential to avoid getting infected by coronavirus. Coronavirus spreads through contact with contaminated surfaces. At present, organizations using a biometric or card-based attendance system had to switch towards a face-based attendance system to avoid direct contact with the attendance system. But, facial recognition algorithms did not identify 20-50% of images of individuals wearing face masks according to a report from the National Institute of Standards and Technology. As the corona virus spreads rapidly we can’t trouble people by removing masks for attendance purposes. Here, deep learning based masked face recognition techniques can play a vital role in handling attendance with a masked face. Our reliable method for face recognition is safer without any need to touch any device. It uses parts of the face that are not covered up, such as eyes, to verify their identity. The system can be used at various places like exam centers, government offices, MNCs, travel industry, wrongful or criminal deception can be reduced, police can monitor whether people are wearing masks.

Two types of tests have been carried out, first one if the system is able to detect whether a person is wearing a mask or not, the other is to determine if it is able to recognize a person if he or she is wearing a mask. The technology is 99% accurate when no mask is present. For the first case, we have trained CNN algorithm so that it can detect whether a person is wearing – or not – wearing – a mask. In the second case, face is compared with possible matches from the dataset and the mapped face is recognized using a deep learning-based method. We use a pre-trained deep-learning based model so as to extract features from the unmasked face regions out of the mask region.

PROPOSED SYSTEM :

In this paper, we apply pre-trained deep Convolutional neural networks (CNN) to extract the best features from the obtained regions like eyes, eyebrows, hairline and general shape of the face. Flask support is used for deployment as a python web framework in this project. The system is built in order to manage attendance for students in a class. It contains input images of students which can be collected during the admission process by the admin. If a new student is to be added to the database then the teacher can login and add details with name and image, this data will be saved in the classroom folder. We have set constraints like single face and frontal view for high accuracy. When a teacher wants to take attendance he/she has to login to the system and student simply have to face towards the webcam. First, the face will be detected and a frame can be seen for the face. Faces in existing datasets will be mapped while the camera scans the face. If the student is wearing a mask the system will detect the mask and recognize the person by discarding the masked face region and extract information through the uncovered part. Once, the person is recognized the attendance will be marked. If the person is not wearing the mask then the attendance will not be recorded. Teachers can download the attendance file whenever needed. The dataset used to train the classifier contains (1) masked and (2) unmasked images. In a masked dataset there are frontal face images with masks on their face and an unmasked dataset contains the same faces without masks on their face. So, basically the primary step is to discard the masked face region. Next, we apply pre trained deep Convolutional neural networks for feature extraction. Bag-of-features (BoF) paradigm is applied to get a slight

representation compared to the fully connected layer of classical CNN. Finally, Multi-layer perceptron (MLP) is applied for the classification process.

FEATURES OF THE SYSTEM :

User Friendly : The proposed system is user friendly because the retrieval and storing of data is fast and data is maintained efficiently. Moreover the graphical user interface is provided in the proposed system, which provides user to deal with the system very easily.

Reports are easily generated : Reports can be easily generated in the proposed system so user can generate the report as per the requirement (monthly) or in the middle of the session. User can give the notice to the students so he/she become regular.

Very less paper work : The proposed system requires very less paper work. All the data is feted into the computer immediately and reports can be generated through computers. Moreover work becomes very easy because there is no need to keep data on papers.

Computer operator control : Computer operator control will be there so no chance of errors. Moreover storing and retrieving of information is easy. So work can be done speedily and in time.

FEASIBILITY STUDY :

Economically Feasibility : Development of this application is highly economically feasible. The only thing to be done is making an environment with an effective supervision. The system being developed is economic with respect to School or Collage's point of view. It is cost effective in the sense that has eliminated the paper work completely. The system is also time effective because the calculations are automated which are made at the end of the month or as per the user requirement. The result obtained contains minimum errors and are highly accurate as the data is required.

Technical feasibility : The technical requirement for the system is economic and it does not use any other additional Hardware and software. Technical evaluation must also assess whether the existing systems can be upgraded to use the new technology and whether the organization has the expertise to use it. Install all upgrades framework into the .Net package supported widows based application. this application depends on Microsoft office and intranet service, database. Enter their attendance and generate report to excel sheet.

Behavioral Feasibility : The system working is quite easy to use and learn due to its simple but attractive interface. User requires no special training for operating the system. Technical performance include issues such as determining whether the system can provide the right information for the Department personnel student details, and whether the system can be organized so that it always delivers this information at the right place and on time using intranet services. Acceptance revolves around the current system and its personnel.

