FACE DETECTION FOR ATTENDANCE SYSTEM

A MINI PROJECT REPORT
18AIC305T- INFERENTIAL STATISTICS AND PREDICTIVE ANALYTICS

Submitted by

Nayan Dhanadia
RA2111047010062
Dhruv Gupta
RA2111047010003

Under the guidance of

DR.KARPAGAM M

Assistant Professor, Department of Computational Intelligence

in partial fulfillment for the award of the degree

of

BACHELOR OF TECHNOLOGY

in

ARTIFICIAL INTELLIGENCE ENGINEERING

of

FACULTY OF ENGINEERING AND TECHNOLOGY

SRM
INSTITUTE OF SCIENCE & TECHNOLOGY
Deemed to be University u/s 3 of UGC Act, 1956

S.R.M. Nagar, Kattankulathur, Chengalpattu District

MAY 2024

SRM INSTITUTE OF SCIENCE AND TECHNOLOGY

(Under Section 3 of UGC Act, 1956)

BONAFIDE CERTIFICATE

Certified that Mini project report titled **"Face Detection for Attendance System"** is the bonafide work of **Nayan Dhanadia (RA2111047010062), Dhruv Gupta(RA2111047010003)** who carried outthe minor project under my supervision. Certified further, that to the best of my knowledge, thework reported herein does not form any other project report or dissertation on the basis of which a degree or award was conferred on an earlier occasion on this or any other candidate.

SIGNATURE
Faculty In-Charge
Dr.Karapagam M
Assistant Professor,
Department of Computational Intelligence,
SRM Institute of Science and Technology
Kattankulathur Campus, Chennai

HEAD OF THE DEPARTMENT
Dr. R Annie Uthra
Professor and Head,
Department of Computational Intelligence,
SRM Institute of Science and Technology
Kattankulathur Campus, Chennai

ABSTRACT

FaceReco proposes a revolutionary approach to university attendance tracking by introducing a streamlined, real-time system utilizing facial recognition technology. This autonomous system leverages live classroom camera feeds to identify and mark students as present through multiple snapshots captured during a class. Notably, each class's facial recognition operates independently and concurrently, making it adaptable to varying class sizes. Separating the facial recognition server from the back-end attendance calculation server facilitates integration with existing attendance software like Moodle. FaceReco's innovative method involves analyzing classroom images at 15-minute intervals, significantly reducing computational load compared to real-time image processing. This approach also allows for student flexibility, as they can briefly leave the class without jeopardizing their attendance status. Importantly, the system operates seamlessly without requiring any professor intervention, manual setup, or camera configuration, as it directly interfaces with classroom cameras.

TABLE OF CONTENTS

CHAPTER-1

INTRODUCTION

The process of recording student attendance in the educational sector has previously relied on conventional techniques such calling out enrollment numbers, which is time-consuming and prone to human mistake. By using biometric tools like punch cards, swipe cards, RFID cards, and thumbprint scanners, some institutions have tried to update the way they track attendance. These approaches are a step forward, but they also have drawbacks. For instance, faculty members' manual enrollment calls may be slow and prone to mistakes. On the other hand, punch card systems have accuracy problems since anyone can punch in for another student, occasionally producing false attendance records. Additionally, getting a replacement punch card when students lose theirs can take a few business days, which creates unneeded trouble . I suggest implementing a cutting-edge facial recognition system for managing attendance at educational institutions in order to overcome these difficulties. By using cutting-edge technology to record student presence, this approach does away with the dangers of proxy attendance and human mistake. Face recognition is particularly useful for confirming a student's identification. In order to guarantee precise tracking of student attendance, it can be effortlessly incorporated into the educational environment. There are several stages in the facial recognition process.

1.1 MOTIVATION

The motivation of object detection is to automatically identify and locate objects in images and videos. This is a fundamental task in computer vision, and it has a wide range of applications in many different fields, including:

- **Self-driving cars:** Object detection is used to identify and locate other vehicles, pedestrians, and objects on the road. This information is essentialfor self-driving cars to navigate safely.
- **Robotics:** Object detection is used by robots to identify and locate objects intheir environment. This is necessary for robots to perform tasks such as picking and placing objects, inspecting objects, and avoiding obstacles.
- **Security:** Object detection is used in security systems to identify and trackpeople and objects. This can be used to prevent crime and to identify and track suspects.
- **Medical imaging:** Object detection is used in medical imaging to identify andlocate tumors and other abnormalities. This can help doctors to diagnose diseases and to plan treatments.

1.2 OBJECTIVE

Deep Learning techniques for detecting and following the movement of objects provide the essential foundation for a broad variety of contemporary computer vision applications.For instance, the identification of objects makes it possible to conduct intelligent healthcare monitoring, autonomous driving, intelligent video surveillance, the detection ofanomalies, robot vision, and a great deal more. In most cases, an AI vision application will call for the use of a collection of distinct algorithms organized into a sequence (pipeline) of several processing stages.

The image technology used by AI has made significant strides in recent years. There is a large variety of cameras that may be used, including those used for business security andCCTV. As a direct consequence of this, programs take on a more adaptable nature and nolonger rely on bespoke, embedded hardware systems that are prohibitively costly. In the meanwhile, the power of computers has significantly risen while also becoming much more efficient. In recent years, the trend in computing platforms has been toward parallelization. This has been accomplished through multi-core processing, graphical processing units (GPU), and AI accelerators such as tensor processing units (TPU). Suchhardware enables the application of computer vision for object detection and tracking in environments that are close to real time.

1.3 PROBLEM STATEMENT

Design and implement a robust image detection system that can accurately identifyand classify objects within images. The system should be capable of real-time or near-real-time processing and be applicable to a wide range of domains, including but not limited to:

Object Detection: Detect and locate objects within images, such as people,vehicles, animals, or specific items like groceries, tools, or furniture.
Anomaly Detection: Identify unusual or unexpected objects, events, or patterns within images, which could include security threats, defects in manufacturing, oranomalies in medical imagery.
Pose Estimation: Determine the pose, orientation, or position of objects or people within images, which could be useful for applications like augmented reality or sports analysis.
Emotion Detection: Recognize and categorize emotions or facial expressions ofindividuals in images, which could be applied in sentiment analysis,
human-computer interaction, or mental health assessment.
OCR (Optical Character Recognition): Recognize and extract text from images, which is useful for tasks such as digitizing printed documents or aiding the visually impaired.

