A PROJECT REPORT

ON

Automated Attendance System Using Machine Learning

Submitted in partial fulfillment of the requirements for the award of degree in

MASTER OF COMPUTER APPLICATIONS

SUBMITTED BY

Katari.Mukesh

Regd. No: K6225245

UNDER THE GUIDANCE

OF

Dr. V T Ram Pavan Kumar. M

M.Tech APSET TSSET Ph.D., (PDF)

HOD

PG Department of Computer Science & Applications

DEPARTMENT OF MCA

ISO9001-2015 Certified Re-accredited ‘A^{++}’ by NAAC

KAKARAPARTI BHAVANARAYANA COLLEGE (AUTONOMOUS)

(Approved by AICTE, Affiliated to KRISHNA UNIVERSITY, MACHILIPATNAM)

Kothapet, Vijayawada, Krishna (Dst), pincode-520001

2022-2024

ISO:9001-2015Certified Re-accredited ‘A^{++}’ by NAAC

KAKARAPARTI BHAVANARAYANA PG COLLEGE(AUTONOMOUS)

(Approved by AICTE, Affiliated to KRISHNA UNIVERSITY, MACHILIPATNAM)

Kothapet, Vijayawada, Krishna (Dst), pincode-520001

DEPARTMENT OF MCA

CERTIFICATE

This is to certify that this work entitled “**Automated Attendance System Using Machine Learning**” is bonafide work carried out by Katari.Mukesh(2205045) in the partial fulfillment for the award of the degree in **MASTER OF COMPUTER APPLICATIONS of KRISHNA UNIVERSITY, MACHILIPATNAM** during the Academic year **2022-2024**. It is certify that the corrections / suggestions indicated for internal assessment have been incorporated in the report. The project work has been approved satisfies the academic requirements in respect of project work prescribed for the above degree.

Project Guide Head of the Department

External Examiner

ACKNOWLEDGMENT

The satisfaction that accompanies the successful completion of any task would be incomplete without mentioning the people who made it possible and whose constant guidance and encouragement crown all the efforts with success. This acknowledgement transcends the reality of formality when we would like to express deep gratitude and respect to all those people behind the screen who guided, inspired and helped me for the completion of the work. I wish to place on my record my deep sense gratitude to my project guide, Dr. V T Ram Pavan Kumar. M.Tech APSET TSSET Ph.D., (PDF)HOD for his constant motivation and valuable help throughout the project work.

My sincere thanks to **Dr. V T Ram Pavan Kumar. M**, **HOD, PG Department of Computer Science & Applications** for his guidance regarding the project. I extend gratitude to **Dr. S Venkatesh, DIRECTOR for P.G. COURSES** for his valuable suggestions.

STUDENT NAME

Katari.Mukesh

Regd.NO: K6225245

DECLARATION

I hereby declare the project work entitled “**Automated Attendance System Using Machine Learning”** submitted to K.B.N P.G COLLEGE affiliated to KRISHNA UNIVERSITY, has been done under the guidance of **Dr. V T Ram Pavan Kumar. M.Tech APSET TSSET Ph.D., (PDF)HOD, Department of MCA** during the period of study in that it has found formed the basis for the award of the degree/diploma or other similar title to any candidate of University.

Signature of the student

Name :Katari.Mukesh

Regd No : K6225245

College Name : K.B.N PG COLLEGE

DATE:

PLACE: VIJAYAWADA

ABSTRACT

Maintaining attendance is very important in all learning institutes for checking the performance of students. In most learning institutions, student attendances are manually taken by the use of attendance sheets issued by the department heads as part of regulation. The students sign in these sheets which are then filled or manually logged into a computer for future analysis. This method is tedious,time consuming and inaccurate as some students often sign for their absent colleagues. This method also makes it difficult to track the attendance of individual students in a large classroom environment. In this project, we propose the design and use of a face detection and recognition system to automatically detect students attending a lecture in a classroom and mark their attendance by recognizing their faces.

While other biometric methods of identification (such as iris scans or fingerprints) can be more accurate, students usually have to queue for long at the time they enter the classroom. Face recognition is chosen owing to its non-intrusive nature and familiarity as people primarily recognize other people based on their facial features. This (facial) biometric system will consist of an enrollment process in which the unique features of a persons' face will be stored in a database and then the processes of identification and verification. In these, the detected face in an image (obtained from the camera) will be compared with the previously stored faces captured at the time of enrollment.

The overall objective is to develop an automated class attendance management system comprising of a desktop application working in conjunction with a mobile application to perform

The Following Tasks:

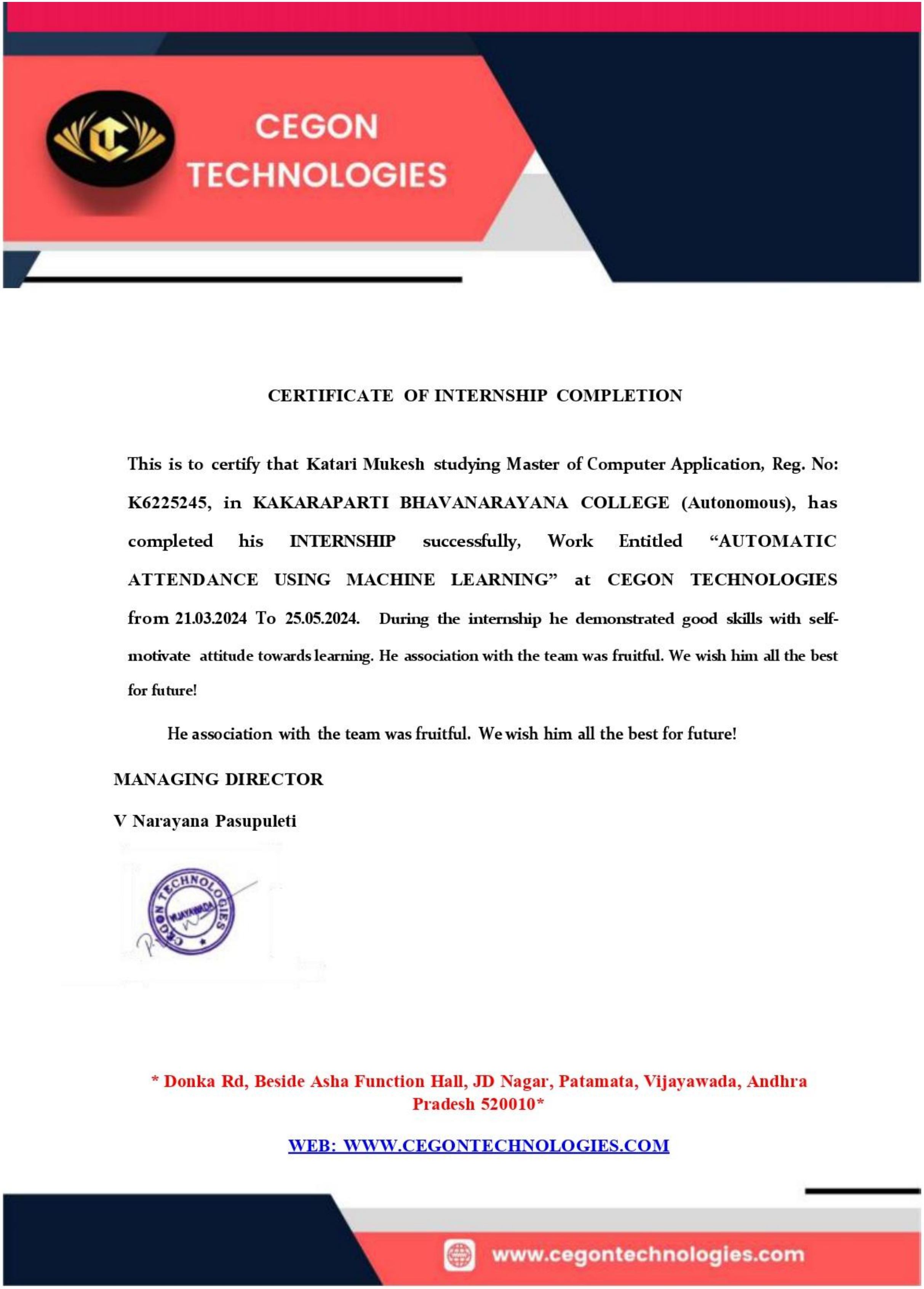
To detect faces real time.

To recognize the detected faces by the use of suitable algorithm.

To update the class attendance register after a successful match.

To design an architecture that constitutes the various components working harmoniously.

INDEX



CEGON TECHNOLOGIES

CERTIFICATE OF INTERNSHIP COMPLETION

This is to certify that Katari Mukesh studying Master of Computer Application, Reg. No: K6225245, in KAKARAPARTI BHAVANARAYANA COLLEGE (Autonomous), has completed his INTERNSHIP successfully, Work Entitled “AUTOMATIC ATTENDANCE USING MACHINE LEARNING” at CEGON TECHNOLOGIES from 21.03.2024 To 25.05.2024. During the internship he demonstrated good skills with self-motivate attitude towards learning. He association with the team was fruitful. We wish him all the best for future!

He association with the team was fruitful. We wish him all the best for future!

MANAGING DIRECTOR

V Narayana Pasupuleti

CEGON TECHNOLOGIES VIJAYAWADA

*** Donka Rd, Beside Asha Function Hall, JD Nagar, Patamata, Vijayawada, Andhra Pradesh 520010***

WEB: WWW.CEGONTECHNOLOGIES.COM

www.cegontechnologies.com

CHAPTER-1

INTRODUCTION

1.1 ABOUT THE PROJECT

The traditional manual methods of monitoring student attendance in lectures are tedious as the signed attendance sheets have to be manually logged into a computer system for analysis This is tedious, time consuming and prone to inaccuracies as some students in the department often traditional sign for their absent colleagues, rendering this method in effective in tracking the students' class attendance.

Use of the face detection and recognition system in lieu of the methods will provide a fast and effective method of capturing student attendance accurately while offering secure, stable and robust storage of the system records,mal-attendance will be control, where upon authorization, one can access them for purposes like administration, parents or even the students themselves.

automatic attendance is fast and time saving ,we can connect to the webcam and take attendance and without human interference,we can access the throught the excel or csv format

The purpose of this project is to develop an automatic attendance system using machine learning and facial recognition technology. The system aims to replace traditional manual attendance methods with a more efficient, accurate, and secure approach that saves time, reduces administrative workload, and prevents fraudulent attendance practices.

1.2 OBJECTIVE

The overall objective is to develop an automated class attendance management system comprising of a desktop application working in conjunction with a mobile application to perform the following tasks:

- To detect faces real time.
- To recognize the detected faces by the use of a suitable algorithm.
- To update the class attendance, register after a successful match.
- To design an architecture that constitutes the various components working harmoniously.
- Ensure the system has high accuracy in recognizing faces to minimize errors.
- Reduce the time and effort required for taking attendance.
- Protect against unauthorized access and tampering with attendance records.

1.3 SCOPE OF THE PROJECT

We are setting up to design a system comprising of two modules. The first module (face detector) is a mobile component, which is basically a camera application that captures student faces and stores them in a file using computer vision face detection algorithms and face extraction techniques.

The second module is a desktop application that does face recognition of the captured images (faces) in the file, marks the students register and then stores the results in a database for future analysis. The module is a desktop application that does face recognition of the captured images (faces) in the file, marks the students register and then stores the results in a database for future analysis.

To design a system comprising of two modules. The first module (face detector) is a mobile component, which is basically a camera application that captures student t faces and stores them in a file using computer vision face detection algorithms and face extraction techniques

We are designed the project with the help of opencv it was a deep learning library to analyze face with the given data set and take the attendance for periodically time the attendance are store in csv file .

Facial Acknowledgment technology Facial or confirming a person’s personality utilization their facial highlights . key components incorporate confront location, include extraction , and confront matching .

Face Location: Finds faces inside an image.Feature Extraction Recognizes and measures key facial features. Face Coordinating Compares extricated highlights against a database of known faces.

CHAPTER-2
LITERATURE SURVEY

Digital Image Processing

Digital Image Processing is the processing of images which are digital in nature by a digital computer.Digital image processing techniques are motivated by three major applications mainly:

- Improvement of pictorial information for human perception
- Image processing for autonomous machine application
- Efficient storage and transmission.

Human Perception

This application employs methods capable of enhancing pictorial information for human interpretation and analysis. Typical applications include, noise filtering, content enhancement mainly contrast enhancement or deblurring and remote sensing .

Facial Recognition Technology

Facial recognition involves identifying or verifying a person's identity using their facial features. Key components include face detection, feature extraction, and face matching.

- **Face Detection**: Locates faces within an image.'
- **Feature Extraction**: Identifies and quantifies key facial features.
- **Face Matching**: Compares extracted features against a database of known faces.

Machine Vision Applications

In this, the interest is on the procedures for extraction of image information suitable for computer processing, Typical applications include :

- Industrial machine vision for product assembly and inspection.
- Automated target detection and tracking .
- Finger print recognition.
- Machine processing of aerial and satellite imagery for weather

Facial detection and recognition fall within the machine vision application of

digital image processing.

A digital image f (x, y) is discretized both in spatial co-ordinates by grids

and in brightness by quantization. Effectively, the image can be represented as a matrix whose

row, column indices specify a point in the image and the element value identifies gray level

value at that point. These elements are referred to as pixels orpels.

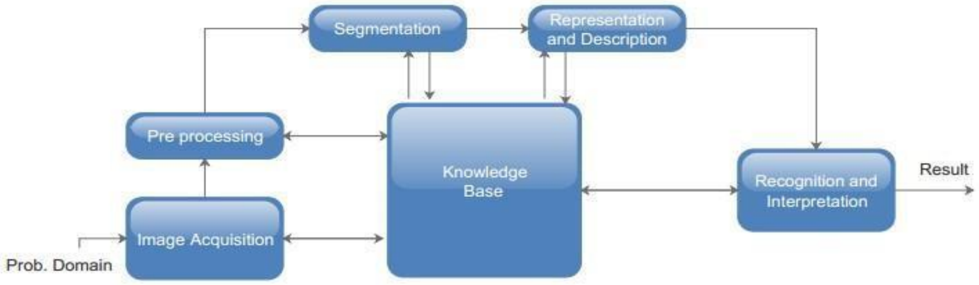
Typically following image processing applications, the image size which is used is × , elements, × pels or × pixels.Quantization of these matrix pixels is done at 8 bits for black and white images and 24 bits for colored images (because of the three-color planes Red,Green and Blue each at 8 bits) A digital image f (x, y) is discretized both in spatial co-ordinates by grids and in brightness by quantization. Effectively, the image can be represented as a matrix whose row, column indices specify a point in the image and the element value identifies gray level value at that point. These elements are referred to as pixels orpelsA digital image f (x, y) is discretized both in spatial co-ordinates by grids and in brightness by quantization. Effectively, the image can be represented as a matrix whose row, column indices specify a point in the image and the element value identifies gray level value at that point. These elements are referred to as pixels orpels

Steps In Digital Image Processing

Digital image processing involves the following basic tasks;

- Image Acquisition - An imaging sensor and the capability to digitize the signal produced by the sensor.

- Preprocessing – Enhances the image quality, filtering, contrast enhancement etc.
- Segmentation – Partitions an input image into constituent parts of objects.
- Description/featureSelection extracted description of image objects suitable for further computer processing.
- Recognition and Interpretation – Assigning a label to the objectbased on the information provided by its descriptor.
- Interpretation assigns meaning to a set of labeled objects.Knowledge Base–This helps for efficient processing as well as inter module cooperation



CHAPTER-3
SYSTEM ANALYSIS

3.1 EXISTING SYSTEM

Fingerprint Based recognition system: In the Fingerprint based existing attendance system, a portable fingerprint device needs to be configured with the students fingerprint earlier. Later either during the lecture hours or before, the student needs to record the fingerprint on the configured device to ensure their attendance for the day. The problem with this approach is that during the lecture time it may distract the attention of the students.

RFID (Radio Frequency Identification) Based recognition system: In the RFID based existing system, the student needs to carry a Radio Frequency Identity Card with them and place the ID on the card reader to record their presence for the day. The system is capable of to connect to RS232 and record the attendance to the saved database. The system is capable of to connect to RS232 and record the attendance to the saved database There are possibilities for the fraudulent access may occur. Some are students may make use of other students ID to ensure their presence when the particular student is absent or they even try to misuse it sometimes.