OBJECTIVE :

AI FACE RECOGNITION :AI based computer vision for face detection & face recognition.

MASK COPLIANCE:Inbuilt algorithm for optional mask detection

REAL PERSON DETECTION:Inbuilt algorithm for optional mask detection

HIGH ACCURACY AND SPEED:Face Recognition takes less than a second with an accuracy of >99.9%

SMART ALERTS:System alerts in cases of anomalies like employee not wearing a mask etc.

CHAPTER 02
METHODOLOGY

METHODOLOGY

2.1 FACE RECOGNITION

DIFFERENT APPROACHES OF FACE RECOGNITION:

There are two predominant approaches to the face recognition problem: Geometric (feature based) and photometric (view based). As researcher interest in face recognition continued, many different algorithms were developed, three of which have been well studied in face recognition literature.

Recognition algorithms can be divided into two main approaches:

1. **Geometric:** Is based on geometrical relationship between facial landmarks, or in other words the spatial configuration of facial features. That means that the main geometrical features of the face such as the eyes, nose and mouth are first located and then faces are classified on the basis of various geometrical distances and angles between features. (Figure 3)
2. **Photometric stereo:** Used to recover the shape of an object from a number of images taken under different lighting conditions. The shape of the recovered object is defined by a gradient map, which is made up of an array of surface normals (Zhao and Chellappa, 2006) (Figure 2)

Popular recognition algorithms include:

1. Principal Component Analysis using Eigenfaces, (PCA)
2. Linear Discriminate Analysis,

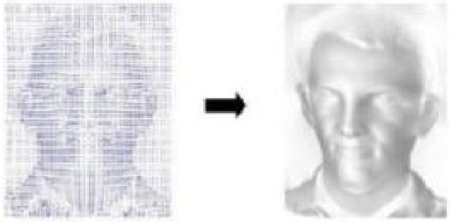


Figure 2 -Photometric stereo image.

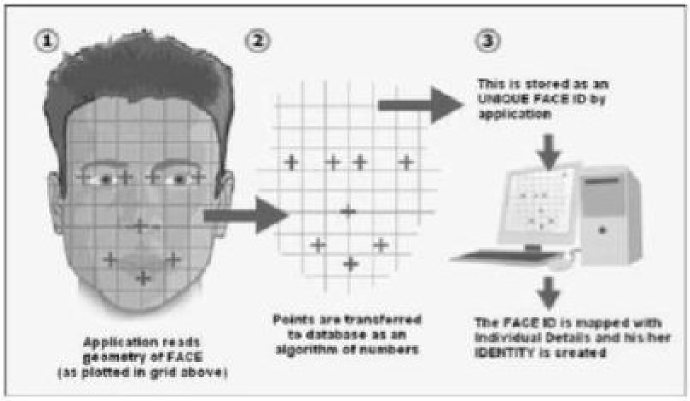


Figure 3 - Geometric facial recognition.

2.2 FACE DETECTION

Face detection involves separating image windows into two classes; one containing faces (tarning the background (clutter). It is difficult because although commonalities exist between faces, they can vary considerably in terms of age, skin colour and facial expression. The problem is further complicated by differing lighting conditions, image qualities and geometries, as well as the possibility of partial occlusion and disguise. An ideal face detector would therefore be able to detect the presence of any face under any set of lighting conditions, upon any background. The face detection task can be broken down into two steps. The first step is a classification task that takes some arbitrary image as input and outputs a binary value of yes or no, indicating whether there are any faces present in the image. The second step is the face localization task that aims to take an image as input and output the location of any face or faces within that image as some bounding box with (x, y, width, height).

The face detection system can be divided into the following steps:-

Pre-Processing: To reduce the variability in the faces, the images are processed before they are fed into the network. All positive examples that is the face images are obtained by cropping images with frontal faces to include only the front view. All the cropped images are then corrected for lighting through standard algorithms.

Classification: Neural networks are implemented to classify the images as faces or nonfaces by training on these examples. We use both our implementation of the neural network and the Matlab neural network toolbox for this task. Different network configurations are experimented with to optimize the results.