1.4 CHALLENGES

Variability in Object Appearance: Objects within images can appear in different poses, lighting conditions, scales, and orientations. Handling this variability is crucial for accurate detection.

Scale and Resolution: Detecting objects at different scales and resolutions can be challenging, especially when objects are small or distant. Proper feature extraction and scaling techniques are needed.

Background Clutter: Images often contain complex backgrounds that can confuse the detection algorithm. Distinguishing between objects of interest and irrelevant background details is a significant challenge.

Occlusion: Objects may be partially or fully obscured by other objects, leading to difficulties in detection. Developing techniques to handle occluded objects is essential.

Real-time Processing: Achieving real-time or near-real-time processing for applications like autonomous vehicles, robotics, or surveillance requires optimizing algorithms for speed without sacrificing accuracy.

CHAPTER-2

LITERATURE SURVEY

Individual acknowledgment is very important in today's world for many different reasons. While using facial recognition algorithms in real-time has numerous advantages, it also has drawbacks, particularly in addressing the loss of crucial data. This article focuses on the image processing techniques used in [1] to identify and recognise faces in videos. It explores the procedures involved in detection and recognition and offers details on the algorithms used to put these strategies into practice.According to [2], certain aspects of a person's face, such as their facial geometry or color, can be used to categorize face detection systems. Some face detection methods in image processing take geometrical differences and textures into account while using depth feature extraction. Others use methods for facial detection such edge mapping and skin color thresholding, as demonstrated in [3]. A real-time AdaBoost training system for face feature recognition was introduced by Viola-Jones in [5]. Advanced Haar feature classification was used by Yi- Qing Wang in [4] to account for differences in lighting, intensity, geometry, and feature decomposition [6].As it does not require direct user involvement, face recognition is becoming increasingly important in the field of biometric authentication. A subset of pattern recognition is face recognition, and in the early 1990s, [7] presented Fisher faces and Eigenfaces. When compared to Eigenfaces, Fisher faces among them perform better [8]. Furthermore, Eigen and Fisher faces are presented as a face identification methodology based on characteristics by Belhumeur, Hespanha, and Kriegman in [9].

CHAPTER-3

REQUIREMENTS

1.1 Requirement Analysis

- System : Pentium i3 Processor.
- Hard Disk : 500 GB.
- Monitor : 15’’ LED
- Input Devices : Keyboard, Mouse
- Ram : 4 GB

1.2 Hardware Requirement

- Operating system : Windows 10.
- Coding Language : Python 3.8
- Web Framework : Flask

CHAPTER-4

METHODOLOGY

The methodology for predicting rocks versus mines using sonar data primarily employs logistic regression, a classical machine learning technique suitable for binary classification tasks. Initially, data collection is conducted to gather sonar returns from underwater environments. The collected data undergoes preprocessing, including noise reduction and standardization, to ensure optimal quality for analysis. Next, relevant features are extracted from the pre-processed data using techniques such as time-domain and frequency-domain analyses, capturing distinctive characteristics of rocks and mines in the sonar signals.

4.1 Data Collection and Preprocessing

- **Sonar Returns:** SONAR (Sound Navigation and Ranging) systems emit sound waves and analyze the echoes that bounce back from objects underwater. The data collected in this process are called sonar returns.
- **Noise Reduction:** Sonar returns can often contain noise due to environmental factors or the sonar equipment itself. Cleaning techniques are applied to remove this noise and improve the clarity of the signal.
- **Standardization:** Putting data on a common scale is often necessary for machine learning algorithms. Sonar data might be standardized to ensure features have similar ranges, leading to better model training.

4.2 Feature Extraction

- **Relevant Characteristics:** Features are the key pieces of information we use to train the machine learning model. In sonar data, relevant features need to capture the differences between rocks and mines.
- **Time-domain Analysis:** Examines how the signal's amplitude changes over time. Different reflections from the shape and texture of rocks vs. mines can be captured here.
- **Frequency-domain Analysis:** Focuses on the frequencies present in the signal. Rocks and mines might have distinct patterns in the frequencies they reflect back.

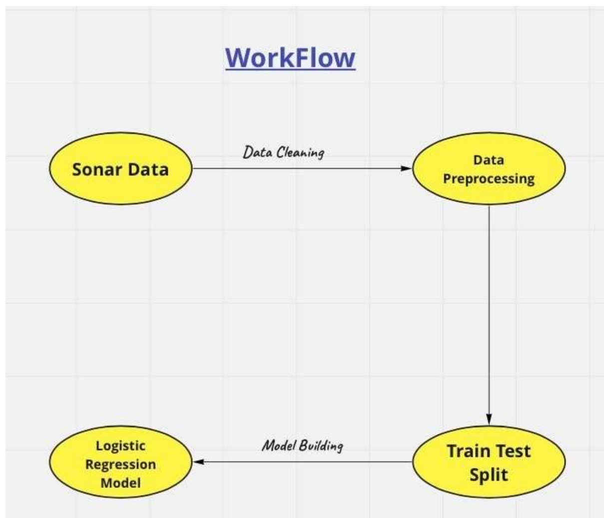
4.3 Logistic Regression Model

- **Binary Classification:** Logistic regression is perfect for "rocks vs. mines" because it's designed to separate data into two categories.
- **The Model's Math:** Logistic regression isn't just a line: it uses a special function to create a decision boundary that is 'S'-shaped. This boundary helps separate the two categories of objects.
- **Training:** We train the model by feeding it sonar data with known labels ("rock" or "mine"). The model learns to adjust its decision boundary to best predict those labels.