Iris Based Recognition System: In the Iris based student attendance system, the student needs to stand in front of a camera, so that the camera will scan the Iris of the student. The scanned iris is matched with data of student stored in the database and the attendance on their presence needs be updated. This reduces the paper and pen workload of the faculty member of the institute. This also reduces the chances of proxies in the class, and helps in maintaining the student records safe. It is awireless biometric technique that solves the problem of spurious attendance and the trouble of laying the corresponding network. . This reduces the paper and pen workload of the faculty member of the institute. This also reduces the chances of proxies in the class, and helps in maintaining the student records safe. It is a wireless biometric technique that solves the problem of spurious attendance and the trouble of laying the corresponding network

3.2 PROPOSED SYSTEM

The task of the proposed system is to capture the face of each student and to store it in the database for their attendance. The face of the student needs to be captured in such a manner that all the feature of the students' face needs to be detected, even the seating and the posture of the student need to be recognized. There is no need for the teacher to manually take attendance in the class because the system records a video and through further processing steps the face is being recognized and the attendance database is updated**.**

The proposed system aims to automate the attendance process using face detection and recognition technology, leveraging machine learning algorithms to identify and verify individuals in real-time. This system is designed for educational institutions, workplaces, and events, providing a more efficient, accurate, and secure method for tracking attendance.

Advantages of the Proposed System

- Efficiency: Automates the attendance process, reducing time and effort.
- Accuracy: High accuracy in face detection and recognition, minimizing errors.
- Security: Secure storage of attendance data and protection against fraudulent practices.
- Scalability: Can handle a large number of users and easily integrate with existing systems.
- Convenience: Provides real-time updates and easy access to attendance data.

3.2.1 DETAILS

Machine Learning Infrastructure: The automatic attendance system is built upon a robust machine learning infrastructure, designed to handle large datasets and perform complex analyses to ensure accurate attendance tracking. This infrastructure utilizes advanced algorithms, including facial recognition and pattern recognition, to identify and verify individuals, thus automating the attendance process.

User Registration and Biometric Enrollment: Users register on the platform by providing personal information and biometric data, such as facial images, which are securely stored in the system. Multi-factor authentication, including biometrics and secure passwords, enhances the security of the registration process and prevents unauthorized access.

Attendance Recording and Verification: The system automatically records attendance by matching captured images with the registered database, logging timestamps and attendance status in real-time. Verification mechanisms, such as liveness detection, ensure that the system distinguishes between live individuals and photos or videos, reducing the risk of spoofing.

Privacy and Data Security: Sensitive user data, including biometric information and attendance records, is encrypted and securely stored to protect against unauthorized access or breaches. Privacy-preserving techniques, like different privacy and secure multi-party computation, provide users with control over their data and ensure compliance with data protection regulations.

Real-Time Monitoring and Notifications: The system provides real-time monitoring capabilities, allowing administrators to view attendance statuses and trends through a user-friendly dashboard. Automated notifications alert users and administrators about attendance discrepancies, upcoming classes, or important announcements, ensuring timely communication and intervention.

Integration with Existing Systems: The attendance system seamlessly integrates with existing educational or organizational management systems, enabling smooth data exchange and enhanced functionality. APIs facilitate interoperability with various platforms, such as learning management systems (LMS), human resource management systems (HRMS), and payroll systems, streamlining administrative processes.

Performance Evaluation and Continuous Improvement: The system continually evaluates its performance by analyzing accuracy rates, false positives, and user feedback to identify areas for improvement. Machine learning models are regularly retrained with new data to adapt to changing conditions and improve recognition accuracy over time.

Compliance and Ethical Considerations: The system adheres to relevant regulations and ethical guidelines, ensuring that the use of biometric data respects user privacy and complies with legal standards. Regular audits and transparency reports provide accountability and build trust among users, ensuring the ethical deployment of the technology.

Emergency and Exception Handling: The system includes protocols for handling emergencies or exceptions, such as power outages, camera malfunctions, or unauthorized access attempts. Backup procedures and manual attendance options ensure continuity and reliability, even in unforeseen circumstances, maintaining the integrity of attendance records.

3.2.2 Impact on Environment

Implementing an automatic attendance system using machine learning can have several environmental impacts, both positive and negative. Here are the key aspects to consider:

Positive Impacts:

Reduction in Paper Usage:

Traditional attendance systems often rely on paper-based records, which contribute to deforestation and waste. An automatic attendance system eliminates the need for paper, significantly reducing paper consumption and waste.

Energy Efficiency in Operations:

Modern machine learning models, particularly those optimized for energy efficiency, can run on low-power hardware. This reduces the overall energy consumption compared to older, more manual systems that may require more resources for record-keeping and management.

Decreased Carbon Footprint:

Automating attendance reduces the need for physical travel to manage and verify attendance records, such as supervisors traveling between locations. This can lead to a reduction in transportation-related emissions.

Optimized Resource Utilization:

Efficient management of attendance can lead to better utilization of resources such as classroom space, heating, cooling, and lighting. Accurate attendance data can help institutions optimize their schedules and physical resources, leading to reduced energy consumption.

Negative Impacts:

Energy Consumption of Data Centers:

Machine learning systems, particularly those that involve heavy data processing and storage, require substantial computational power. Data centers that host these systems can be energy-intensive, potentially increasing the carbon footprint if not managed sustainably.

Resource Intensive Manufacturing:

The production of electronic components required for the system involves mining and manufacturing processes that can have environmental impacts, such as habitat destruction and pollution..

Energy-Efficient Hardware:

Investing in energy-efficient hardware and optimizing machine learning algorithms for lower power consumption can reduce the overall environmental footprint.

E-Waste Management:

Implementing robust e-waste management protocols to recycle and repurpose old equipment can minimize the negative environmental impact.

3.2.3 SAFETY

Implementing an automatic attendance system using face recognition offers several benefits in terms of safety and efficiency, but it also raises some concerns that need to be addressed. Here's a breakdown of the safety aspects:

Benefits:

1.**Contactless Verification:**

1.Reduces the need for physical interaction, which is particularly advantageous in preventing the spread of infectious diseases.

2.Eliminates the need for touch-based attendance systems like fingerprint scanners.

2**.Accuracy and Efficiency:**

1.Minimizes human error and the potential for buddy punching (when one person clocks in for another).

2.Provides real-time data and reports, making attendance tracking more efficient.

3.**Security Enhancements:**

1.Difficult to forge or duplicate facial data compared to traditional ID cards or passwords.

2.Can be integrated with other security systems for enhanced campus or workplace security.

Safety Measures and Best Practices:

Data Protection and Privacy:

1. Implement strong encryption methods for storing and transmitting biometric data.
2. Develop clear policies on data usage and retention, and ensure users are informed about how their data will be used.

Regulatory Compliance:

1. Ensure the system complies with relevant data protection laws and regulations.
2. Regularly audit the system to ensure compliance and address any vulnerabilities.

System Accuracy:

1. Use high-quality, unbiased datasets to train the recognition model.
2. Regularly test the system for accuracy and adjust as necessary to reduce errors and biases.

User Consent:

1. Obtain explicit consent from individuals before enrolling their biometric data into the system.
2. Provide an opt-out option for those who do not wish to participate in the biometric attendance system.

Security Measures:

1. Use multi-factor authentication (MFA) where possible to enhance security.
2. Regularly update and patch the system to protect against new vulnerabilities.

3.2.4 ETHICS:

Privacy:

- Data Collection: Biometric data is highly sensitive. Collecting and storing such data should be done with utmost care, ensuring that individuals' privacy is protected.
- Informed Consent: Individuals should be fully informed about how their data will be used and should give explicit consent before their data is collected.

Transparency:

- Clear Communication: Organizations must clearly communicate the purpose, scope, and functioning of the face recognition system.
- Access to Information: Individuals should have the right to access their data and understand how it is being used.

Bias and Fairness:

- Algorithmic Bias: Face recognition systems have been shown to have varying levels of accuracy across different demographic groups. Ensuring that the system is fair and unbiased is crucial.

- Equal Treatment: The system should treat all individuals equally, regardless of race, gender, or other personal characteristics.

3.2.5 COST

Initial Setup Costs

Hardware:

- Standard Cameras: ₹4,100 to ₹16,400 each. For a small project, you might need 2-5 cameras, totaling ₹8,200 to ₹82,000.On-premises Server: A basic server might cost between ₹82,000 to ₹200,000.
- **Software:**Facial Recognition Software License: For a small project, simpler and less expensive software could cost between ₹200,000 to ₹820,000 per year.
- **Integration Costs**: Basic integration with existing systems might range from ₹41,000 to ₹200,000.

Installation and Setup:

- Camera Installation: ₹8,200 to ₹20,500 per camera, totaling ₹16,400 to ₹102,500 for 2-5 cameras.
- Server Setup: ₹41,000 to ₹100,000.

Training and Support:

- Training: Basic training for staff might cost around ₹20,000 to ₹100,000.
- Initial Support: Included in the software license or a small additional cost, say around ₹20,000 to ₹50,000.
-

Ongoing Costs

- **Data Management:**Data Storage: For a small project, monthly costs might be around ₹4,100 to ₹20,500.
- Bandwidth: Costs will vary, but can be estimated at ₹1,000 to ₹5,000 per month.

System Maintenance:

- Regular Maintenance: Annual costs might be lower, around ₹20,000 to ₹100,000.
- Software Updates: Small additional costs, estimated at ₹10,000 to ₹50,000 annually.

Compliance and Security:

- Data Protection Measures: Basic measures might cost around ₹20,000 to ₹50,000 annually.

Ongoing Annual Costs:

- Data Management: ₹61,200 to ₹306,000
- Maintenance and Updates: ₹30,000 to ₹150,000
- Compliance and Security: ₹20,000 to ₹50,000
- Total Ongoing Annual Costs: ₹111,200

Cost Optimization Tips

- Pilot Project: Start with a limited deployment to evaluate the system's performance before expanding.
- Cloud Services: Consider using cloud-based solutions to reduce initial hardware costs.
- Open-Source Software: Explore open-source options to lower software costs, though it may require more technical expertise.
- Vendor Negotiation: Negotiate with vendors for better deals on hardware and software, especially if purchasing multiple units.

3.2.6 TYPE

Educational Institutions:

Schools and Colleges:

- Purpose: Automate student and staff attendance to enhance efficiency and accuracy.
- Scale: Small to medium, typically requiring coverage for classrooms, labs, and administrative offices.

- Key Features: Real-time attendance tracking, integration with existing school management systems, and reporting capabilities.

Universities:

- Purpose: Manage attendance for large numbers of students and faculty across various departments.
- Scale: Medium to large, covering multiple buildings and campuses.
- Key Features: Scalability, advanced reporting, integration with student information systems, and support for diverse schedules.

Corporate Offices:

- Small and Medium Enterprises (SMEs)
- Purpose: Improve accuracy of employee attendance tracking and streamline payroll processing.
- Scale: Small to medium, typically covering one or a few office locations.
- Key Features: Integration with HR and payroll systems, real-time monitoring, and basic access control.

3.2.7 STANDARDS

The samples for database should be increase, as to increase the efficiency of detection. Also, the more the expensive the camera, the easier its algorithm is likely detecting the person.

3.2.8 Feasibility Study

A feasibility study for implementing an automatic attendance system using face recognition involves assessing various factors to determine if the project is viable and beneficial for your organization. Here’s a comprehensive outline of a feasibility study:

Technical Feasibility

- Hardware Requirements: High-resolution cameras, powerful processors, and adequate storage.

Software Requirements: Face recognition software, integration with existing systems, and a robust database management system.

- Infrastructure: Reliable network connectivity and secure data storage facilities.
- Accuracy and Reliability: Modern face recognition algorithms achieve high accuracy rates, typically over 95%, but require regular updates and maintenance.

Economic Feasibility

- Initial Costs: Investment in high-quality cameras, servers, and software licenses.
- Operational Costs: Maintenance, software updates, and potential staffing for system monitoring.
- Cost-Benefit Analysis: Long-term savings from reduced administrative work and improved accuracy.

Operational Feasibility

- Integration with Existing Systems: The system should seamlessly integrate with current attendance and HR systems.
- User Training: Adequate training for staff and users to ensure smooth adoption.
- Support and Maintenance: Establishing a support system for troubleshooting and regular maintenance

legal and Ethical Feasibility

- Data Privacy Laws: Compliance with data protection regulations such as GDPR or CCPA.
- Consent: Ensuring informed consent from all users for data collection and processing.
- Ethical Considerations: Addressing concerns about surveillance and misuse of biometric data.
- A feasibility study for implementing an automatic attendance system using face recognition involves assessing various factors to determine .
- if the project is viable and beneficial for your organization.
- Modern face recognition algorithms achieve high accuracy rate

CHAPTER-4

SYSTEM REQUIREMENTS SPECIFICATION

4.1 FUNCTIONAL REQUIREMENTS

- Use machine learning algorithms to process and recognize faces.
- Achieve high accuracy in face recognition (95%+).
- Handle variations in lighting, angle, and facial expressions.

Attendance Marking

Description: The system marks attendance based on recognized faces.

Functional Requirements:

- Automatically update attendance records in the database.
- Handle instances of absence and partial attendance.
- Provide manual override for errors.

Reporting

Description: The system generates attendance reports for analysis.

Functional Requirements:

- Generate daily, weekly, and monthly attendance reports.
- Export reports in various formats (PDF, Excel).
- Allow filtering and sorting of data.

4.2 Non-functional Requirements

Performance Requirements

- The system should recognize faces within 2 seconds.

- The system should support simultaneous recognition of up to 50 faces.

Security Requirements

- Data encryption for stored images and personal information.
- Role-based access control for different user classes.
- Regular security audits and updates.

Usability Requirements

- Intuitive GUI for easy navigation.
- Mobile compatibility for accessing reports.

Reliability Requirements

- System uptime of 99.9%.
- Robust error handling and recovery mechanisms.

Other Requirements

Data Backup

- Regular automated backups of the database.
- Offsite backup storage.

Maintenance

- Regular software updates.
- Technical support and maintenance services.

Appendices:

Appendix A: Glossary

Appendix B: Use Case Diagrams

Appendix C: Data Flow Diagrams

CHAPTER-5

SYSTEM DESIGN

5.1 SYSTEM SPECIFICATIONS

Hardware Requirements:

System	:	Pentium i3
Hard Disk	:	500 GB.
Monitor	:	14’ Colour Monitor.
Mouse	:	Optical Mouse.
Ram	:	4 GB.