Localization: The trained neural network is then used to search for faces in an image and if present localize them in a bounding box. Various Feature of Face on which the work has done on:- Position Scale Orientation Illumination

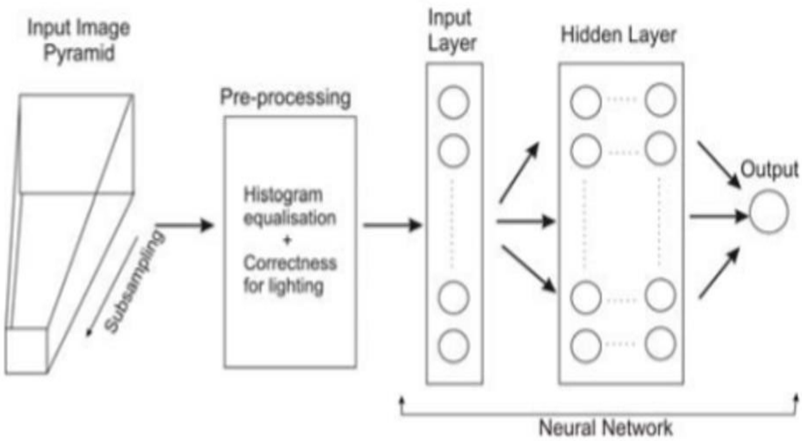


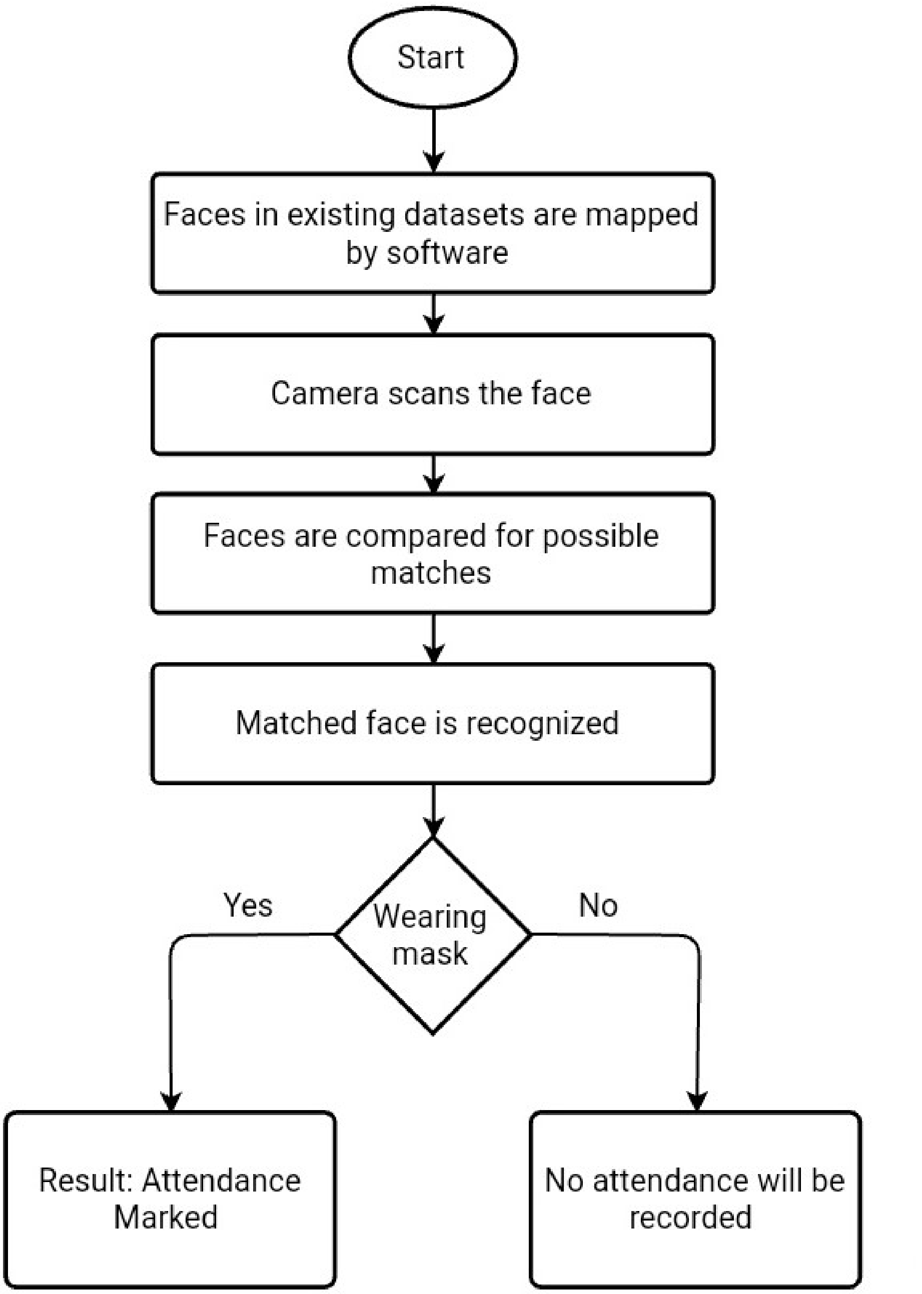
Fig: Face detection algorithm

2.3 MASK DETECTION

Amid the global crisis, new demand has emerged in the market, and that is of face mask detection. It is one such technology capable of detecting a face with a mask and verifying that person’s identity. It incorporates an AI-based pattern recognition system that uses biometric data of individuals. It extracts facial features and classifies them in different categories. Besides, it can also identify people without masks by generating an alarm or a notification to notify security or officials. They can see who has not covered faces with masks through software, mobile app, device, or a website

The face mask recognition system uses AI technology to detect the person with or without a mask. It can be connected with any surveillance system installed at your premise. The authorities or admin can check the person through the system to confirm their identity. The system sends an alert message to the authorized person if someone has entered the premise without a face mask. The accuracy rate of detecting a person with a face mask is 95-97% depending on the digital capabilities. The data has been transferred and stored automatically in the system to enable reports whenever you want.

Flow Chart of the System



CHAPTER 03

SYSTEM SPECIFICATION

.

SYSTEM SPECIFICATION

➢ HARDWARE REQUIREMENTS :

Minimum RAM : 4 GB

Hard Disk : 128 GB

Processor : Microprocessor (2.40 GHz)

Keyboard : 101 Keys

Mouse : Serial

➢ SOFTWARE REQUIREMENTS :

Visual Studio

Python 3.6.8

Flask web framework

XAMPP server

HTML

MS-Excel

CHAPTER.04
SYSTEM IMPLEMENTATION

.

SYSTEM IMPLEMENTATION

4.1 System Prerequisites