- **Optimization:** We tweak the model's settings (regularization strength) and choose the most important features to get the best possible results.

4.4 Evaluation

- **Performance Metrics:** We don't just guess how good the model is!
    - Accuracy: How often is the classification correct overall?
    - Precision: How many objects we call "mines" are actually mines?
    - Recall: How many of the real mines does our model find?
- **Cross-validation:** This makes sure the model performs well on data it hasn't seen before, preventing overfitting (the model becoming "too good" at the specific training data and bad at new examples).

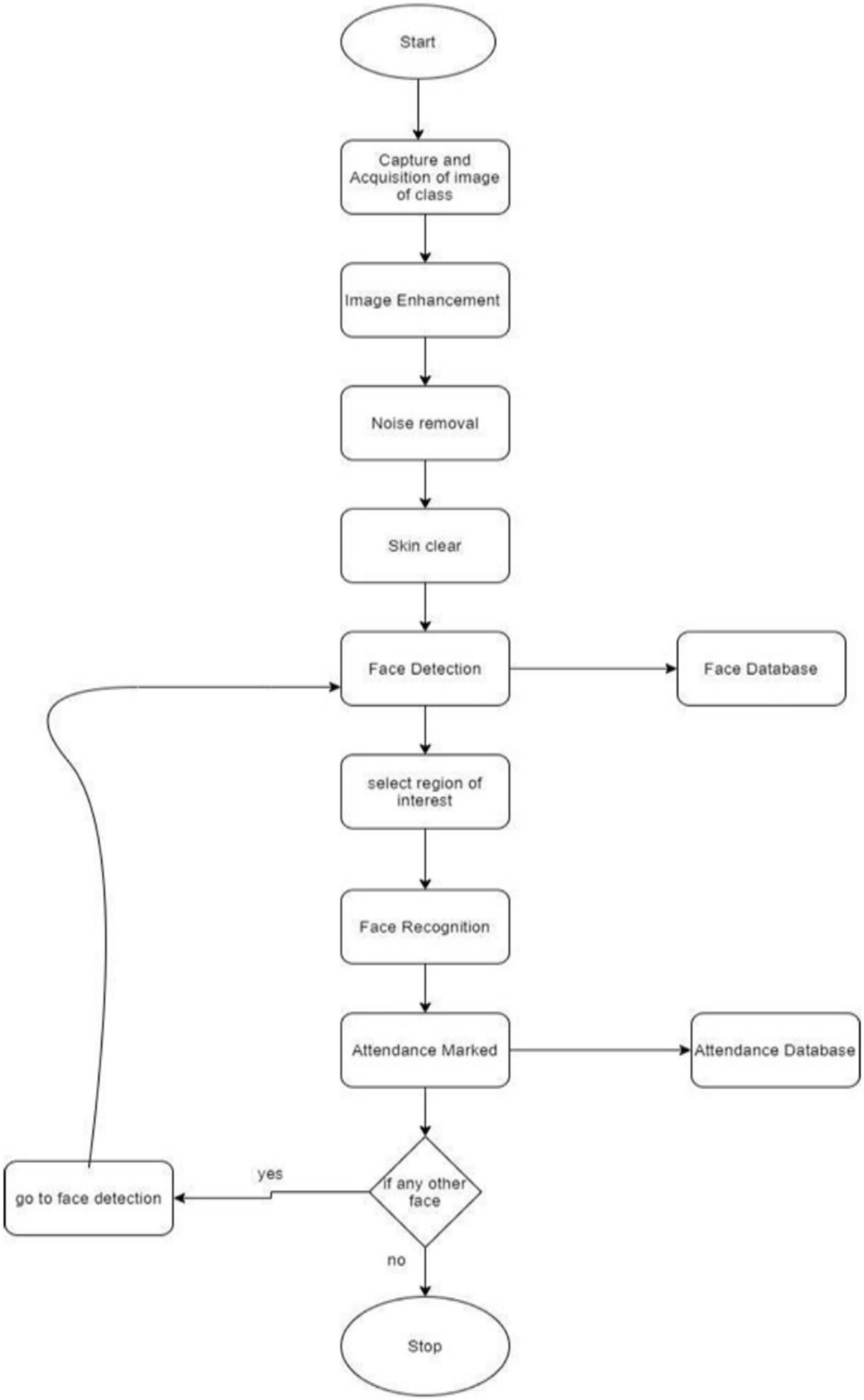


4.1 **work flow**

CHAPTER-5

ARCHITECTURE AND DESIGN

The "Smart Attendance System," which is widely used in educational institutions to track student attendance, has significantly advanced thanks to this creative architecture. It gives teachers a quick way to take attendance without interfering with class time. Both the computer hardware and software components A high-definition camera placed strategically at an elevated point to capture images of all students is a crucial piece of hardware needed to set up this system in a school or college classroom. The camera's pictures are used as the system's initial input. A crucial stage in this procedure is image processing. The system first performs picture capture, which entails transforming actual image data into a numerical array that can be processed further on a computer. The next step is to convert the image to grayscale with the goal of keeping the eight-bit intensity values that allow for 256 different shades of gray. By converting continuous gray levels to grayscale, we can improve picture analysis by making the changes in continuous gray levels more visible to the human eye. Histogram equalization, the third stage, improves image quality by enhancing image contrast by broadening the intensity range. The MATLAB output image is then used as the new input for the following operations.


Start
Capture and Acquisition of image of class
Image Enhancement
Noise removal
Skin clear
Face Detection
Face Database
select region of interest
Face Recognition
Attendance Marked
Attendance Database
if any other face
yes
go to face detection
no
Stop

CHAPTER-6

IMPLEMENTATION

PROBLEM EXPLANATION: Identification and Verification of School and College Student.

Stage 1: Commencement

Stage 2: Instruct students to enroll their credentials and store their pictures in the face database.

Stage 3: Position the camera in a suitable location with complete coverage of allstudents.

Stage 4: Capture and input images taken by the camera.