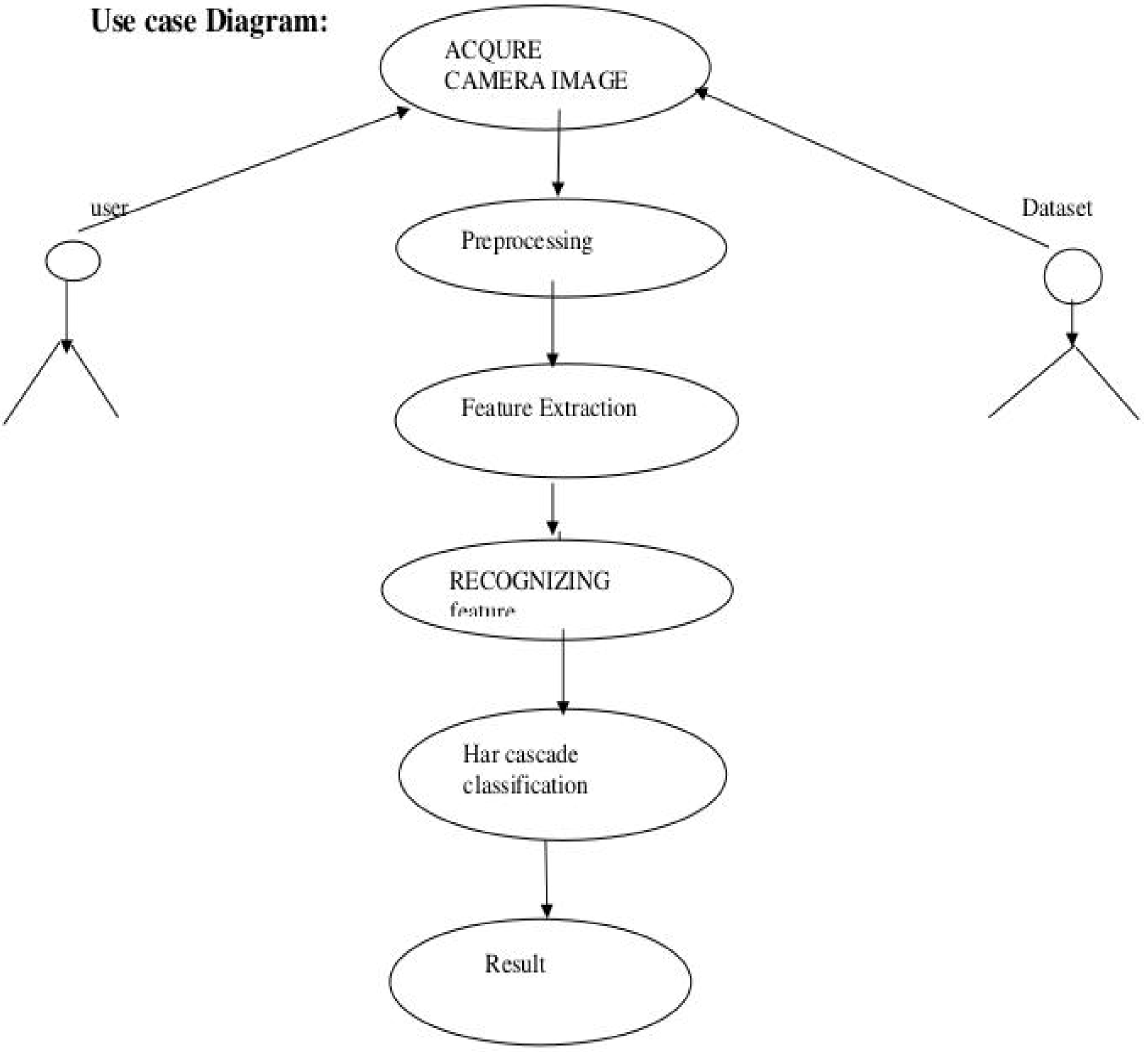
Software Requirements:

Operating system	:	Windows 8/10.
Coding Language	:	PYTHON
Software	:	Jupyter
Applications	:	visual studios,ms excel

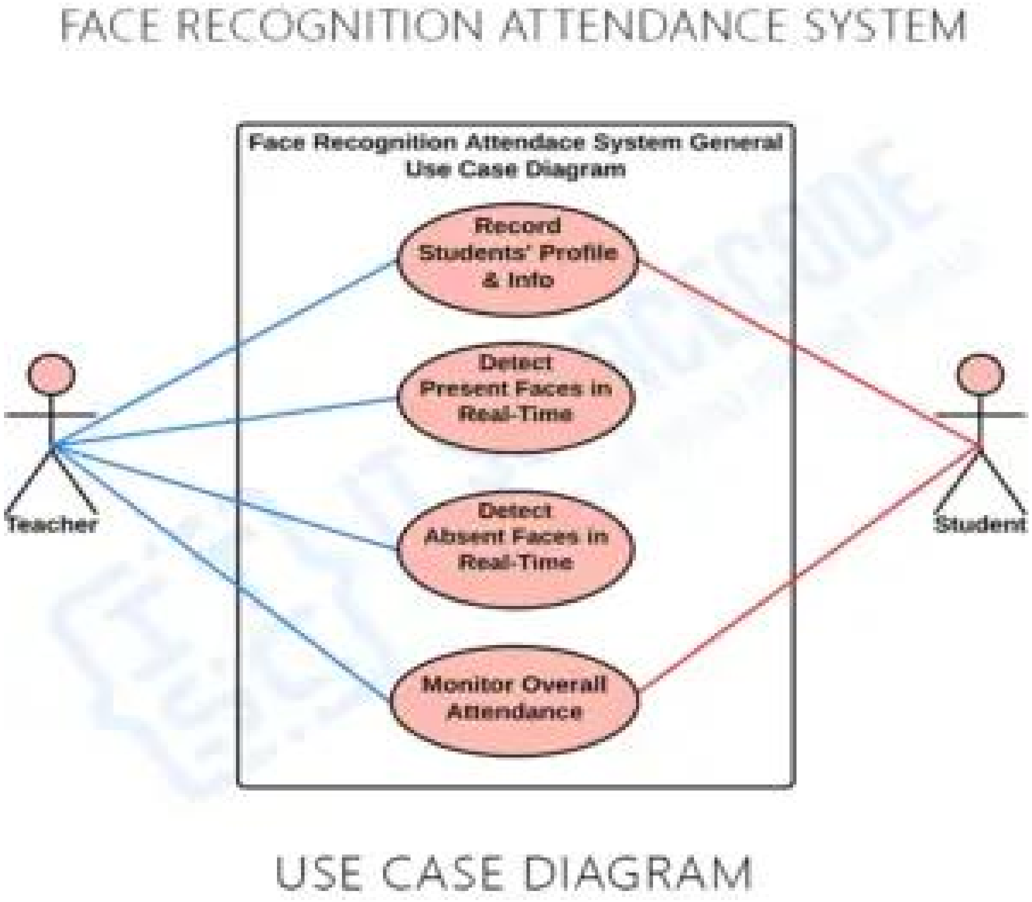
5.2 UML DIAGRAMS

GOALS:

Use case Diagram:

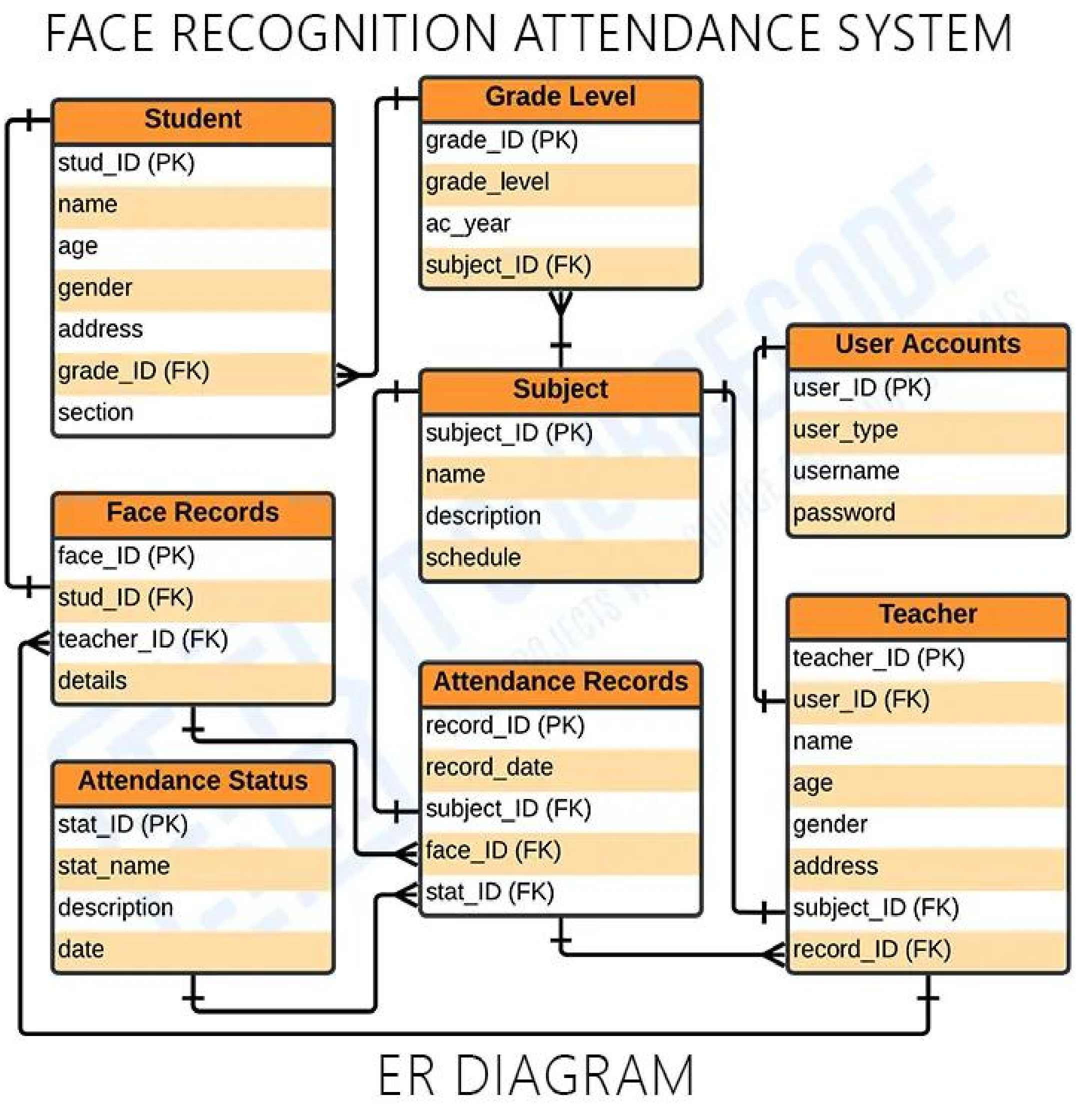


5.2.1 USE CASE DIAGRAM

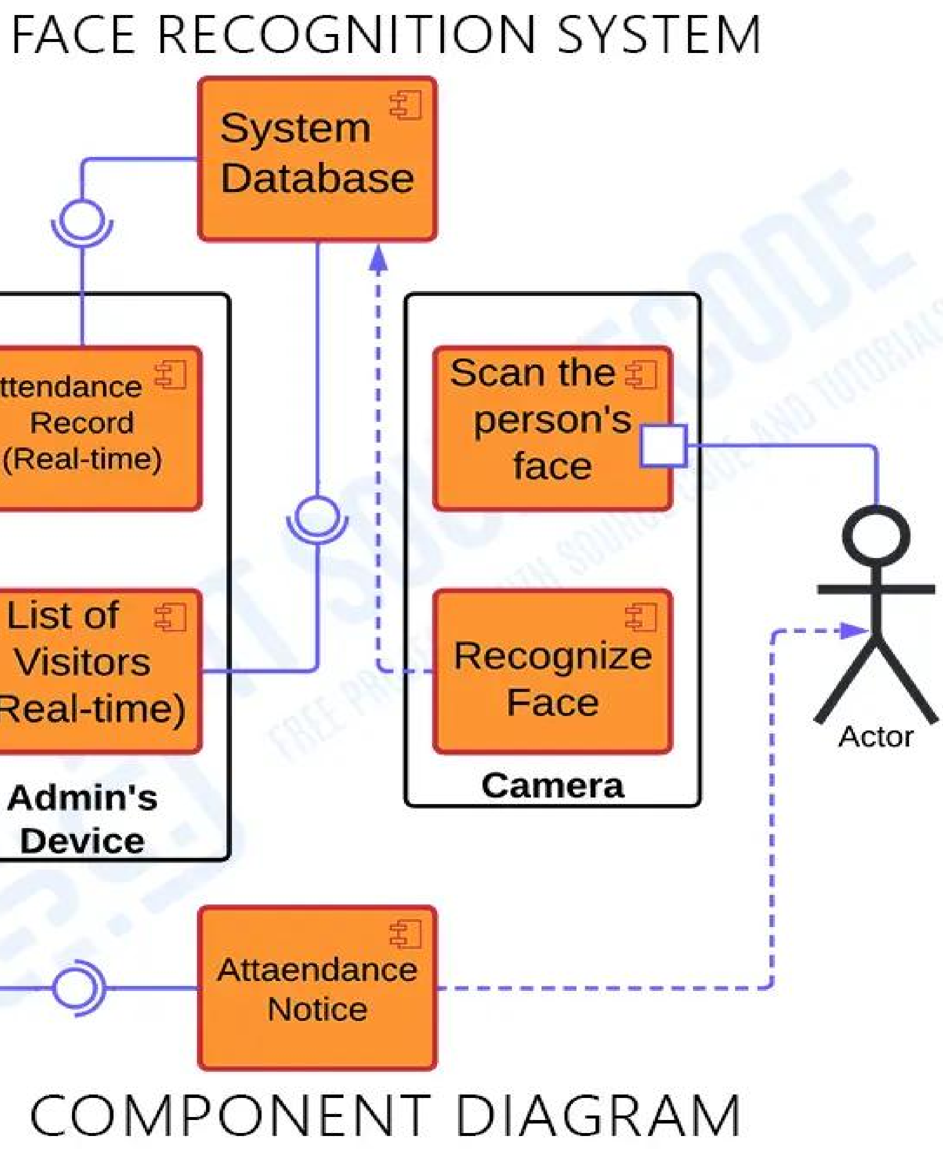

FACE RECOGNITION ATTENDANCE SYSTEM
Face Recognition Attendace System General Use Case Diagram
Record Students' Profile & Info
Detect Present Faces in Real-Time
Detect Absent Faces in Real-Time
Monitor Overall Attendance
Teacher
Student
USE CASE DIAGRAM

- this diagram captures the system's functionality from a user's perspective. It shows the interactions between actors (users or external systems) and the system.
- In the context of face recognition attendance, the actors could be school administrators, personnel, and students.
- Use cases might include "Mark Attendance," "View Attendance Records," and "Manage User Profiles."