The proposed model focuses on recognizing the person with a mask on their face using deep learning structure based on CNN. The training of the network was carried out on a core i5 8th Generation PC with 8GB physical memory and standard webcam. In order to handle masked face recognition tasks, we used two types of masked face datasets, including Real-World Masked Face Recognition Dataset (RMFRD) and Simulated Masked Face Recognition (SMFRD). These datasets consist of many images of 1024 x 1024 dimension renamed with proper labels in folders named 'mask' and 'no-mask' that have been collected from various social networks and search engines[1]. The most important part of this dataset is for feature extraction using CNN. The database is created using phpMyAdmin which contains login information of two modules namely (1) Admin (2) Teacher. There is no student module so that he/she cannot login to the system to prevent proxy attendance. Then, a database is to be created of enrolled students.This is our testing data, where the images collected are without masks and the folder is named classroom. This step can be as a part of the admission process while collecting the student's information. This is accomplished using a dashboard on the web deployed using flask. Here, student's details like Name, Roll number, class, image of student, etc. to be uploaded by clicking on 'Upload Student Data'. This is stored physically in JSON format for faster access to image data and to recognize it.

4.2 Face Mask Detection

The face detector used Haar cascade algorithm to train the classifier, we used 'haarcascade-fronfalface-default.xml' file in this project. We proposed a model that uses MobileNetV2 for image processing. It can classify images into 1000 categories. The captured face or video stream in real-time from the webcam will show the bounding box location of the face and detect face mask with OpenCV and Keras/Tensorflow. Once the mask is detected the input image will label 'Mask' as an output on top of the bounded box, but if the person is not wearing the mask then it will label 'No-Mask'. The system achieved an accuracy of 99% on detecting the face as 'mask' or 'no-mask'.

.