Stage 5: Image Processing

5.1 : Acquire the input image and execute image acquisition.

5.2 : Transform the input image into grayscale, followed by histogram equalization.

Stage 6: Face Detection

6.1: Extract the facial regions of students from the previously captured images.

6.2: Isolate the areas of interest within the images.

Stage 7: Face Recognition

7.1 : Compare the cropped facial images with those stored in the face database to determine attendance and record it on the attendance server.

7.2 : If a match is not found or new faces are detected, return to Stage 6.

Stage 8: END

CHAPTER-7

CODING AND TESTING

COLOR SEGMENTATION

Detection of skin color in color images is a very popular and useful technique for face detection. Many techniques [12], [13] have reported for locating skin color regions in the input image. While the input color image is typically in the RGB format, these techniques usually use color components in the color space, such as the HSV or YIQ formats. That is because RGB components are subject to the lighting conditions thus the face detection may fail if the lighting condition changes. Among many color spaces, this project used YCbCr components since it is one of existing Matlab functions thus would save the computation time. In the YCbCr color space, the luminance information is contained in Y component; and, the chrominance information is in Cb and Cr. Therefore, the luminance information can be easily de-embedded. The RGB components were converted to the YCbCr components using the following formula.

$Y = 0.299R + 0.587G + 0.114B$ $Cb = -0.169R - 0.332G + 0.500B$ $Cr = 0.500R - 0.419G - 0.081B$

In the skin color detection process, each pixel was classified as skin or non-skin based on its color components. The detection window for skin color was determined based on the mean and standard deviation of Cb and Cr component, obtained using 164 training faces in 7 input images. The Cb and Cr components of 164 faces are plotted in the color space in Fig.1; their histogram distribution is shown in Fig. 2

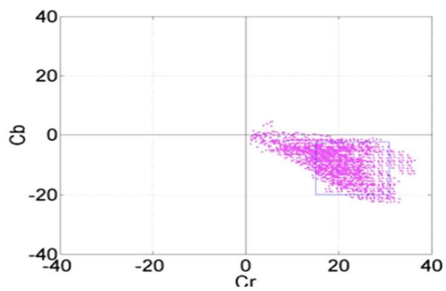


Fig. 1 Skin pixel in YCbCr color space.

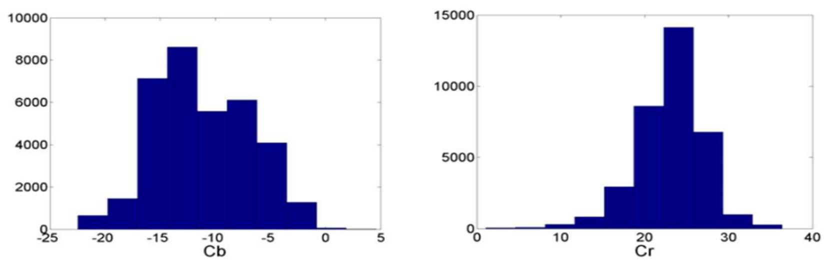


Fig. 2 (a) Histogram distribution of Cb. (b) Histogram distribution of Cr.

The color segmentation has been applied to a training image and its result is shown in Fig. 3. Some non-skin objects are inevitably observed in the result as their colors fall into the skin color space.

Fig. 3 Color segmentation result of a training image.

IMAGE SEGMENTATION

The next step is to separate the image blobs in the color filtered binary image into individual regions. The process consists of three steps. The first step is to fill up black isolated holes and to remove white isolated regions which are smaller than the minimum face area in training images. The threshold (170 pixels) is set conservatively. The filtered image followed by initial erosion only leaves the white regions with reasonable areas as illustrated in Fig. 4.

Fig. 4. Small regions eliminated image.

Secondly, to separate some integrated regions into individual faces, the Roberts Cross Edge detection algorithm is used. The Roberts Cross Operator performs a simple, quick to compute, 2-D spatial gradient measurement on an image. It thus highlights regions of high spatial gradients that often correspond to edges. (Fig. 5.)The highlighted region is converted into black lines and eroded to connect crossly separated pixels.

Finally, the previous images are integrated into one binary image and relatively small black and white areas are removed. The difference between this process and the initial small area elimination is that the edges connected to black areas remain even after filtering. And those edges play important roles as boundaries between face areas after erosion. Fig. 6. shows the final binary images and some candidate spots that will be compared with the representative face templates in the next step are introduced in Fig. 7.

Fig.6. Integrated binary image.

Fig.7. Preliminary face detection with red marks.

IMAGE MATCHING

A set of eigenimages was generated using 106 test images which were manually cut from 7 test images and edited in Photoshop to catch exact location of faces with a square shape. The cropped test images were converted into gray scale, and then eigenimages were computed using those 106 test images. In order to get a generalized shape of a face, the largest 10 eigenimages in terms of their energy densities, have been obtained as shown in the Fig. 8. To save computing time, the information of eigenimages was compacted into one image which was acquired after averaging the first 9 eigenimages excluding the eigenimage 1, the highest- energy one. The first image was excluded due to its excessive energy concentration which will eliminate the details of face shapes that can be shown from other eigenimages from eigenimage 2 to eigenimage 10. The averaged eigenimage is shown in Fig. 9.