5.2.2 CLASS DIAGRAM

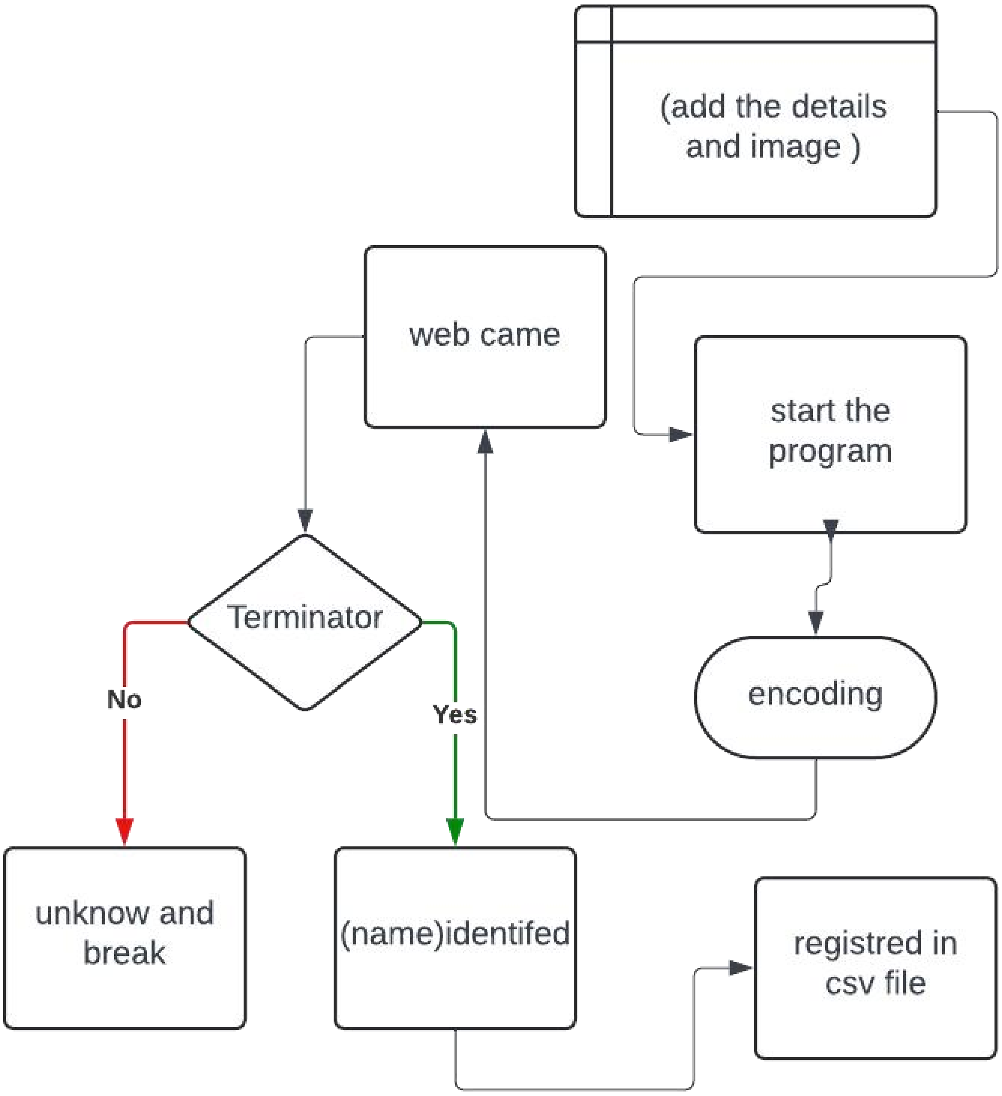


5.2.3 SEQUENCE DIAGRAM



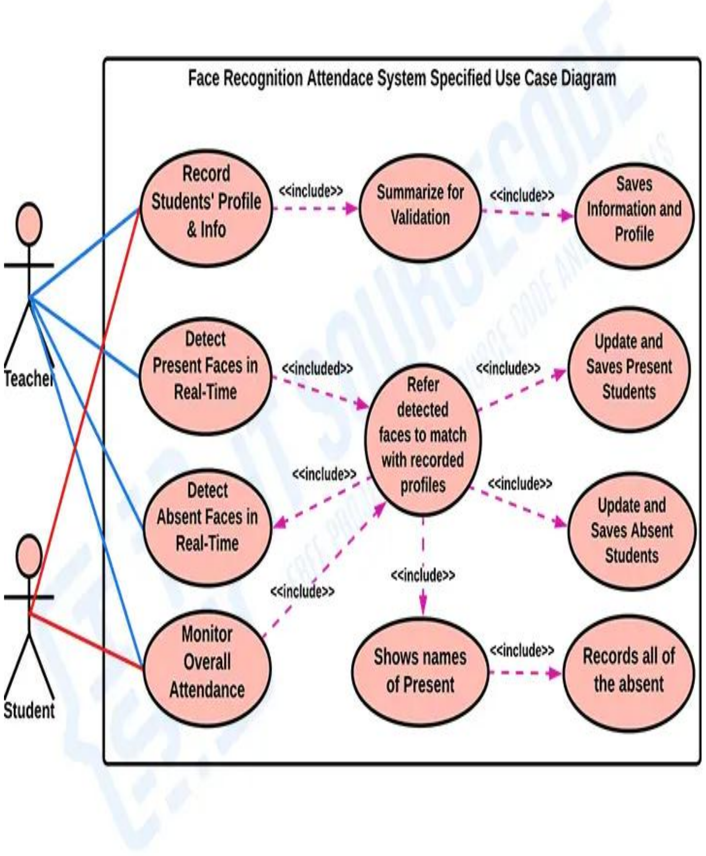
5.2.4 COLLABORATION DIAGRAM

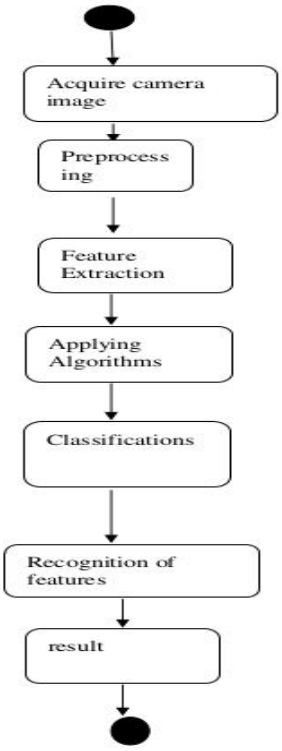
Activity diagrams are graphical representations of workflows of stepwise activities and actions with support for choice, iteration and concurrency. In the Unified Modeling Language, activity diagrams can be used to describe the business and operational step-by-step workflows of components in a system. An activity diagram shows the overall flow of control.


(add the details and image)
web came
start the program
Terminator
No
Yes
encoding
unknow and break
(name)identifed
registred in csv file

5.2.5 ACTIVITY DIAGRAM

FACE RECOGNITION ATTENDANCE SYSTEM


Face Recognition Attedace System Specified Use Case Diagram
Teacher
Student
Record Students' Profile & Info
<<include>>
Summarize for Validation
<<include>>
Saves Information and Profile
Detect Present Faces in Real-Time
<<included>>
Refer detected faces to match with recorded profiles
<<include>>
Update and Saves Present Students
<<include>>
Detect Absent Faces in Real-Time
<<include>>
Update and Saves Absent Students
<<include>>
<<include>>
Monitor Overall Attendance
Shows names of Present
<<include>>
Records all of the absent



SOFTWARE ENVIRONMENT :

What is Python :

Below are some facts about Python.

Python is currently the most widely used multi-purpose, high-level programming language.

Python allows programming in Object-Oriented and Procedural paradigms. Python programs generally are smaller than other programming languages like Java.

Programmers have to type relatively less and indentation requirement of the language, makes them readable all the time.

Python language is being used by almost all tech-giant companies like – Google, Amazon, Facebook, Instagram, Dropbox, Uber… etc.

The biggest strength of Python is huge collection of standard library which can be used for the following –

Machine Learning

GUI Applications (like Kivy, Tkinter, PyQt etc.)

Web frameworks like Django (used by YouTube, Instagram, Dropbox)

Image processing (like Opencv, Pillow)

Web scraping (like Scrapy, BeautifulSoup, Selenium)

Test frameworks

Multimedia

Advantages of Python :-

Let’s see how Python dominates over other languages.

1. Extensive Libraries :

Python downloads with an extensive library and it contain code for various purposes like regular expressions, documentation-generation, unit-testing, web browsers, threading, databases, CGI, email, image manipulation, and more. So, we don’t have to write the complete code for that manually.

2. Extensible:

As we have seen earlier, Python can be extended to other languages. You can write some of your code in languages like C++ or C. This comes in handy, especially in projects.

3. Embeddable:

Complimentary to extensibility, Python is embeddable as well. You can put your Python code in your source code of a different language, like C++. This lets us add scripting capabilities to our code in the other language.

4. Improved Productivity

The language's simplicity and extensive libraries render programmers more productive than languages like Java and C++ do. Also, the fact that you need to write less and get more things done.

5. IOT Opportunities

Since Python forms the basis of new platforms like Raspberry Pi, it finds the future bright for the Internet Of Things. This is a way to connect the language with the real world. When working with Java, you may have to create a class to print 'Hello World'. But in Python, just a print statement will do. It is also quite easy to learn, understand, and code. This is why when people pick up Python, they have a hard time adjusting to other more verbose languages like Java.

7. Readable:

Because it is not such a verbose language, reading Python is much like reading English. This is the reason why it is so easy to learn, understand, and code. It also does not need curly braces to define blocks, and indentation is mandatory. This further aids the readability of the code.

8. Object-Oriented

This language supports both the procedural and object-oriented programming paradigms. While functions help us with code reusability, classes and objects let us model the real world. A class allows the encapsulation of data and functions into one.

9. Free and Open-Source

Like we said earlier, Python is freely available. But not only can you download Python for free, but you can also download its source code, make changes to it, and even distribute it. It downloads with an extensive collection of libraries to help you with your tasks.

10. Portable

When you code your project in a language like C++, you may need to make some changes to it if you want to run it on another platform. But it isn't the same with Python. Here, you need to code only once, and you can run it anywhere. This is called Write Once Run Anywhere (WORA). However, you need to be careful enough not to include any system-dependent features.

11. Interpreted

Lastly, we will say that it is an interpreted language. Since statements are executed one by one, debugging is easier than in compiled languages.

Any doubts till now in the advantages of Python? Mention in the comment section.

Advantages of Python Over Other Languages :

1. Less Coding

Almost all of the tasks done in Python requires less coding when the same task is done in other languages. Python also has an awesome standard library support, so you don’t have to search for any third-party libraries to get your job done. This is the reason that many people suggest learning Python to beginners.

2. Affordable

Python is free therefore individuals, small companies or big organizations can leverage the free available resources to build applications. Python is popular and widely used so it gives you better community support.

The 2019 Github annual survey showed us that Python has overtaken Java in the most popular programming language category.

3. Python is for Everyone

Python code can run on any machine whether it is Linux, Mac or Windows. Programmers need to learn different languages for different jobs but with Python, you can professionally build web apps, perform data analysis and machine learning, automate things, do web scraping and also build games and powerful visualizations. It is an all-rounder programming language.

Disadvantages of Python

So far, we’ve seen why Python is a great choice for your project. But if you choose it, you should be aware of its consequences as well. Let’s now see the downsides of choosing Python over another language.

1. Speed Limitations

We have seen that Python code is executed line by line. But since Python is interpreted, it often results in slow execution. This, however, isn’t a problem unless speed is a focal point for the project. In other words, unless high speed is a requirement, the benefits offered by Python are enough to distract us from its speed limitations.

2. Weak in Mobile Computing and Browsers

While it serves as an excellent server-side language, Python is much rarely seen on the client-side. Besides that, it is rarely ever used to implement smartphone-based applications. One such application is called Carbonnelle.

The reason it is not so famous despite the existence of Brython is that it isn’t that secure.

3. Design Restrictions

As you know, Python is dynamically-typed. This means that you don’t need to declare the type of variable while writing the code. It uses duck-typing. But wait, what’s that? Well, it just means that if it looks like a duck, it must be a duck. While this is easy on the programmers during coding, it can raise run-time errors.

4. Underdeveloped Database Access Layers

Compared to more widely used technologies like JDBC (Java DataBase Connectivity) and ODBC (Open DataBase Connectivity), Python’s database access layers are a bit underdeveloped. Consequently, it is less often applied in huge enterprises.

5. Simple

No, we’re not kidding. Python’s simplicity can indeed be a problem. Take my example. I don’t do Java, I’m more of a Python person. To me, its syntax is so simple that the verbosity of Java code seems unnecessary.

This was all about the Advantages and Disadvantages of Python Programming Language.

History of Python : -

What do the alphabet and the programming language Python have in common? Right, both start with ABC. If we are talking about ABC in the Python context, it's clear that the programming language ABC is meant. ABC is a general-purpose programming language and programming environment, which had been developed in the Netherlands, Amsterdam, at the CWI (Centrum Wiskunde &Informatica). The greatest achievement of ABC was to influence the design of Python.Python was conceptualized in the late 1980s. Guido van Rossum worked that time in a project at the CWI, called Amoeba, a distributed operating system. In an interview with Bill Venners1, Guido van Rossum said: "In the early 1980s, I worked as an implementer on a team building a language called ABC at Centrum voor Wiskunde en Informatica (CWI). I don't know how well people know ABC's influence on Python.

I try to mention ABC's influence because I'm indebted to everything I learned during that project and to the people who worked on it."Later on in the same Interview, Guido van Rossum continued: "I remembered all my experience and some of my frustration with ABC. I decided to try to design a simple scripting language that possessed some of ABC's better properties, but without its problems. So I started typing. I created a simple virtual machine, a simple parser, and a simple runtime. I made my own version of the various ABC parts that I liked. I created a basic syntax, used indentation for statement grouping instead of curly braces or begin-end blocks, and developed a small number of powerful data types: a hash table (or dictionary, as we call it), a list, strings, and numbers."

What is Machine Learning : -

Before we take a look at the details of various machine learning methods, let's start by looking at what machine learning is, and what it isn't. Machine learning is often categorized as a subfield of artificial intelligence, but I find that categorization can often be misleading at first brush. The study of machine learning certainly arose from research in this context, but in the data science application of machine learning methods, it's more helpful to think of machine learning as a means of building models of data.

Fundamentally, machine learning involves building mathematical models to help understand data. "Learning" enters the fray when we give these models tunable parameters that can be adapted to observed data; in this way the program can be considered to be "learning" from the data. Once these models have been fit to previously seen data, they can be used to predict and understand aspects of newly observed data. I'll leave to the reader the more philosophical digression regarding the extent to which this type of mathematical, model-based "learning" is similar to the "learning" exhibited by the human brain.Understanding the problem setting in machine learning is essential to using these tools effectively, and so we will start with some broad categorizations of the types of approaches we'll discuss here.

Categories Of Machine Leaning :-

At the most fundamental level, machine learning can be categorized into two main types: supervised learning and unsupervised learning.

Supervised learning involves somehow modeling the relationship between measured features of data and some label associated with the data; once this model is determined, it can be used to apply labels to new, unknown data. This is further subdivided into classification tasks and regression tasks: in classification, the labels are discrete categories, while in regression, the labels are continuous quantities. We will see examples of both types of supervised learning in the following section.

Unsupervised learning involves modeling the features of a dataset without reference to any label, and is often described as "letting the dataset speak for itself." These models include tasks such as clustering and dimensionality reduction. Clustering algorithms identify distinct groups of data, while dimensionality reduction algorithms search for more succinct representations of the data. We will see examples of both types of unsupervised learning in the following section.

Need for Machine Learning

Human beings, at this moment, are the most intelligent and advanced species on earth because they can think, evaluate and solve complex problems. On the other side, AI is still in its initial stage and haven’t surpassed human intelligence in many aspects. Then the question is that what is the need to make machine learn? The most suitable reason for doing this is, “to make decisions, based on data, with efficiency and scale”.

Lately, organizations are investing heavily in newer technologies like Artificial Intelligence, Machine Learning and Deep Learning to get the key information from data to perform several real-world tasks and solve problems. We can call it data-driven decisions taken by machines, particularly to automate the process. These data-driven decisions can be used, instead of using programing logic, in the problems that cannot be programmed inherently. The fact is that we can’t do without human intelligence, but other aspect is that we all need to solve real-world problems with efficiency at a huge scale. That is why the need for machine learning arises.

Challenges in Machines Learning :-

While Machine Learning is rapidly evolving, making significant strides with cybersecurity and autonomous cars, this segment of AI as whole still has a long way to go. The reason behind is that ML has not been able to overcome number of challenges.

The challenges that ML is facing currently are −

Quality of data − Having good-quality data for ML algorithms is one of the biggest challenges. Use of low-quality data leads to the problems related to data preprocessing and feature extraction.

Time-Consuming task − Another challenge faced by ML models is the consumption of time especially for data acquisition, feature extraction and retrieval.

Lack of specialist persons − As ML technology is still in its infancy stage, availability of expert resources is a tough job.

No clear objective for formulating business problems − Having no clear objective and well-defined goal for business problems is another key challenge for ML because this technology is not that mature yet.

Issue of overfitting & underfitting − If the model is overfitting or underfitting, it cannot be represented well for the problem.

Curse of dimensionality − Another challenge ML model faces is too many features of data points. This can be a real hindrance.

Difficulty in deployment − Complexity of the ML model makes it quite difficult to be deployed in real life.

Applications of Machines Learning :-

Machine Learning is the most rapidly growing technology and according to researchers we are in the golden year of AI and ML. It is used to solve many real-world complex problems which cannot be solved with traditional approach. Following are some real-world applications of ML −

Emotion analysis

Sentiment analysis

Error detection and prevention

Weather forecasting and prediction

Stock market analysis and forecasting

Speech synthesis

Speech recognition

Customer segmentation

Object recognition

Fraud detection

Face recognition

Fraud prevention

Recommendation of products to customer in online shopping

How to Start Learning Machine Learning?

Arthur Samuel coined the term “Machine Learning” in 1959 and defined it as a “Field of study that gives computers the capability to learn without being explicitly programmed”.

And that was the beginning of Machine Learning! In modern times, Machine Learning is one of the most popular (if not the most!) career choices. According to Indeed, Machine Learning Engineer Is The Best Job of 2019 with a 344% growth and an average base salary of $146,085 per year.

But there is still a lot of doubt about what exactly is Machine Learning and how to start learning it? So this article deals with the Basics of Machine Learning and also the path you can follow to eventually become a full-fledged Machine Learning Engineer. Now let’s get started!!!

How to start learning ML?

This is a rough roadmap you can follow on your way to becoming an insanely talented Machine Learning Engineer. Of course, you can always modify the steps according to your needs to reach your desired end-goal!

Step 1 – Understand the Prerequisites

In case you are a genius, you could start ML directly but normally, there are some prerequisites that you need to know which include Linear Algebra, Multivariate Calculus, Statistics, and Python. And if you don’t know these, never fear! You don’t need a Ph.D. degree in these topics to get started but you do need a basic understanding.

(a) Learn Linear Algebra and Multivariate Calculus

Both Linear Algebra and Multivariate Calculus are important in Machine Learning. However, the extent to which you need them depends on your role as a data scientist. If you are more focused on application heavy machine learning, then you will not be that heavily focused on maths as there are many common libraries available. But if you want to focus on R&D in Machine Learning, then mastery of Linear Algebra and Multivariate Calculus is very important as you will have to implement many ML algorithms from scratch.

(b) Learn Statistics

Data plays a huge role in Machine Learning. In fact, around 80% of your time as an ML expert will be spent collecting and cleaning data. And statistics is a field that handles the collection, analysis, and presentation of data. So it is no surprise that you need to learn it!!!
Some of the key concepts in statistics that are important are Statistical Significance, Probability Distributions, Hypothesis Testing, Regression, etc. Also, Bayesian Thinking is also a very important part of ML which deals with various concepts like Conditional Probability, Priors, and Posteriors, Maximum Likelihood, etc.

(c) Learn Python

Some people prefer to skip Linear Algebra, Multivariate Calculus and Statistics and learn them as they go along with trial and error. But the one thing that you absolutely cannot skip is Python! While there are other languages you can use for Machine Learning like R, Scala, etc. Python is currently the most popular language for ML. In fact, there are many Python libraries that are specifically useful for Artificial Intelligence and Machine Learning such as Keras, TensorFlow, Scikit-learn, etc.

So if you want to learn ML, it’s best if you learn Python! You can do that using various online resources and courses such as Fork Python available Free on GeeksforGeeks.