4.3 Face Recognition and Cropping

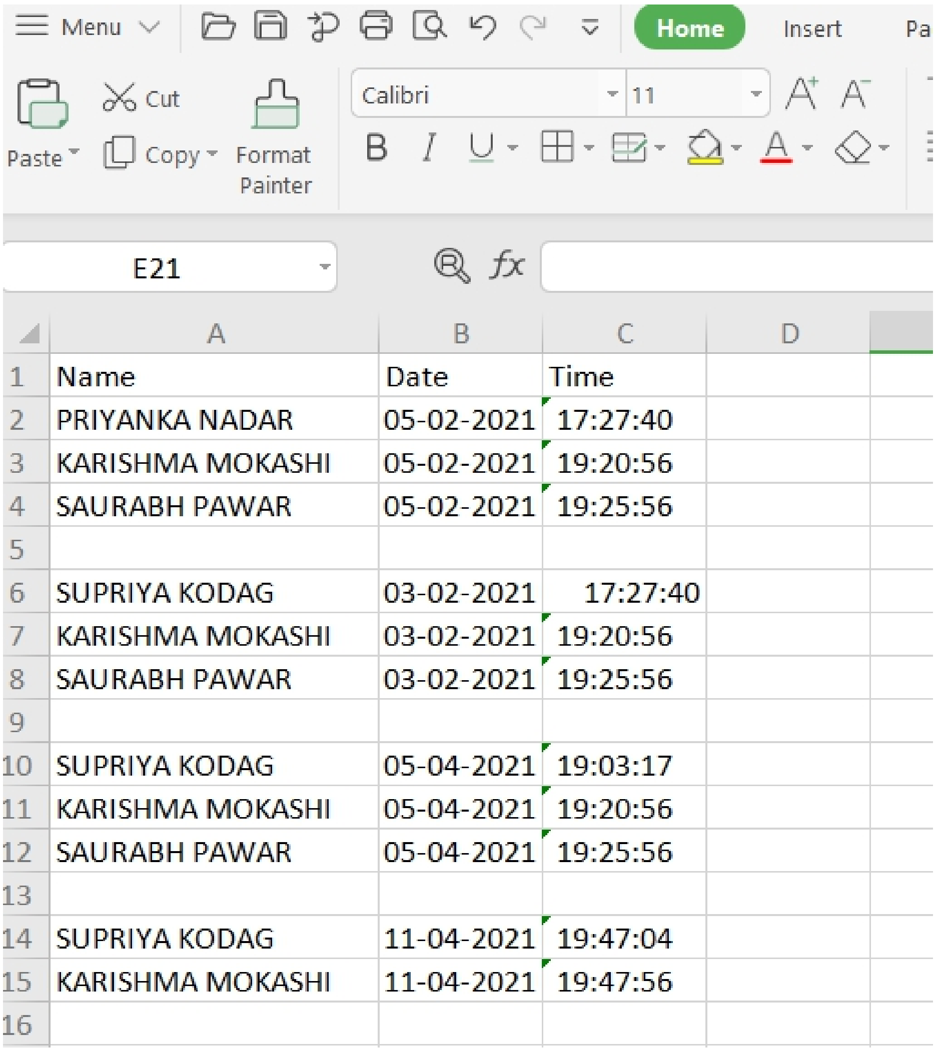
When a person wearing masks is streamed on a webcam the masked region is discarded completely from the feature extraction and classification process. A cropping filter is applied in order to obtain only the informative regions of the masked face which are eyes and forehead. Then, we apply a deep learning model which is more suitable in real-world applications compared to restoration approaches. To do this we first extract local features from training images, each feature represents a region from the image. We extract deep features using VGG-16 face CNN descriptor from images. BoF is largely used for image classification tasks. The sigmoid function is used by the Multi-layer Perceptron classifier (MLP) for representation of faces in terms of vectors. Each image in the classroom folder will be taken and its face encodings will be generated, basically, a cross-validation strategy will be used to evaluate the recognition performance. When a masked face is recognized it will display a name inside the bound box also the attendance will be recorded for the identified person. The accuracy on the test dataset is 95%, the highest to our knowledge.

4.4 Store Recognized Entries

Whenever the algorithm finds a match, the name of the person identified is stored as a .csv file with the name, timestamp and date. The attendance will not get recorded if the person is not wearing the mask; otherwise the attendance will be recorded and will display 'Attendance Marked' on the screen.

4.5 View the Attendance

The attendance sheets are often downloaded by clicking on the 'Download Attendance' button from the dashboard. The downloaded attendance sheet will be viewed as an excel file with the name, date and timestamp.



	A	B	C	D
1	Name	Date	Time	
2	PRIYANKA NADAR	05-02-2021	17:27:40	
3	KARISHMA MOKASHI	05-02-2021	19:20:56	
4	SAURABH PAWAR	05-02-2021	19:25:56	
5				
6	SUPRIYA KODAG	03-02-2021	17:27:40	
7	KARISHMA MOKASHI	03-02-2021	19:20:56	
8	SAURABH PAWAR	03-02-2021	19:25:56	
9				
10	SUPRIYA KODAG	05-04-2021	19:03:17	
11	KARISHMA MOKASHI	05-04-2021	19:20:56	
12	SAURABH PAWAR	05-04-2021	19:25:56	
13				
14	SUPRIYA KODAG	11-04-2021	19:47:04	
15	KARISHMA MOKASHI	11-04-2021	19:47:56	
16				

Fig : Sample of downloaded attendance sheet

CHAPTER 05
PROGRAM CODE

.