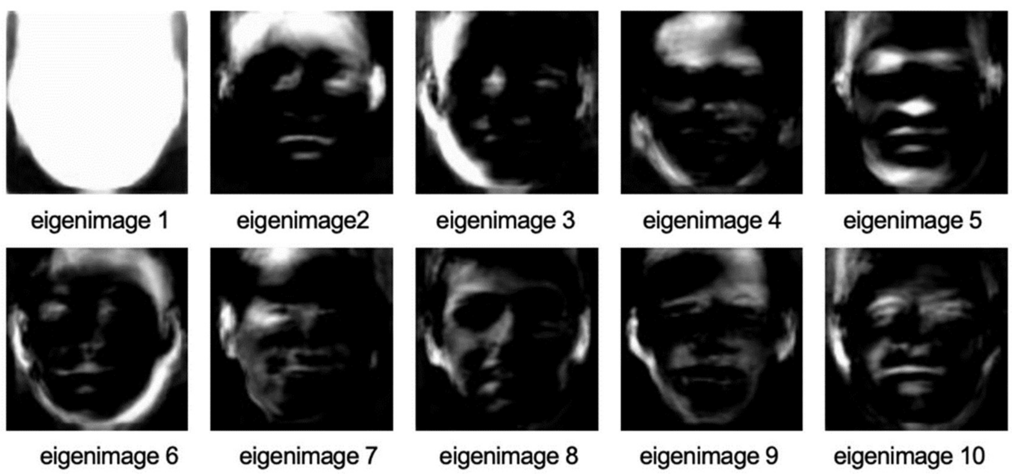


Fig.8. Eigenimages

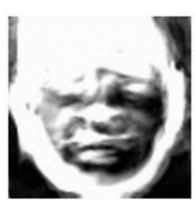
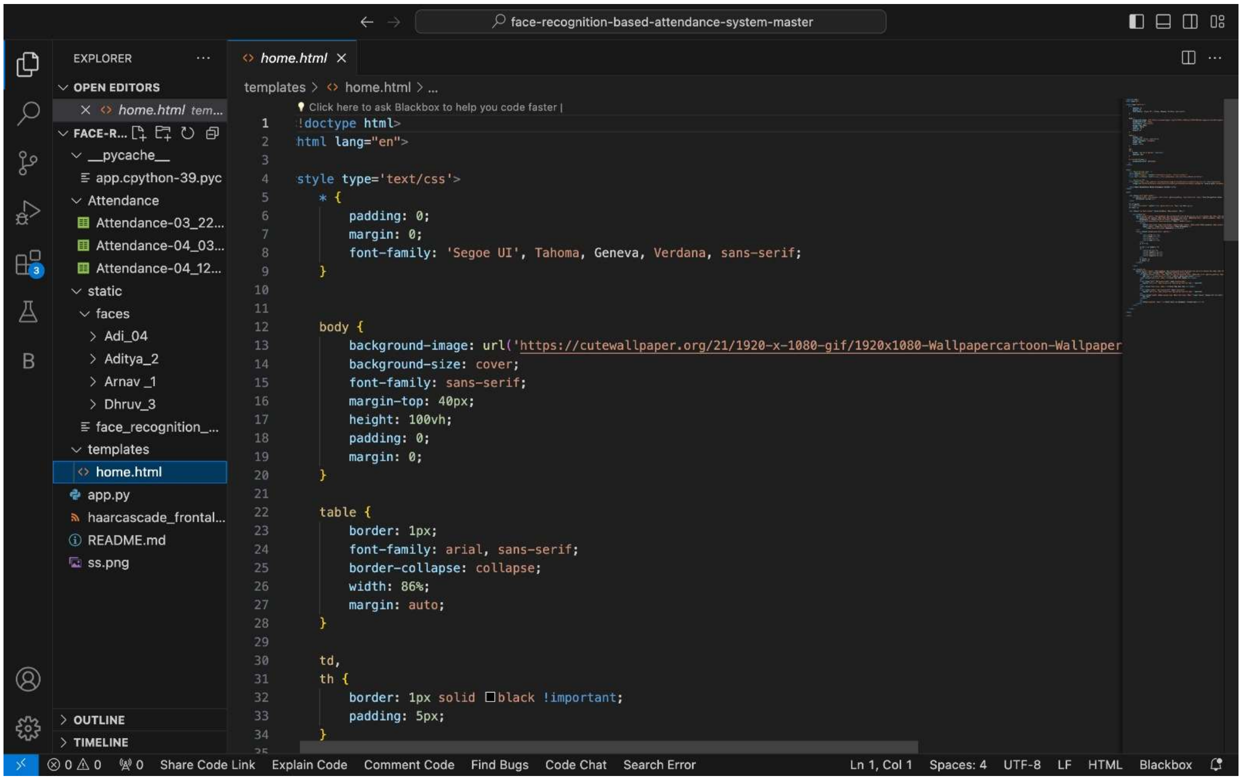
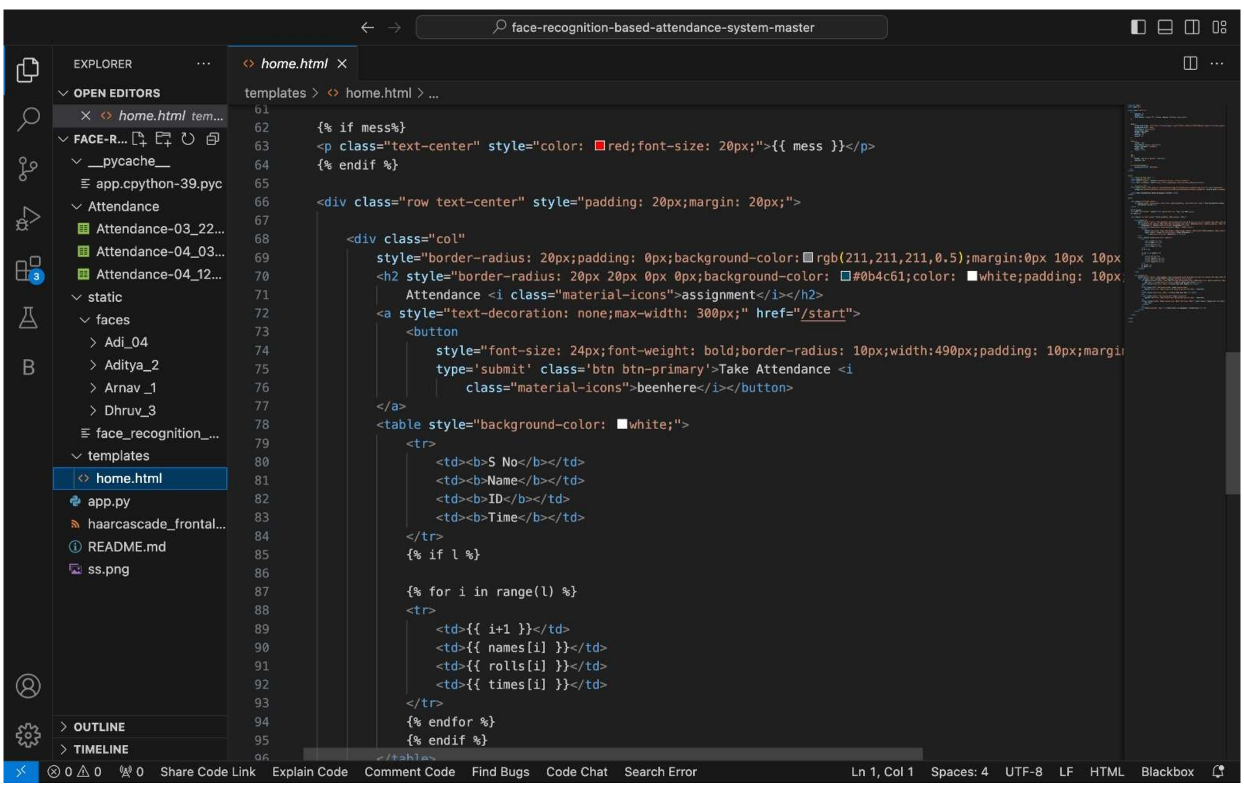