Step 2 – Learn Various ML Concepts

Now that you are done with the prerequisites, you can move on to actually learning ML (Which is the fun part!!!) It’s best to start with the basics and then move on to the more complicated stuff. Some of the basic concepts in ML are:

(a) Terminologies of Machine Learning

Model – A model is a specific representation learned from data by applying some machine learning algorithm. A model is also called a hypothesis.

Feature – A feature is an individual measurable property of the data. A set of numeric features can be conveniently described by a feature vector. Feature vectors are fed as input to the model. For example, in order to predict a fruit, there may be features like color, smell, taste, etc.

Target (Label) – A target variable or label is the value to be predicted by our model. For the fruit example discussed in the feature section, the label with each set of input would be the name of the fruit like apple, orange, banana, etc.

Training – The idea is to give a set of inputs(features) and it's expected outputs(labels), so after training, we will have a model (hypothesis) that will then map new data to one of the categories trained on.

Prediction – Once our model is ready, it can be fed a set of inputs to which it will provide a predicted output(label).

(b) Types of Machine Learning

Supervised Learning – This involves learning from a training dataset with labeled data using classification and regression models. This learning process continues until the required level of performance is achieved.

Unsupervised Learning – This involves using unlabelled data and then finding the underlying structure in the data in order to learn more and more about the data itself using factor and cluster analysis models.

Semi-supervised Learning – This involves using unlabelled data like Unsupervised Learning with a small amount of labeled data. Using labeled data vastly increases the learning accuracy and is also more cost-effective than Supervised Learning.

Reinforcement Learning – This involves learning optimal actions through trial and error. So the next action is decided by learning behaviors that are based on the current state and that will maximize the reward in the future.

Advantages of Machine learning :-

1. Easily identifies trends and patterns -

Machine Learning can review large volumes of data and discover specific trends and patterns that would not be apparent to humans. For instance, for an e-commerce website like Amazon, it serves to understand the browsing behaviors and purchase histories of its users to help cater to the right products, deals, and reminders relevant to them. It uses the results to reveal relevant advertisements to them.

2. No human intervention needed (automation)

With ML, you don't need to babysit your project every step of the way. Since it means giving machines the ability to learn, it lets them make predictions and also improve the algorithms on their own. A common example of this is anti-virus softwares; they learn to filter new threats as they are recognized. ML is also good at recognizing spam.

3. Continuous Improvement

As ML algorithms gain experience, they keep improving in accuracy and efficiency. This lets them make better decisions. Say you need to make a weather forecast model. As the amount of data you have keeps growing, your algorithms learn to make more accurate predictions faster.

4. Handling multi-dimensional and multi-variety data

Machine Learning algorithms are good at handling data that are multi-dimensional and multi-variety, and they can do this in dynamic or uncertain environments.

5. Wide Applications

You could be an e-tailer or a healthcare provider and make ML work for you. Where it does apply, it holds the capability to help deliver a much more personal experience to customers while also targeting the right customers.

Disadvantages of Machine Learning :-

1. Data Acquisition

Machine Learning requires massive data sets to train on, and these should be inclusive/unbiased, and of good quality. There can also be times where they must wait for new data to be generated.

2. Time and Resources

ML needs enough time to let the algorithms learn and develop enough to fulfill their purpose with a considerable amount of accuracy and relevancy. It also needs massive resources to function. This can mean additional requirements of computer power for you.

3. Interpretation of Results

Another major challenge is the ability to accurately interpret results generated by the algorithms. You must also carefully choose the algorithms for your purpose.

4. High error-susceptibility

Machine Learning is autonomous but highly susceptible to errors. Suppose you train an algorithm with data sets small enough to not be inclusive. You end up with biased predictions coming from a biased training set. This leads to irrelevant advertisements being displayed to customers. In the case of ML, such blunders can set off a chain of errors that can go undetected for long periods of time. And when they do get noticed, it takes quite some time to recognize the source of the issue, and even longer to correct it.

Python Development Steps : -

Guido Van Rossum published the first version of Python code (version 0.9.0) at alt.sources in February 1991. This release included already exception handling, functions, and the core data types of list, dict, str and others. It was also object oriented and had a module system.
Python version 1.0 was released in January 1994. The major new features included in this release were the functional programming tools lambda, map, filter and reduce, which Guido Van Rossum never liked.Six and a half years later in October 2000, Python 2.0 was introduced. This release included list comprehensions, a full garbage collector and it was supporting unicode.Python flourished for another 8 years in the versions 2.x before the next major release as Python 3.0 (also known as "Python 3000" and "Py3K") was released. Python 3 is not backwards compatible with Python 2.x. The emphasis in Python 3 had been on the removal of duplicate programming constructs and modules, thus fulfilling or coming close to fulfilling the 13th law of the Zen of Python: "There should be one -- and preferably only one -- obvious way to do it."Some changes in Python 7.3:

Print is now a function

Views and iterators instead of lists

The rules for ordering comparisons have been simplified. E.g. a heterogeneous list cannot be sorted, because all the elements of a list must be comparable to each other.

There is only one integer type left, i.e. int. long is int as well.

The division of two integers returns a float instead of an integer. "//" can be used to have the "old" behaviour.

Text Vs. Data Instead Of Unicode Vs. 8-bit

Purpose :-

We demonstrated that our approach enables successful segmentation of intra-retinal layers—even with low-quality images containing speckle noise, low contrast, and different intensity ranges throughout—with the assistance of the ANIS feature.

Python

Python is an interpreted high-level programming language for general-purpose programming. Created by Guido van Rossum and first released in 1991, Python has a design philosophy that emphasizes code readability, notably using significant whitespace.

Python features a dynamic type system and automatic memory management. It supports multiple programming paradigms, including object-oriented, imperative, functional and procedural, and has a large and comprehensive standard library.

Python is Interpreted − Python is processed at runtime by the interpreter. You do not need to compile your program before executing it. This is similar to PERL and PHP.

Python is Interactive − you can actually sit at a Python prompt and interact with the interpreter directly to write your programs.

Python also acknowledges that speed of development is important. Readable and terse code is part of this, and so is access to powerful constructs that avoid tedious repetition of code. Maintainability also ties into this may be an all but useless metric, but it does say something about how much code you have to scan, read and/or understand to troubleshoot problems or tweak behaviors. This speed of development, the ease with which a programmer of other languages can pick up basic Python skills and the huge standard library is key to another area where Python excels. All its tools have been quick to implement, saved a lot of time, and several of them have later been patched and updated by people with no Python background - without breaking.

Modules Used in Project :-

face_recognition:

- The face_recognition library is not part of the standard Python library but is a third-party package.
- It simplifies face recognition tasks by providing high-level functions for face detection and encoding.
- You use it to find face locations, compute face encodings, and compare faces for recognition.

Numpy

Numpy is a general-purpose array-processing package. It provides a high-performance multidimensional array object, and tools for working with these arrays.

It is the fundamental package for scientific computing with Python. It contains various features including these important ones:

A powerful N-dimensional array object

Sophisticated (broadcasting) functions

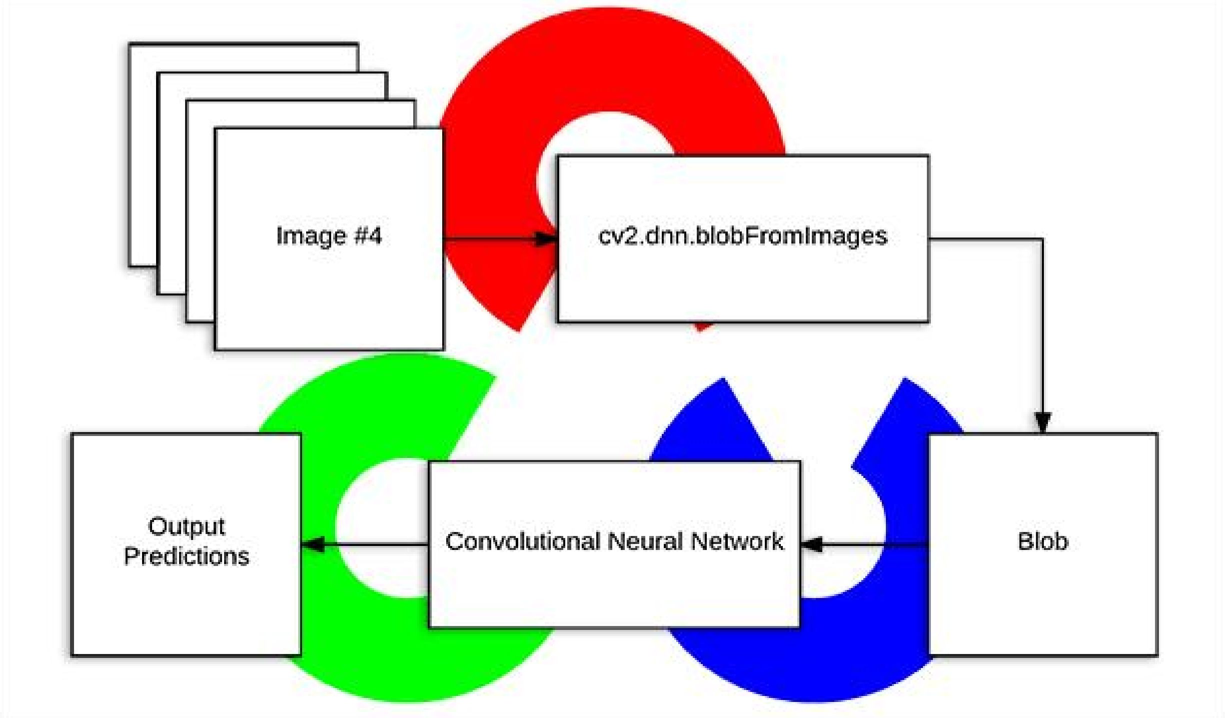
Tools for integrating C/C++ and Fortran code

Useful linear algebra, Fourier transform, and random number capabilities

Besides its obvious scientific uses, Numpy can also be used as an efficient multi-dimensional container of generic data. Arbitrary data-types can be defined using Numpy which allows Numpy to seamlessly and speedily integrate with a wide variety of databases.

OpenCV (cv2):

- OpenCV (Open Source Computer Vision Library) is a popular open-source computer vision and image processing library.
- t provides tools for image and video manipulation, object detection, face recognition, and more.
- In your code, you use OpenCV for capturing video frames from the webcam, drawing rectangles around detected faces, and displaying the results.



Pandas

Pandas is an open-source Python Library providing high-performance data manipulation and analysis tool using its powerful data structures. Python was majorly used for data munging and preparation. It had very little contribution towards data analysis. Pandas solved this problem. Using Pandas, we can accomplish five typical steps in the processing and analysis of data, regardless of the origin of data load, prepare, manipulate, model, and analyze. Python with Pandas is used in a wide range of fields including academic and commercial domains including finance, economics, Statistics, analytics, etc.

Matplotlib

Matplotlib is a Python 2D plotting library which produces publication quality figures in a variety of hardcopy formats and interactive environments across platforms. Matplotlib can be used in Python scripts, the Python and IPython shells, the Jupyter Notebook, web application servers, and four graphical user interface toolkits. Matplotlib tries to make easy things easy and hard things possible. You can generate plots, histograms, power spectra, bar charts, error charts, scatter plots, etc., with just a few lines of code. For examples, see the sample plots and thumbnail gallery.

For simple plotting the pyplot module provides a MATLAB-like interface, particularly when combined with IPython. For the power user, you have full control of line styles, font properties, axes properties, etc, via an object oriented interface or via a set of functions familiar to MATLAB users.

Scikit – learn

Scikit-learn provides a range of supervised and unsupervised learning algorithms via a consistent interface in Python. It is licensed under a permissive simplified BSD license and is distributed under many Linux distributions, encouraging academic and commercial use. Python

Python is an interpreted high-level programming language for general-purpose programming. Created by Guido van Rossum and first released in 1991, Python has a design philosophy that emphasizes code readability, notably using significant whitespace.

Python features a dynamic type system and automatic memory management. It supports multiple programming paradigms, including object-oriented, imperative, functional and procedural, and has a large and comprehensive standard library.

Python is Interpreted − Python is processed at runtime by the interpreter. You do not need to compile your program before executing it. This is similar to PERL and PHP.

Python is Interactive − you can actually sit at a Python prompt and interact with the interpreter directly to write your programs.

Python also acknowledges that speed of development is important. Readable and terse code is part of this, and so is access to powerful constructs that avoid tedious repetition of code. Maintainability also ties into this may be an all but useless metric, but it does say something about how much code you have to scan, read and/or understand to troubleshoot problems or tweak behaviors. This speed of development, the ease with which a programmer of other languages can pick up basic Python skills and the huge standard library is key to another area where Python excels. All its tools have been quick to implement, saved a lot of time, and several of them have later been patched and updated by people with no Python background - without breaking.

Install Python Step-by-Step in Windows and Mac :

Python a versatile programming language doesn’t come pre-installed on your computer devices. Python was first released in the year 1991 and until today it is a very popular high-level programming language. Its style philosophy emphasizes code readability with its notable use of great whitespace.

The object-oriented approach and language construct provided by Python enables programmers to write both clear and logical code for projects. This software does not come pre-packaged with Windows.

How to Install Python on Windows and Mac :

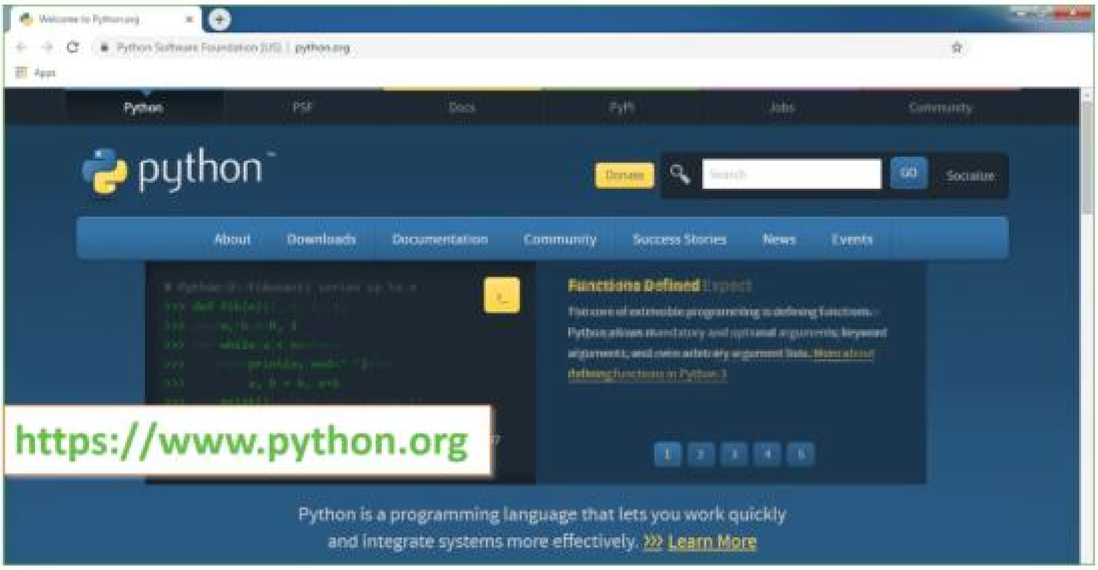
There have been several updates in the Python version over the years. The question is how to install Python? It might be confusing for the beginner who is willing to start learning Python but this tutorial will solve your query. The latest or the newest version of Python is version 3.7.4 or in other words, it is Python 3.

Note: The python version 3.7.4 cannot be used on Windows XP or earlier devices.

Before you start with the installation process of Python. First, you need to know about your System Requirements. Based on your system type i.e. operating system and based processor, you must download the python version. My system type is a Windows 64-bit operating system. So the steps below are to install python version 3.7.4 on Windows 7 device or to install Python 3. Download the Python Cheatsheet here.The steps on how to install Python on Windows 10, 8 and 7 are divided into 4 parts to help understand better.