Program Code

Train_mask_detector.py

```
from tensorflow.keras.preprocessing.image import ImageDataGenerator
from tensorflow.keras.applications import MobileNetV2
from tensorflow.keras.layers import AveragePooling2D
from tensorflow.keras.layers import Dropout
from tensorflow.keras.layers import Flatten
from tensorflow.keras.layers import Dense
from tensorflow.keras.layers import Input
from tensorflow.keras.models import Model
from tensorflow.keras.optimizers import Adam
from tensorflow.keras.applications.mobilenet_v2 import preprocess_input
from tensorflow.keras.preprocessing.image import img_to_array
from tensorflow.keras.preprocessing.image import load_img
from tensorflow.keras.utils import to_categorical
from sklearn.preprocessing import LabelBinarizer
from sklearn.model_selection import train_test_split
from sklearn.metrics import classification_report
from imutils import paths
import matplotlib.pyplot as plt
import numpy as np
import os

# initialize the initial learning rate, number of epochs to train for,
# and batch size
INIT_LR = 1e-4
EPOCHS = 20
BS = 3
```

```
DIRECTORY = r"C:\Users\Shobha\Desktop\BE Project doc\RecognitionWithMaskAttendance\imagesAttendance"
CATEGORIES = ["mask", "no-mask"]
```

```
# grab the list of images in our dataset directory, then initialize
# the list of data (i.e., images) and class images
print("[INFO] loading images...")
```

```
data = []
labels = []
```

```
for category in CATEGORIES:
    path = os.path.join(DIRECTORY, category)
    for img in os.listdir(path):
        img_path = os.path.join(path, img)
        image = load_img(img_path, target_size=(224, 224))
        image = img_to_array(image)
        image = preprocess_input(image)
```

```
        data.append(image)
        labels.append(category)
```

```
# perform one-hot encoding on the labels
lb = LabelBinarizer()
labels = lb.fit_transform(labels)
labels = to_categorical(labels)
```

```
data = np.array(data, dtype="float32")
labels = np.array(labels)
```

```
(trainX, testX, trainY, testY) = train_test_split(data, labels,
    test_size=0.20, stratify=labels, random_state=42)
```

```
# construct the training image generator for data augmentation
aug = ImageDataGenerator(
    rotation_range=20,
    zoom_range=0.15,
    width_shift_range=0.2,
    height_shift_range=0.2,
    shear_range=0.15,
    horizontal_flip=True,
    fill_mode="nearest")
```

```
# load the MobileNetV2 network, ensuring the head FC layer sets are
# left off
baseModel = MobileNetV2(weights="imagenet", include_top=False,
    input_tensor=Input(shape=(224, 224, 3)))
```

```
# construct the head of the model that will be placed on top of the
# the base model
headModel = baseModel.output
headModel = AveragePooling2D(pool_size=(7, 7))(headModel)
headModel = Flatten(name="flatten")(headModel)
headModel = Dense(128, activation="relu")(headModel)
headModel = Dropout(0.5)(headModel)
headModel = Dense(2, activation="softmax")(headModel)
```

```
# place the head FC model on top of the base model (this will become
# the actual model we will train)
model = Model(inputs=baseModel.input, outputs=headModel)
```

```
# loop over all layers in the base model and freeze them so they will
# *not* be updated during the first training process
for layer in baseModel.layers:
    layer.trainable = False
```

```
# compile our model
print("[INFO] compiling model...")
opt = Adam(lr=INIT_LR, decay=INIT_LR / EPOCHS)
model.compile(loss="binary_crossentropy", optimizer=opt,
    metrics=["accuracy"])
```

```
# train the head of the network
print("[INFO] training head...")
H = model.fit(
    aug.flow(trainX, trainY, batch_size=BS),
    steps_per_epoch=len(trainX) // BS,
    validation_data=(testX, testY),
    validation_steps=len(testX) // BS,
    epochs=EPOCHS)
```

```
# make predictions on the testing set
print("[INFO] evaluating network...")
predIdxs = model.predict(testX, batch_size=BS)
```

```
# for each image in the testing set we need to find the index of the
# label with corresponding largest predicted probability
predIdxs = np.argmax(predIdxs, axis=1)
```

```
# show a nicely formatted classification report
print(classification_report(testY.argmax(axis=1), predIdxs,
    target_names=lb.classes_))
```

```
# serialize the model to disk
print("[INFO] saving mask detector model...")
```