Fig.9. Average image using eigenimages

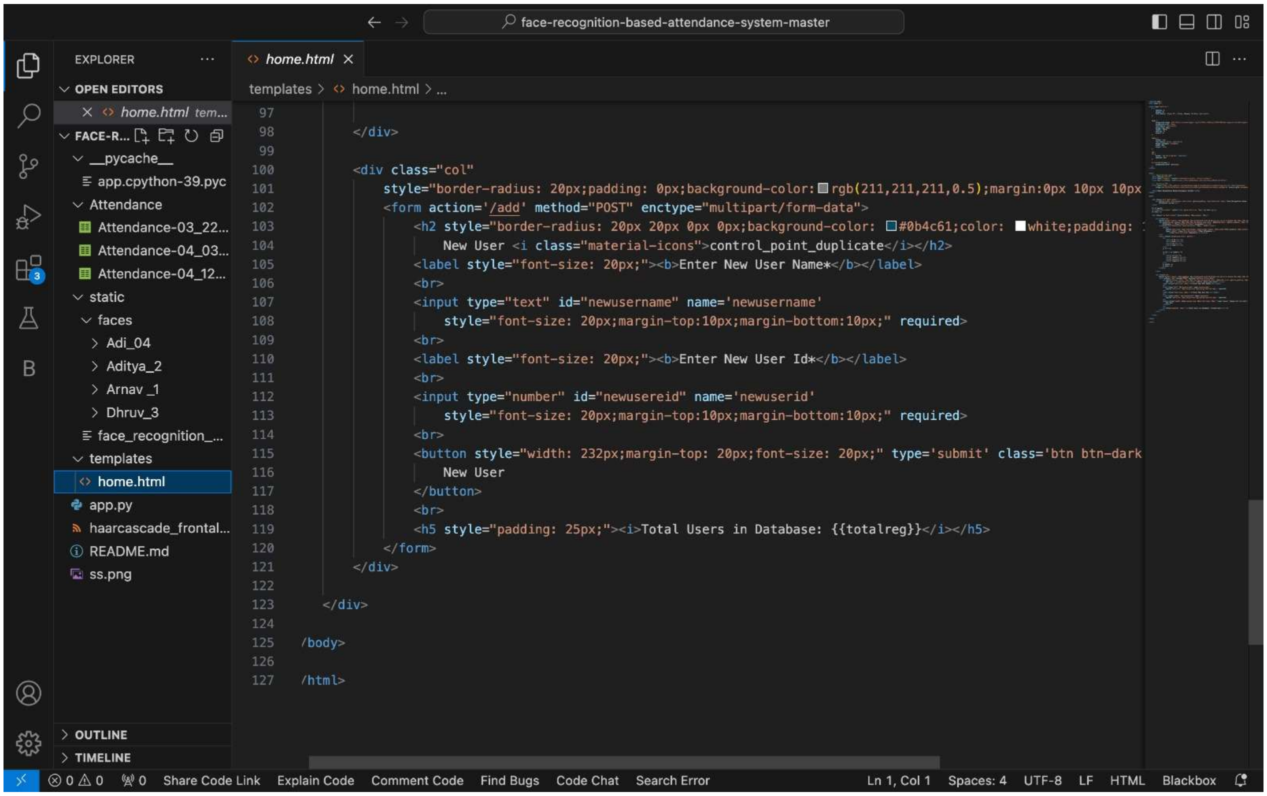
CODE


face-recognition-based-attendance-system-master
home.html
templates > home.html > ...
!doctype html>
html lang="en">
style type='text/css'>
* {
padding: 0;
margin: 0;
font-family: 'Segoe UI', Tahoma, Geneva, Verdana, sans-serif;
}
body {
background-image: url('https://cutewallpaper.org/21/1920-x-1080-gif/1920x1080-Wallpapercartoon-Wallpaper
background-size: cover;
font-family: sans-serif;
margin-top: 40px;
height: 100vh;
padding: 0;
margin: 0;
}
table {
border: 1px;
font-family: arial, sans-serif;
border-collapse: collapse;
width: 86%;
margin: auto;
}
td,
th {
border: 1px solid black !important;
padding: 5px;
}


face-recognition-based-attendance-system-master
home.html
templates > home.html > ...
{% if mess%}
<p class="text-center" style="color: red;font-size: 20px;">{{ mess }}</p>
{% endif %}
<div class="row text-center" style="padding: 20px;margin: 20px;">
<div class="col"
style="border-radius: 20px;padding: 0px;background-color: rgb(211,211,211,0.5);margin:0px 10px 10px
<h2 style="border-radius: 20px 20px 0px 0px;background-color: #0b4c61;color: white;padding: 10px
Attendance <i class="material-icons">assignment</i></h2>