Download the Correct version into the system

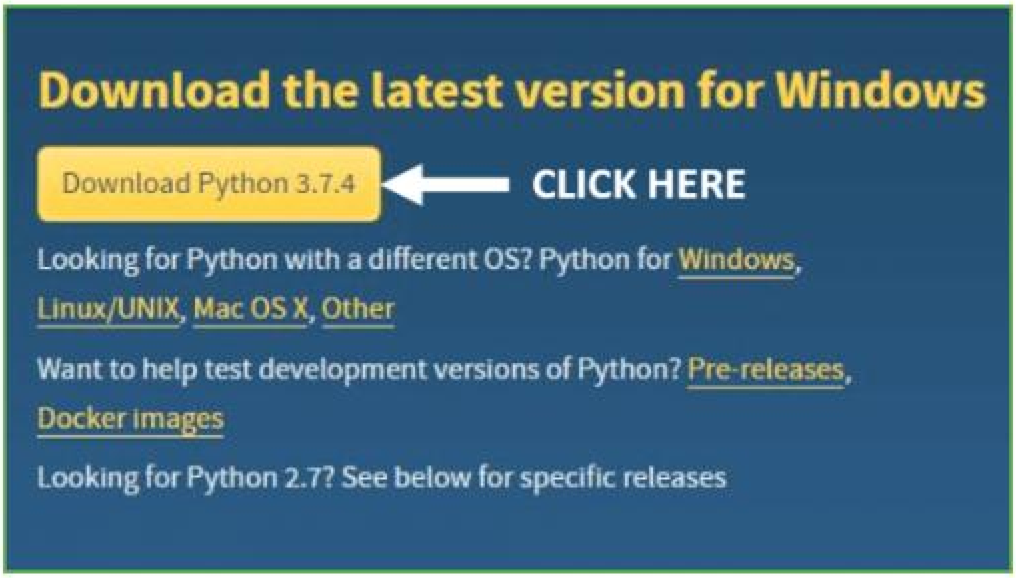
Step 1: Go to the official site to download and install python using Google Chrome or any other web browser. OR Click on the following link: https://www.python.org



Python is a programming language that lets you work quickly and integrate systems more effectively. >>> Learn More

Now, check for the latest and the correct version for your operating system.

Step 2: Click on the Download Tab.



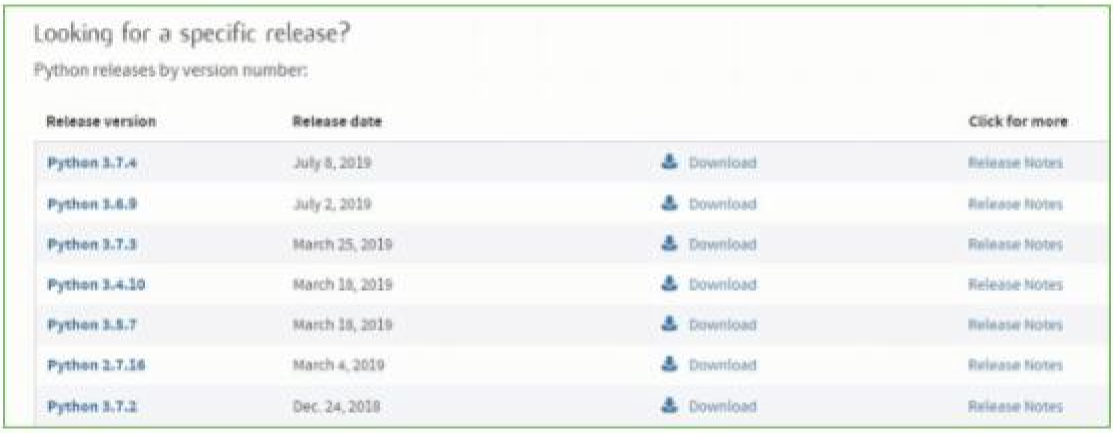
Download the latest version for Windows

Looking for Python with a different OS? Python for Windows, Linux/UNIX, Mac OS X, Other

Want to help test development versions of Python? Pre-releases, Docker images

Looking for Python 2.7? See below for specific releases

Step 3: You can either select the Download Python for windows 3.7.4 button in Yellow Color or you can scroll further down and click on download with respective to their version. Here, we are downloading the most recent python version for windows 3.7.4



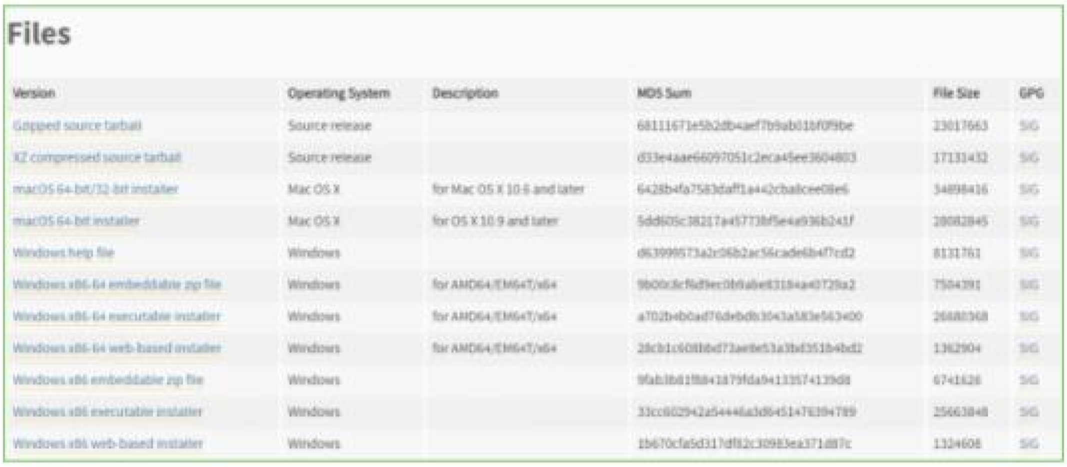
Looking for a specific release?

Python releases by version number:

Release version	Release date		Click for more
Python 3.7.4	July 8, 2019	Download	Release Notes
Python 3.6.9	July 2, 2019	Download	Release Notes
Python 3.7.3	March 25, 2019	Download	Release Notes
Python 3.4.10	March 18, 2019	Download	Release Notes
Python 3.5.7	March 18, 2019	Download	Release Notes
Python 2.7.16	March 4, 2019	Download	Release Notes
Python 3.7.2	Dec. 24, 2018	Download	Release Notes

Step 4: Scroll down the page until you find the Files option.

Step 5: Here you see a different version of python along with the operating system.



Files

Version	Operating System	Description	MD5 Sum	File Size	GPG
Gzipped source tarball	Source release		68111671e5b2db4aef7b9ab01bf0f9be	23017663	SIG
XZ compressed source tarball	Source release		d33e4aae66097051c2eca45ee3604803	17131432	SIG
macOS 64-bit/32-bit installer	Mac OS X	for Mac OS X 10.6 and later	6428b4fa7583daff1a442cba8cee08e6	34898416	SIG
macOS 64-bit installer	Mac OS X	for OS X 10.9 and later	5dd605c38217a45773bf5e4a936b241f	28082845	SIG
Windows help file	Windows		d63999573a2c56b2ac56cade6b4f7cd2	8131761	SIG
Windows x86-64 embeddable zip file	Windows	for AMD64/EM64T/x64	9b00c8cf6d9ec0b9abe83184a40729a2	7504391	SIG
Windows x86-64 executable installer	Windows	for AMD64/EM64T/x64	a702b4b0ad76debdb3043a583e563400	26680368	SIG
Windows x86-64 web-based installer	Windows	for AMD64/EM64T/x64	28cb1c608bbd73ae8e53a3bd351b4bd2	1362904	SIG
Windows x86 embeddable zip file	Windows		9fab3b81f8841879fda94133574139d8	6741626	SIG
Windows x86 executable installer	Windows		33cc602942a54446a3d6451476394789	25663848	SIG
Windows x86 web-based installer	Windows		1b670cfa5d317df82c30983ea371d87c	1324608	SIG

- To download Windows 32-bit python, you can select any one from the three options: Windows x86 embeddable zip file, Windows x86 executable installer or Windows x86 web-based installer.

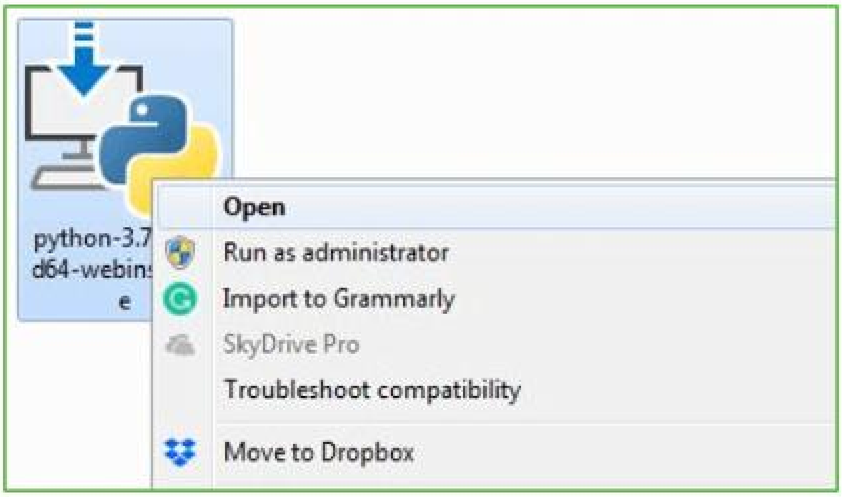
- To download Windows 64-bit python, you can select any one from the three options: Windows x86-64 embeddable zip file, Windows x86-64 executable installer or Windows x86-64 web-based installer.

Here we will install Windows x86-64 web-based installer. Here your first part regarding which version of python is to be downloaded is completed. Now we move ahead with the second part in installing python i.e. Installation

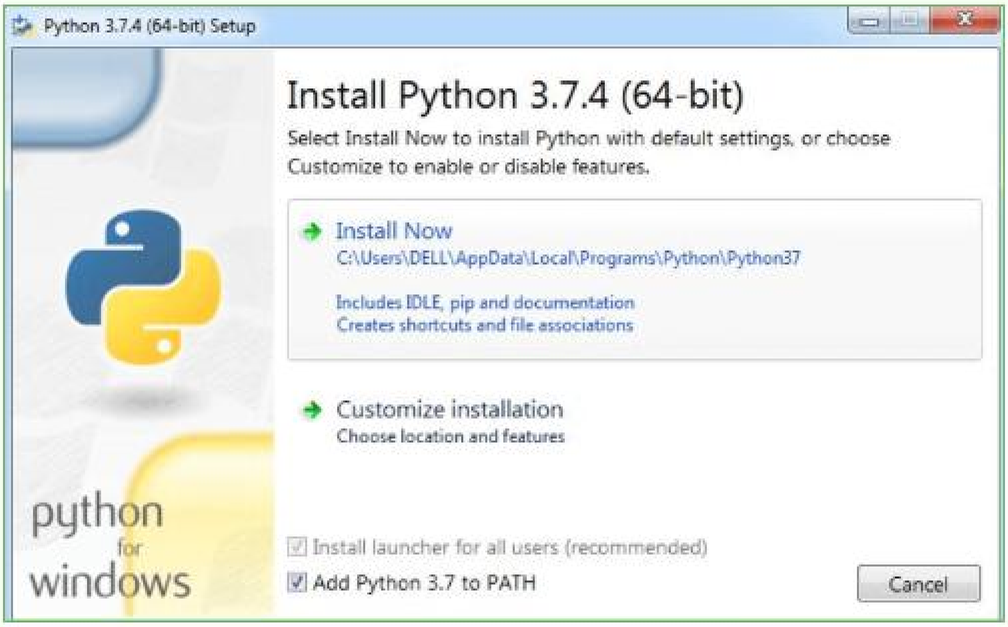
Note: To know the changes or updates that are made in the version you can click on the Release Note Option.

Installation of Python

Step 1: Go to Download and Open the downloaded python version to carry out the installation process.



Step 2: Before you click on Install Now, Make sure to put a tick on Add Python 3.7 to PATH.


Python 3.7.4 (64-bit) Setup

Install Python 3.7.4 (64-bit)

Select Install Now to install Python with default settings, or choose Customize to enable or disable features.

Install Now
C:\Users\DELL\AppData\Local\Programs\Python\Python37

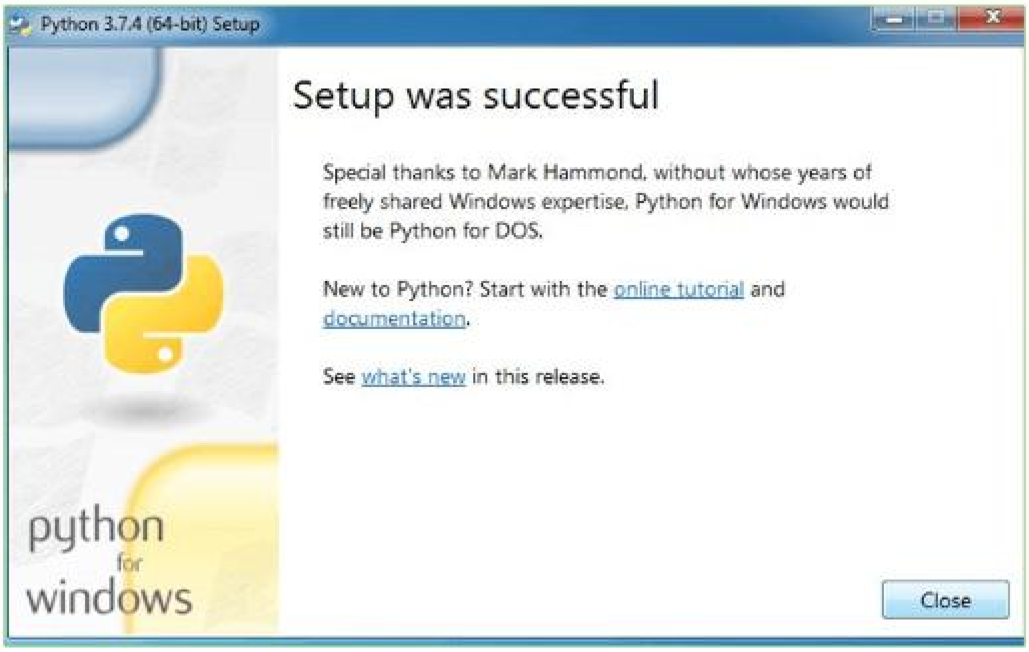
Includes IDLE, pip and documentation
Creates shortcuts and file associations

Customize installation
Choose location and features

☑ Install launcher for all users (recommended)

☑ Add Python 3.7 to PATH

Step 3: Click on Install NOW After the installation is successful. Click on Close.



Python 3.7.4 (64-bit) Setup

Setup was successful

Special thanks to Mark Hammond, without whose years of freely shared Windows expertise, Python for Windows would still be Python for DOS.

New to Python? Start with the online tutorial and documentation.

See what's new in this release.

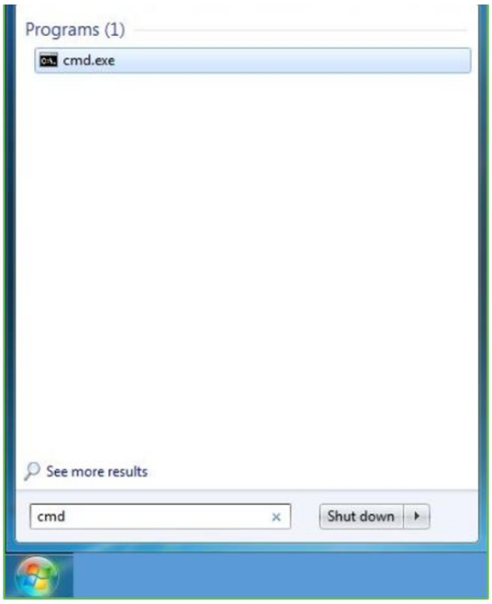
With these above three steps on python installation, you have successfully and correctly installed Python. Now is the time to verify the installation.

Note: The installation process might take a couple of minutes.

Verify the Python Installation

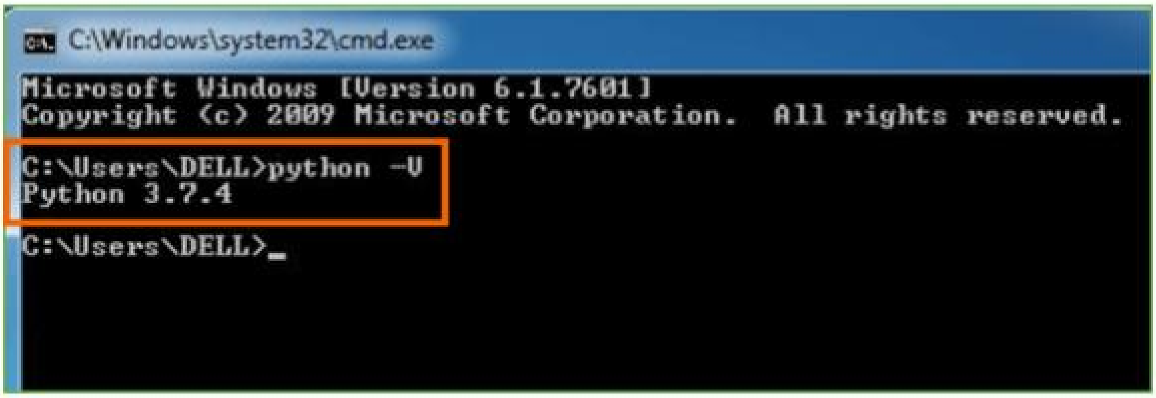
Step 1: Click on Start

Step 2: In the Windows Run Command, type "cmd".