.

```
model.save("mask_detector2.model", save_format="h5")
```

```
# plot the training loss and accuracy
N = EPOCHS
plt.style.use("ggplot")
plt.figure()
# plt.plot(np.arange(0, N), H.history["loss"], label="train_loss")
plt.plot(np.arange(0, N), H.history["val_loss"], label="val_loss")
plt.plot(np.arange(0, N), H.history["accuracy"], label="train_acc")
plt.plot(np.arange(0, N), H.history["val_accuracy"], label="val_acc")
plt.title("Training Loss and Accuracy")
plt.xlabel("Epoch #")
plt.ylabel("Loss/Accuracy")
plt.legend(loc="lower left")
plt.savefig("plot1.png")
```


.

CHAPTER 06

RESULT

.

RESULT

We have built the face recognition attendance system for masked faces. The dataset contained 800 images which have a single and frontal face. These 800 images comprise 400 unique unmasked images and 400 masked images. Once the system is trained with this dataset and tested with unmasked faces, its accuracy was 95%. The system performance and accuracy when the person is without a mask increased by 99%. Thus, high-quality frontal face images are readily acquired, in order that the masked face recognition task is not any longer so difficult. Moreover, dealing with only the informative regions (unmasked ones) and the high generalization method proposed makes it applicable in real-time applications.