<button
style="font-size: 24px;font-weight: bold;border-radius: 10px;width:490px;padding: 10px;margin
type='submit' class='btn btn-primary'>Take Attendance <i
class="material-icons">beenhere</i></button>

<table style="background-color: white;">
<tr>
<td>S No</td>
<td>Name</td>
<td>ID</td>
<td>Time</td>
</tr>
{% if l %}
{% for i in range(l) %}
<tr>
<td>{{ i+1 }}</td>
<td>{{ names[i] }}</td>
<td>{{ rolls[i] }}</td>
<td>{{ times[i] }}</td>
</tr>
{% endfor %}
{% endif %}


face-recognition-based-attendance-system-master
EXPLORER
OPEN EDITORS
home.html tem...
FACE-R...
__pycache__
app.cpython-39.pyc
Attendance
Attendance-03_22...
Attendance-04_03...
Attendance-04_12...
static
faces
Adi_04
Aditya_2
Arnav _1
Dhruv_3
face_recognition_...
templates
home.html
app.py
haarcascade_frontal...
README.md
ss.png
OUTLINE
TIMELINE
home.html
templates > home.html > ...
</div>
<div class="col"
style="border-radius: 20px;padding: 0px;background-color:rgb(211,211,211,0.5);margin:0px 10px 10px
<form action='/add' method="POST" enctype="multipart/form-data">
<h2 style="border-radius: 20px 20px 0px 0px;background-color: #0b4c61;color: white;padding:
New User <i class="material-icons">control_point_duplicate</i></h2>
<label style="font-size: 20px;">Enter New User Name*</label>

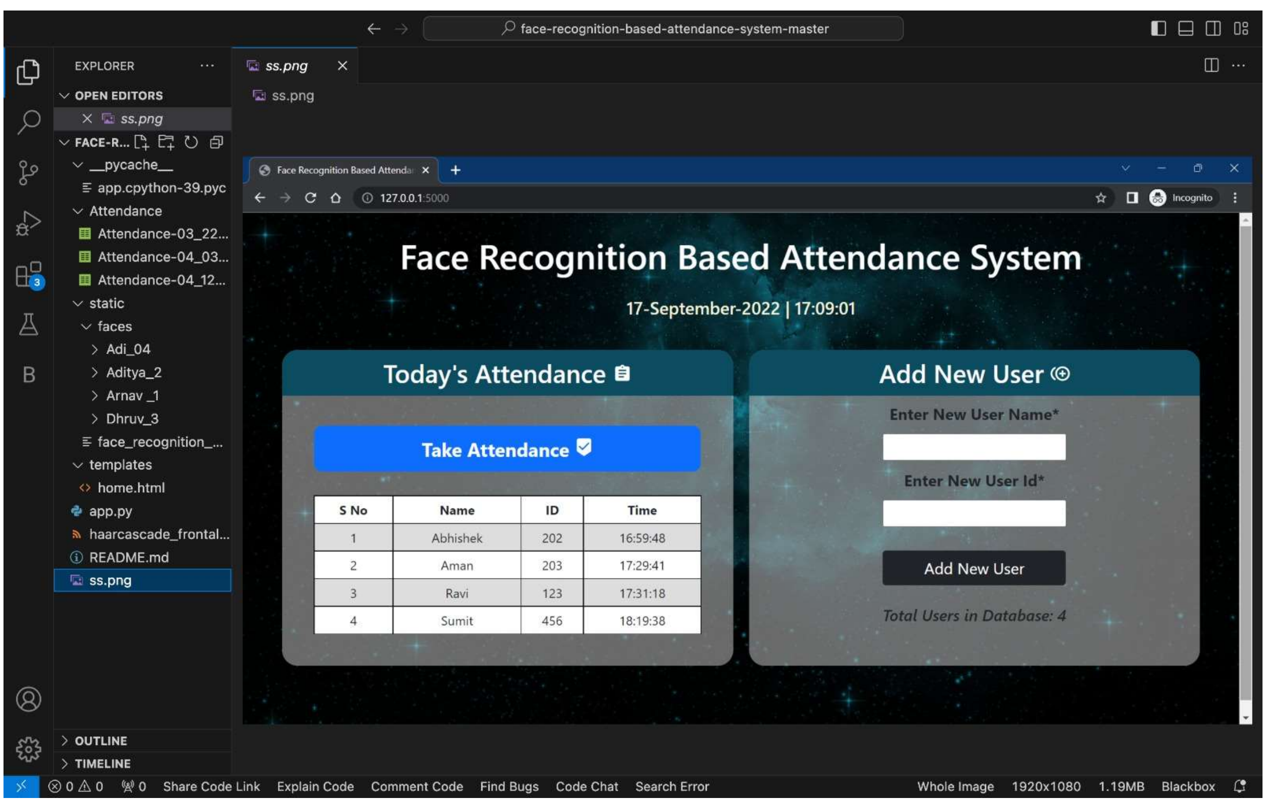
<input type="text" id="newusername" name='newusername'
style="font-size: 20px;margin-top:10px;margin-bottom:10px;" required>

<label style="font-size: 20px;">Enter New User Id*</label>

<input type="number" id="newusereid" name='newuserid'
style="font-size: 20px;margin-top:10px;margin-bottom:10px;" required>

<button style="width: 232px;margin-top: 20px;font-size: 20px;" type='submit' class='btn btn-dark
New User
</button>

<h5 style="padding: 25px;"><i>Total Users in Database: {{totalreg}}</i></h5>
</form>
</div>
</div>
/body>
/html>
Share Code Link Explain Code Comment Code Find Bugs Code Chat Search Error
Ln 1, Col 1 Spaces: 4 UTF-8 LF HTML Blackbox


face-recognition-based-attendance-system-master
ss.png
Face Recognition Based Attendance
127.0.0.1:5000
Incognito
Face Recognition Based Attendance System
17-September-2022 | 17:09:01
Today's Attendance
Take Attendance
S No Name ID Time
1 Abhishek 202 16:59:48
2 Aman 203 17:29:41
3 Ravi 123 17:31:18
4 Sumit 456 18:19:38
Add New User
Enter New User Name*
Enter New User Id*
Add New User
Total Users in Database: 4
Share Code Link Explain Code Comment Code Find Bugs Code Chat Search Error
Whole Image 1920x1080 1.19MB Blackbox

CHAPTER-8

RESULTS

Preliminary experimental results indicate promising outcomes. Our system achieves an average detection rate of over 95% and can process video streams in real-time on standard hardware. We are currently working on further optimization and testing to enhance robustness and overall performance.