Step 3: Open the Command prompt option.

Step 4: Let us test whether the python is correctly installed. Type python –V and press Enter.



```
C:\Windows\system32\cmd.exe
Microsoft Windows [Version 6.1.7601]
Copyright (c) 2009 Microsoft Corporation.  All rights reserved.

C:\Users\DELL>python -V
Python 3.7.4

C:\Users\DELL>_
```

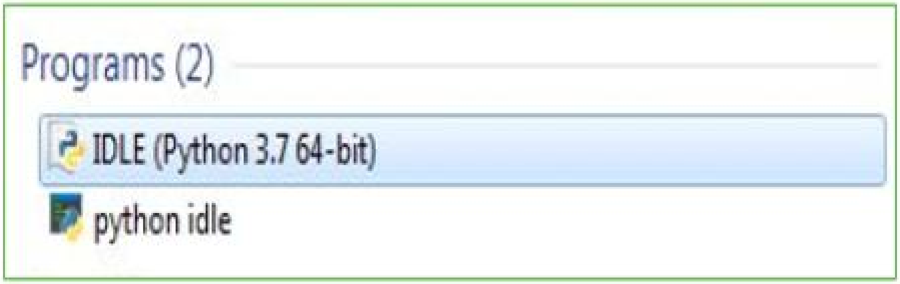
Step 5: You will get the answer as 3.7.4

Note: If you have any of the earlier versions of Python already installed. You must first uninstall the earlier version and then install the new one.

Check how the Python IDLE works

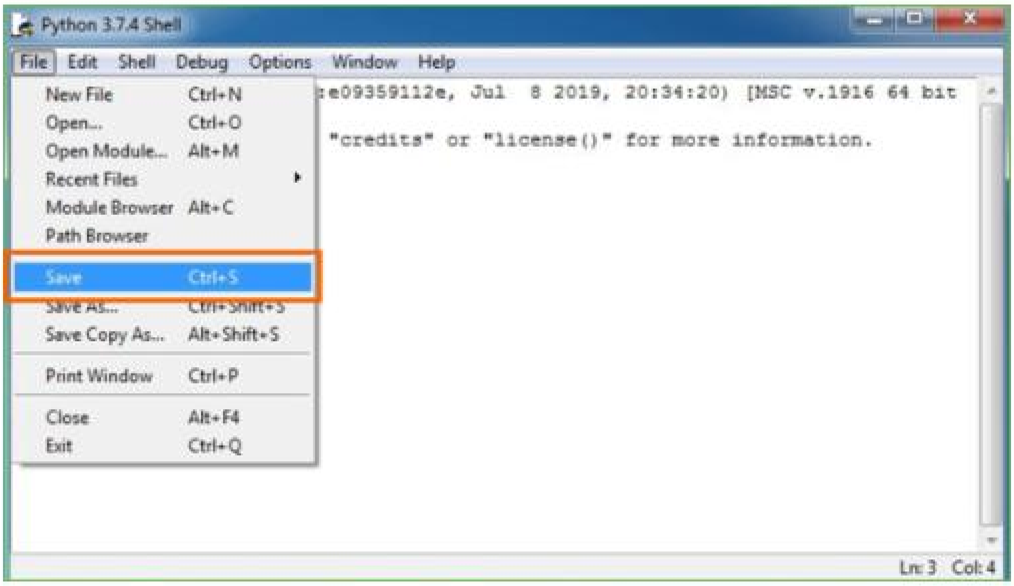
Step 1: Click on Start

Step 2: In the Windows Run command, type "python idle".



Step 3: Click on IDLE (Python 3.7 64-bit) and launch the program

Step 4: To go ahead with working in IDLE you must first save the file. Click on File > Click on Save



```
:e09359112e, Jul  8 2019, 20:34:20) [MSC v.1916 64 bit
"credits" or "license()" for more information.
```

Step 5: Name the file and save as type should be Python files. Click on SAVE. Here I have named the files as Hey World.

Step 6: Now for e.g. enter print

Haar –Cascades

Haar-like feature sore rectangular patterns in data. A cascade is a series of “Haar-like features” that are combined to form a classifier. A Haar wavelet is a mathematical function that produces square wave output.

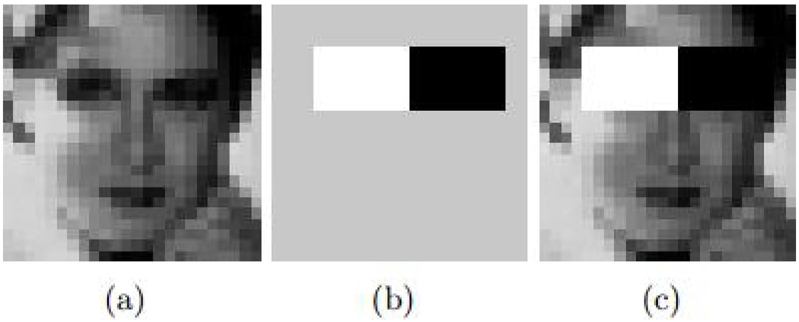


Figure 2-2. Haar like Features [13]

Figure 2.2 shows Haar like features, the background of a template like (b) is painted gray to highlight the pattern’s support. Only those pixels marked in black or white are used when the corresponding feature is calculated.

Since no objective distribution can describe the actual prior probability for a given image to have a face, the algorithm must minimize both the false negative and false positive rates in order to achieve an acceptable performance. This then requires an accurate numerical description of what sets human faces apart from other objects. Characteristics that define a face can be extracted from the images with a remarkable committee learning algorithm called Ada boost. Ada boost (Adaptive boost) relies on a committee of weak classifiers that combine to form a strong one through a voting mechanism. A classifier is weak if, in general, it cannot meet a predefined classification target in error terms. The operational algorithm to be used must also work with a reasonable computational budget. Such techniques as the integral image and attention cascades have made the Viola-Jones algorithm highly efficient: fed with a real time image sequence generated from a standard webcam or camera, it performs well on a standard PC.

The size and position of a pattern’s support can vary provided its black and white rectangles have the same dimension, border each other and keep their relative positions. Thanks to this constraint, the number of features one can draw an image is somewhat manageable: a 24 × 24 image, for instance, has 43200, 27600, 43200, 27600 and 20736 features of category (a), (b), (c), (d) and (e) respectively as shown in figure 2.3, hence 162336 features in all. In practice, five patterns are considered. The derived features are assumed to hold all the information needed to characterize a face. Since faces are large and regular by nature, the use of Haar-like patterns.

How The HAAR – Like Features Work

A scale is chosen for the features say 24 × 24 pixels. This is then slid across the image. The average pixel values under the white are a and the black area are then computed.

If the difference between the areas is above some threshold, then the feature matches.

In face detection, since the eyes are of different color tone from the nose, the Haar feature

(b) from Figure 3.3 can be scaled to fit that area as shown below,

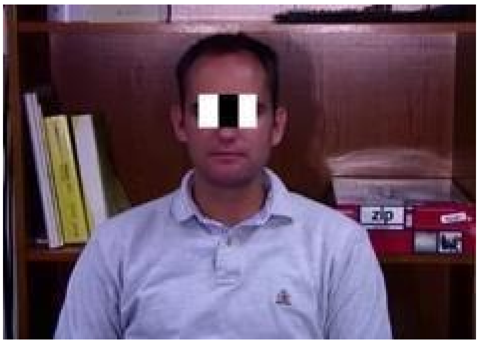


Figure 3-4. How the Haar like feature of figure 3.3 can be used to scale the eyes

One Haar feature is however not enough as there are several features that could

match it (like the zip drive and white are as at the background of the image of figure3.4 it

is called a "weak classifier." Haar cascades, the basis of Viola Jones detection framework

Therefore, consist of a series of weak classifiers whose accuracy is at least50%correct. If

an area passes a single classifier, it moves to the next weak classifier and so on, otherwise,

the area does not match.

Cascaded Classifier:

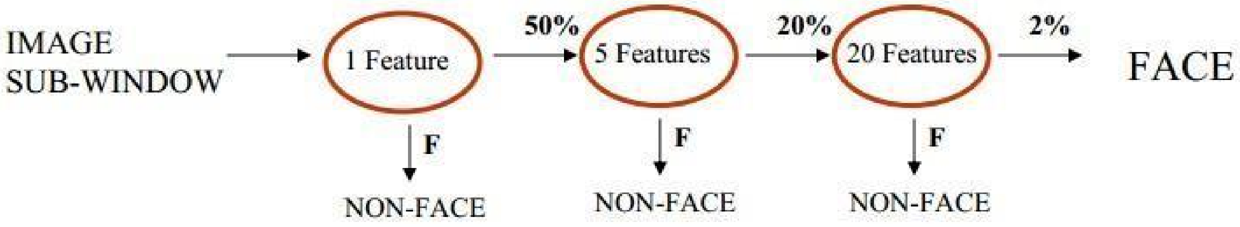


Figure 3-5. several classifiers combined to enhance face detection

From figure 3.5, a 1 feature classifier achieves 100% face detection rate

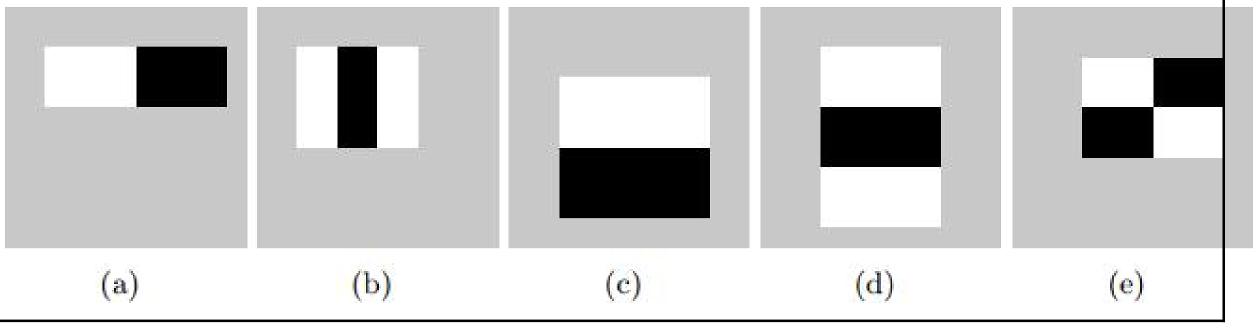
and about 50% false positive rate. A 5-feature classifier achieves 100% detection rate and 40% false positive rate (20% cumulative). A 20-feature classifier achieves 100% detection rate with 10%false positive rate(2%cumulative). Combining several weak classifiers improves the accuracy of detection.

A training algorithm called Ada boost, short for adaptive boosting, which had no application before Haar cascades, was utilized to combine a series of weak classifiers into a strong classifier. Ada boost tries out multiple weak classifiers over several rounds, selecting the best weak classifier in each round and combining the best weak classifier to create a strong classifier. Ada boost can use classifiers .

Minimum Neighbors Threshold:

The minimum neighbor's threshold sets the cutoff level for discarding or keeping rectangle groups as either faces or not. This is based on the number of raw detections in the group and its values ranges from zero to four.

When the face detector is called behind the scenes, each positive face region generates many hits from the Haar detector as in Figure3.8. The face region itself generates a large cluster of rectangles that to a large extend overlap. The detections are usually false detections and are discarded. The multiple face region detections are then merged in to a single detection. The face detection function does all this before returning the list of the detected faces. The merge step groups rectangles that contain a large number of overlaps and then finds the average rectangle for the group. It then replaces all the rectangles in the group with the average rectangle.



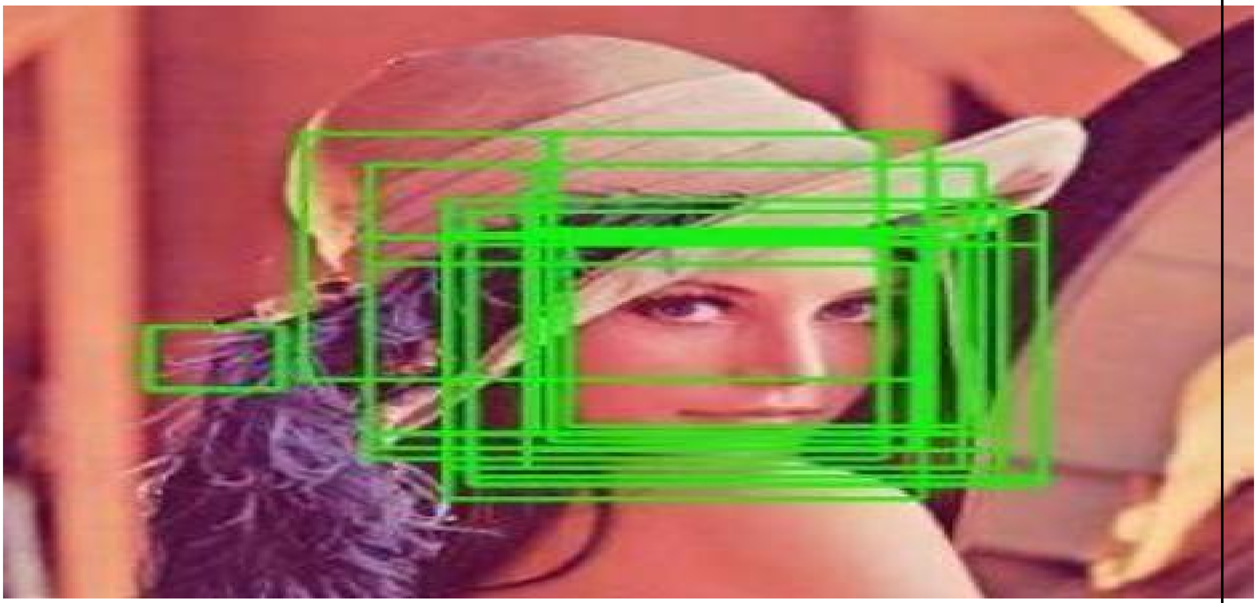
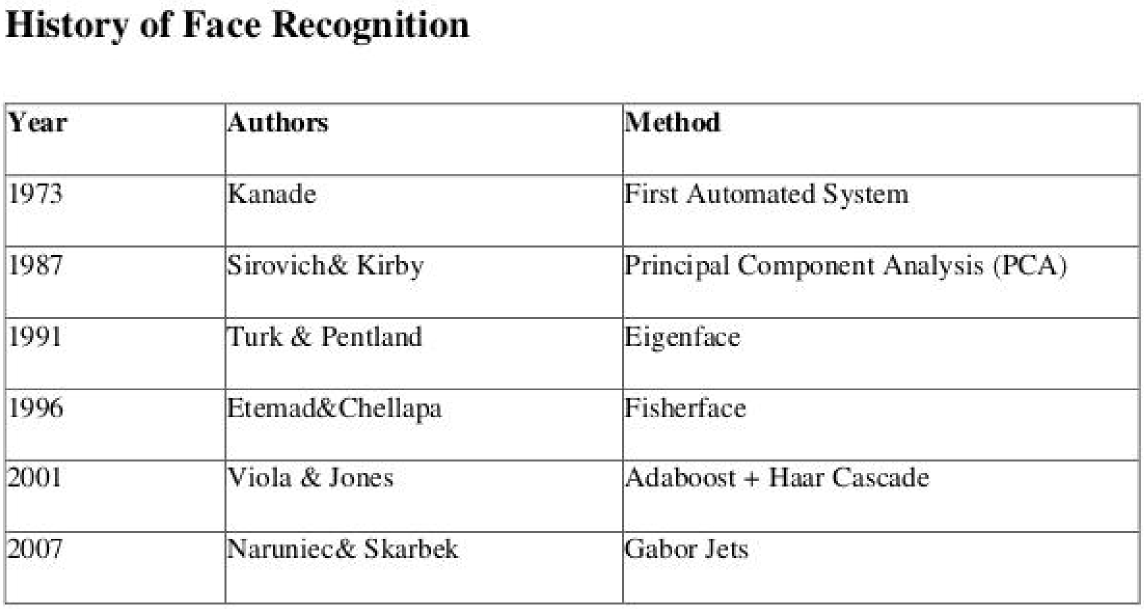


Figure 3-8. Lena’s image showing the list of rectangles



History of Face Recognition

Year	Authors	Method
1973	Kanade	First Automated System
1987	Sirovich& Kirby	Principal Component Analysis (PCA)
1991	Turk & Pentland	Eigenface
1996	Etemad&Chellapa	Fisherface
2001	Viola & Jones	Adaboost + Haar Cascade
2007	Naruniec& Skarbek	Gabor Jets

Table 3.1 A table showing the brief history of the existing face recognition techniques

Takeo Kanade is a Japanese computer scientist and one of the world's for most researchers in computer vision came up with a program which extracted face feature points (such as nose, eyes, ears and mouth) on photographs. These were then compared to reference data

A major milestone that reinvigorated research was the PCA method by Sirovich and Kirby in 1987.The Principal Component Analysis is a standard linear algebra technique, to the face recognition problem,which showed that less than one hundred values were required to accurately code a suitably aligned and normalized face image.Turk and Pentland discovered that while using the Eigen faces technique, the residual error could be used to detect faces in images, a discovery that enabled reliable, real time automatic face recognition systems Although this approach was somehow constrained by environmental factors, it nonetheless created significant interest in furthering development of automated face recognition techniques.