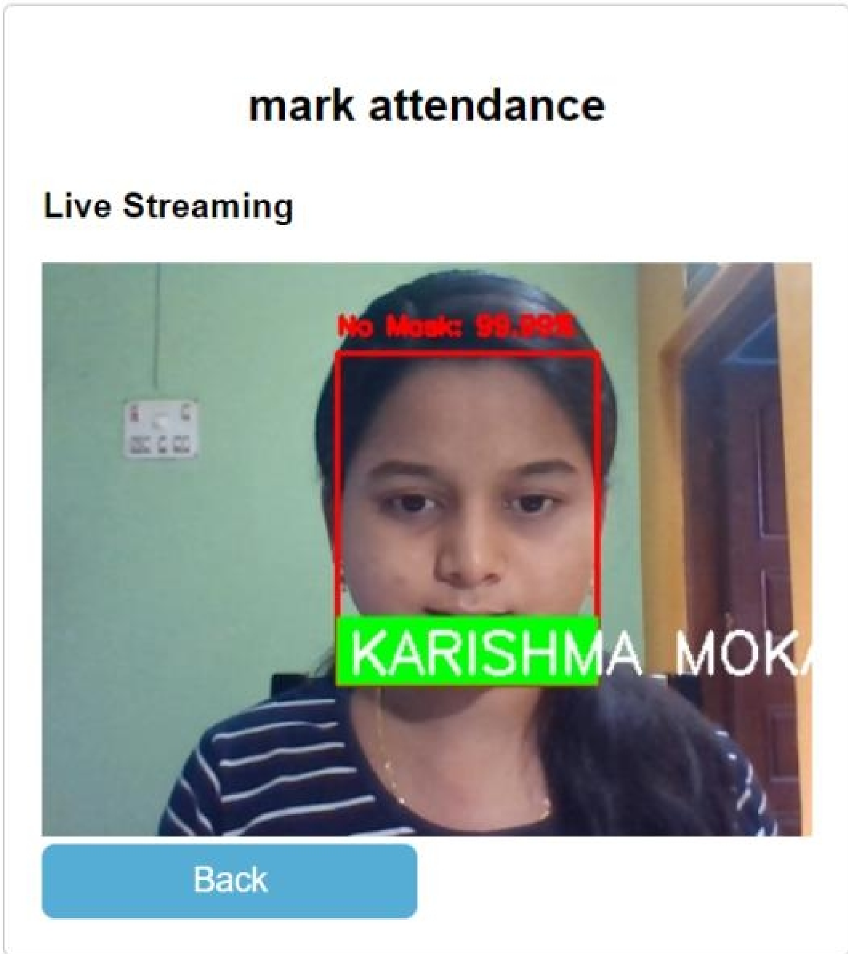


Fig : Result on a real-time image with no mask

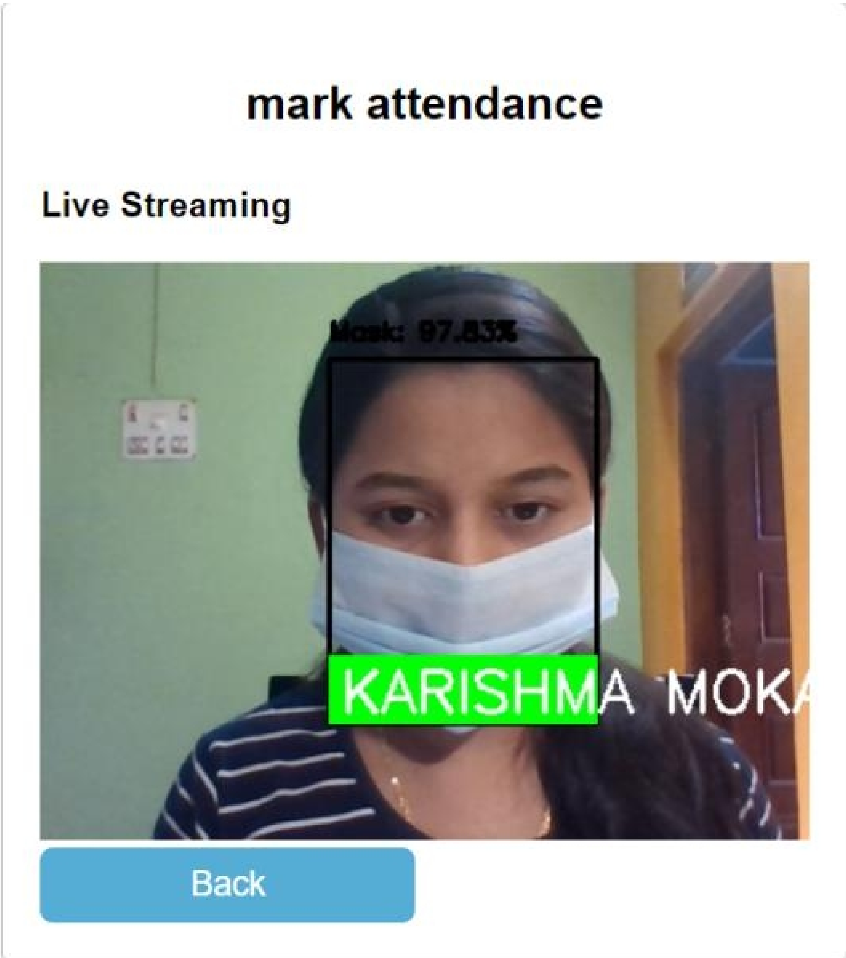


Fig : Result on a real-time image with a mask

CHAPTER 07

CONCLUSION

CONCLUSION

During this pandemic situation existing face recognition technology will easily fail to make efficient recognition. The proposed system improves the face recognition attendance process in the presence of masks using deep learning-based methods. Moreover this is not limited to this pandemic as people wear masks to take care of their health and protect themselves against pollution. In the proposed system, mask detection technology can be used to identify faces in the crowd wearing masks or not. In future, notification to the student can be sent once the attendance is marked.

.

CHAPTER 08

REFERENCE

.

REFERENCE

[1] Dataset:_https://www.kaggle.com/prasoonkottarathil/face-mask-lite-dataset

[2] Lingxiao He, Haiqing Li, Qi Zhang, and Zhenan Sun. Dynamic feature matching for partial face recognition. IEEE Transactions on Image Processing, 28(2):791–802, 2018.

[3] Lingxue Song, Dihong Gong, Zhifeng Li, Changsong Liu, and Wei Liu. Occlusion robust face recognition based on mask learning with pairwise differential siamese network. In Proceedings of the IEEE International Conference on Computer Vision, pages 773–782, 2019.

[4] Zhongyuan Wang, Guangcheng Wang, Baojin Huang, Zhangyang Xiong, Qi Hong, Hao Wu, Peng Yi, Kui Jiang, Nanxi Wang, Yingjiao Pei, et al. Masked face recognition dataset and application. arXiv preprint arXiv:2003.09093, 2020.

[5] Nizam Ud Din, Kamran Javed, Seho Bae, and Juneho Yi. A novel gan-based network for unmasking of masked face. IEEE Access, 8:44276–44287, 2020.

.