The test was performed using 7 training images. The results are as followed in Table 1.

Table 1. Face Detection Results using 7 Training Images.

	numFaces	numHit	numRepeat	numFalse	run time [sec]
Training_1.jpg	21	19	0	0	111
Training_2.jpg	24	24	0	1	101
Training_3.jpg	25	23	0	1	89
Training_4.jpg	24	21	0	1	84
Training_5.jpg	24	22	0	0	93
Training_6.jpg	24	22	0	3	100
Training_7.jpg	22	22	0	1	95

numFaces : total number of faces in the picture
numHit : number of faces successfully detected
numRepeat : number of faces repeatedly detected
numFalse : number of case where a non-face is reported
run time : run time of the face detection routine
* run time measure in Pentium3 700 MHz, 448 Mega Memory; run time includes gender detection algorithm

The face detection algorithm shows 93.3 % of right hit rate, and 0 % of repeat rate, and 4.2 % of false hit rate. The average run time is 96 seconds.

In order to see if this algorithm works for other than the 7 training images, last year's sample picture was test, and the result is as shown in Fig. 17. The results show that 20. out of 24 faces have been successfully located, and there was no repeat or false detection

CHAPTER-9

References

[1] Lu Wei. Research and Implementation of Face Detection and Recognition based on Android Platform [D]. Southwest Jiaotong University, 2014.

[2] Zhang Peng. The number of users of mobile App in China has exploded [J]. Communications World, 2012, 46:11-12.

[3] Liu Chao. Design and Implementation of Face recognition System based on Android Platform [D]. Jilin University, 2013.

[4] Chiara Turati, Viola Macchi Cassia, F. S., and Leo, I. Newborns face recognition: Role of inner and outer facial features. Child Development 77, 2 (2006), 297311.

[5] "LBPH-Based Enhanced Real Time Face Recognition "International Journal of Advanced Computer Science and Applications Vol.10, No. 5, Published in 2019.

[6] " Face Detection using LBP Features"Jo Chang-yeon CS 229 Final Project Report December 12 ,2008.

[7] "Face Recognition System Based on LBPH Algorithm", International Journal of Engineering & advanced technology(IJEAT),Volume-8,Issue- 5S,May,2019.

[8] Himanshu Tiwari1"Live Attendance System via Face Recognition" International Journal for Research in Applied Science & Engineering Technology (IJRASET) Volume 6 Issue IV, April 2018.

[9] "FACE RECOGNITION TECHNIQUE USING ICA AND LBPH", Khusbu Rani1, Sukhbir , kamboj2,IRJET,Volume-04,issu- 08,August-2017

[10] E.Varadharajan, R.Dharani, S.Jeevitha, B.Kavinmathi, S.Hemalatha,Automatic attendance management system using face detection, Online International Conference on Green Engineering and Technologies (IC-GET), 978-1-5090-4556-3/16/$31.00 ©2016 IEEE.

[11] Naveed Khan Balcoh, M.HaroonYousaf, Waqar Ahma nd M.IramBaig, Algorithm for efficient Attendance Management: FaceRecognition Based approach, International Journal of Computer Science Issue, Vol.9, Issue 4, No 1, July 2012.

[12] An Accurate System for Face Detection and Recognition Abdelmgeid A. Ali1, Tarek Abd El-Hafeez and Yosra Khalaf Mohany Journal of Advances in Mathematics and Computer Science33(3): 1-19, 2019; Article no.JAMCS.49994 ISSN: 2456-9968

[13] Smart Attendance Management System Using Face Recognition Kaneez Laila Bhatti1,*, Laraib Mughal1 ,EAI Endorsed Transactions on Creative Technologies 10 2018 | Volume 5 | Issue 17 | e4

[14] Shubhobrata Bhattacharya, Gowtham Sandeep Nainala, Prosenjit Das, Aurobinda Routray, "Smart Attendance Monitoring System (SAMS): A Face Recognition Based Attendance

System for Class Environment", 2018 IEEE 18th International Conference on Advanced Learning Technologies.

[15] Priyanka Wagh, Jagruti Chaudhari, Roshani Thakare, Sweta Patil" Attendance System Based on face Recognition using Eigen Face and PCA Algorithms", 2015 International Conference on Green Computing and Internet of Things (ICGCIoT).

[16] Naveed khan Balcoh, M. Haroon Yousaf, Waqar Ahmad, M. Iram Baig, "Algorithm for Efficient Attendance Management: Face Recognition based Approach", IJCSI International of Journal of Computer Science Issues, Vol 9, Issue 4, No 1, July 2012.

[17] Yasaman Heydarzadeh, Abolfazal Toroghi Haghigh at,"An Efficient Face Detection Method Using Adaboost and Facial Parts", DOI 10.5013/IJSSST.a.12.04.01

[18] Divyansh Methi, Abhishek Chauhan, Divyanshu Gupta" Attendance System Using Face Recognition", International Journal Advanced Research in Science, Engineering and Technology, vol. 4, Issue 5, May 2017.

[19] Abhishek Jha,"Class Room Attendance System Facial Recognition System", The International Journal OF Mathematics, Science, Technology and Management vol. 2, issue 3.