The viola Jones Adaboost and Haar cascade method brought together new algorithms and insights to construct a framework for robust and extremely rapid visual detection. This system was most clearly distinguished from previous approaches in its ability to detect faces extremely rapidly. Operating on 384 y 288-pixel images, faces were detected at 15 frames per second on a 700MHz Intel Pentium 3 Processor. All identification or authentication technologies operate using the following four stages:

- Capture: A physical or behavioral sample is captured by the system during
- Enrollment and also in identification or verification process.
- Extraction: unique data is extracted from the sample and a template is created.
- Comparison: the template is then compared with a new sample.
- Match/non-match: the system decides if the features extracted from the new Samples are a match or anon-match.

Face Recognition Concepts

Although different approaches have been tried by several groups of people across

the world to solve the problem of face recognition, no particular technique has been discovered that yields satisfactory results in all circumstances to solve the problem of face recognition, no particular technique has been discovered that yields satisfactory results in all circumstances.

The different approaches of face recognition for still images can be categorized in to three

main groups namely:

- Holistic Approach – In this, the whole face region is taken as an input in

face detection system to perform face recognition.

- Feature-based Approach – where the local features on the face such as the noise

and eyes are segmented and then fed to the face detection system to ease the

task of face recognition.

- Hybrid Approach – In hybrid approach, both the local features and the whole face

are used as input to the detection system, this approach is more similar to the

behavior of human beings in recognizing faces.

There are two main types of face Recognition Algorithms:

- Geometric – this algorithm focuses at distinguishing features of a face.
- Photometric – a statistical approach that distills an image into values and comparing the values

The Most Popular algorithms are ;

1. Principal Component Analysis based Eigenfaces.

2. Linear Discriminate Analysis.

3. Elastic Bunch Graph Matching using the fisher face algorithm.

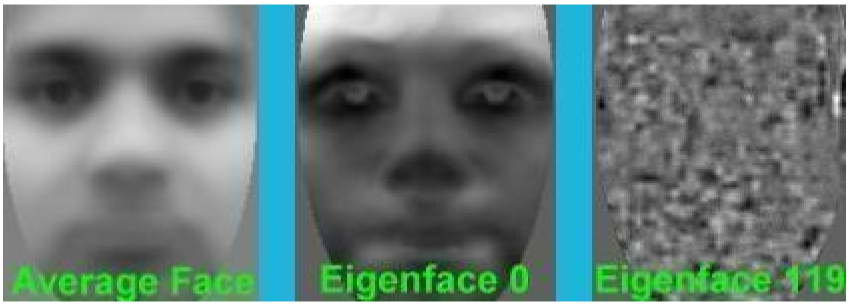
4. The Hidden Markov Model

5. Neuronal Motivated Dynamic Link Matching.

It should however be noted that the existing face recognition techniques are not one hundred percent (100%) efficient just yet. Typical efficiencies range between 40% to 60%.

The computer-based facial recognition industry has made many useful advancements in the past decade; however, the need for higher accuracy remains. Through the determination and commitment of industry, government evaluations, and organized standards bodies, growth and progress will continue, raising the bar for face-recognition technology. The computer-based facial recognition industry has made many useful advancements in the past decade; however, the need for higher accuracy remains. Through the determination and commitment of industry, government evaluations, and organized standards bodies, growth and progress will continue, raising the bar for face-recognition technology

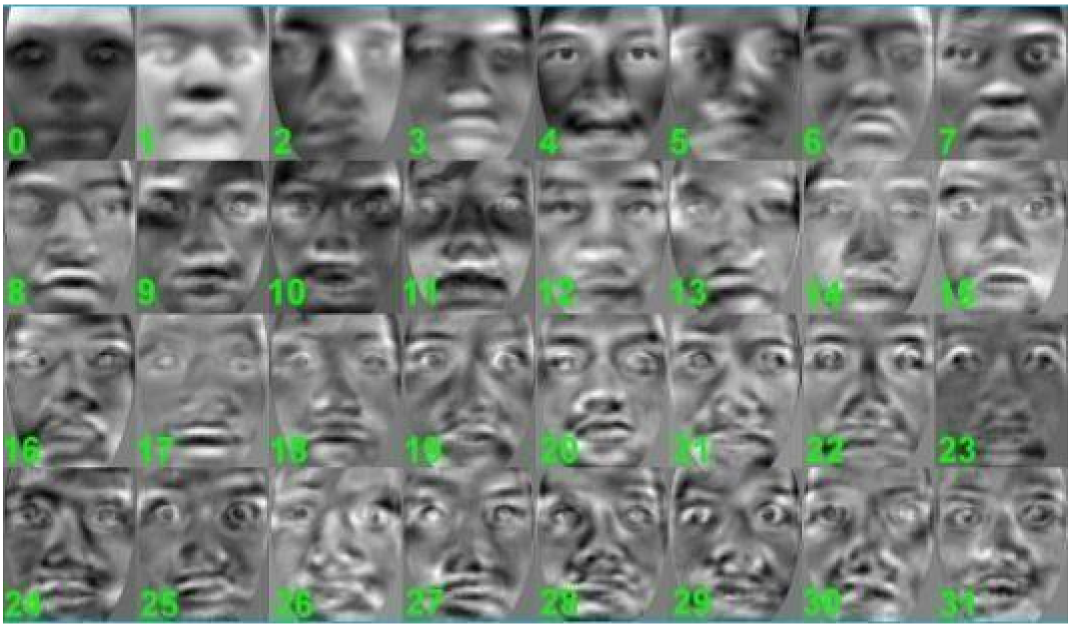
To reduce the calculations needed for finding these eigenfaces, the dimensionality of the original training set is reduced before they are calculated. Since eigenfaces sow the "directions" of data and each preceding eigenface shows less "directions" and more "noise", only few first eigenface (say K) are selected whereas the last of the eigenfaces are discarded .



The average face, the first and last eigenface which is mainly noise

shows the average face and the first and last eigenfaces that were generated from a collection of 30 images each of 4 people. The average face shows the smooth face structure of a generic person, the first few eigen faces will show some dominant features of faces, and the last eigenfaces (e.g.:Eigenface 119) are mainly image noise.

shows the first 32 eigenfaces. Images 32 - 119 are discarded because they are mainly no.



The dominant 32 eigenfaces representing all the images in the set [11]

Working Principle of PCA Eigenface for Face Recognition.

Given a training set of M images as shown in the figure and an unknown face all

of the same size, PCA Eigen face method aims at representing the face image as a linear

combination of a set of Eigenfaces/Eigenvectors as in figure3.11 These Eigenfaces (eigenvectors) are in fact the principle components of the training set of face images generated after reducing the dimensionality of the training set. Once Eigenfaces are selected, each training set image is represented in terms of these eigenfaces. When an unknown face comes for recognition, it is also represented in terms of the selected eigenfaces. The eigenface representation of the unknown face is compared with that of each training set face image. Once Eigenfaces are selected, each training set image is represented in terms of these eigenfaces. When an unknown face comes for recognition, it is also represented in terms of the selected eigenfaces Once Eigenfaces are selected, each training set image is represented in terms of these eigenfaces. When an unknown face comes for recognition, it is also represented in terms of the selected eigenfaces Once Eigenfaces are selected, each training set image is represented in terms of these eigenfaces. When an unknown face comes for recognition, it is also represented in terms of the selected eigenfaces The distance between them is calculated. If the distance is above some specified threshold, then it recognizes the unknown face as that person. PCA Eigenfaces method considers each pixel in an image as a separate dimension.

Eg:

A 50 * 50 image = 2500 pixels thus has 2500 dimensions. This method does not work on images directly, it first converts them to a matrix (vector) form.

PCA Face Recognition Algorithm:

Step1: Acquire training set of ‘N’ number of images at the initial stage. In this project the images are of 92*112 pixels each. Training set is shown in the figure.

Step2: Calculation of the eigenface from the “N” training set images keeping only few M images that is correspond to that of the highest Eigen values. The M images describe the “face space”. When new faces encountered, the “eigenfaces” can be recalculated accordingly.

Step3: The corresponding distribution of the “M” dimensional weight space for every known individual is **Calculated** by projecting their respective face images onto “face space”.

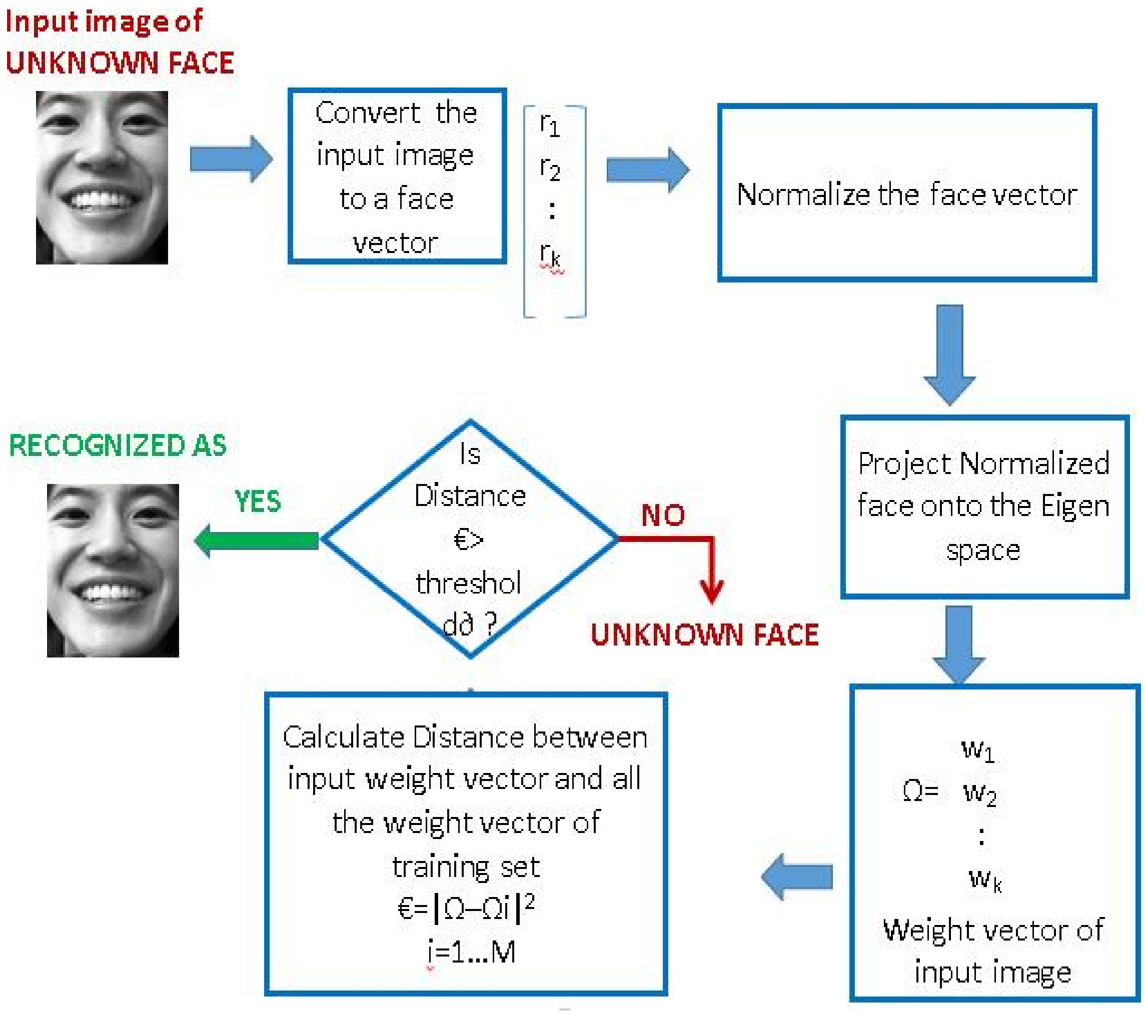
Step 4: Compute set of the weights anticipating or projecting the data picture or input image to M “Eigen faces”.

Step5: **Determine** if the given image is face image or not by checking to the closeness of given image or picture to "face space".

Step 6: If the **image** is sufficiently close enough, then classify the weight pattern as either an unknown or as a known person based on measured Euclidean distance.

Step7: If the image is sufficiently close enough then refer to the recognition is successful and give applicable information about recognized face from the **database** which hold data of faces.

The face recognition algorithm flow chart of Figure 3.13 is basically a method of checking which training image is most similar to the input image, out of the whole training


Input image of
UNKNOWN FACE
Convert the input image to a face vector
r1
r2
:
rk
Normalize the face vector
Project Normalized face onto the Eigen space
Ω= w1 w2 : wk
Weight vector of input image
Calculate Distance between input weight vector and all the weight vector of training set
€=|Ω–Ωi|²
i=1...M
Is Distance €> threshol dð ?
NO
UNKNOWN FACE
YES
RECOGNIZED AS

Training Set:

A faces database also known as a training set in machine learning is a collection of faces of people or subjects that your system is built to recognize. The face images are obtained from multiple snaps of people who the system has to recognize such that each person's set of snaps covers all possible facial expressions, posture and light conditions that may be possible at recognition time. The camera should be able to produce high quality pictures to be stored in the database. A faces database can be implemented as a folder in Windows or as a table Database Management System (DBMS) like MS Access.

Limitations of a Faces Database implementation as a Folder

- Not secure as faces are visible, one could delete them
- Difficult to associate labels with the proper images.

Advantages of Folder Implementation

- Simple code to access data
- It is faster to manage
- No database design
- Dependency

Benefits of using MS Access:

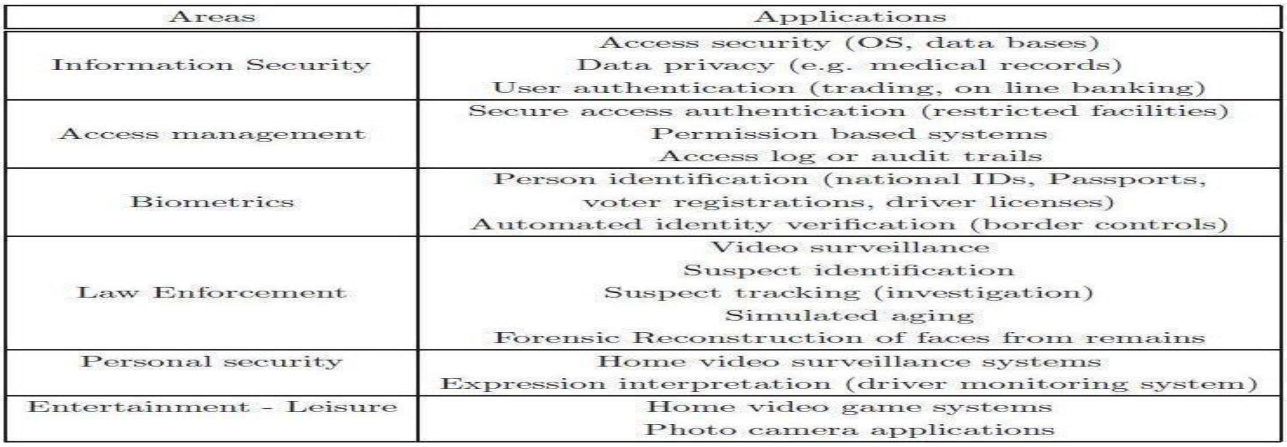
- Faster in performance
- Powerful as a proper database as various fields such as Name, Registration number, Year of Birth etc. can be linked in tables.
- System becomes design dependent i.e., reads columns in the proper order.

Logical Steps to store Faces to a Database:

- Detect and extract a face
- Label the extracted face
- Insert the extracted face label pair to the faces database

Applications:

- System becomes design dependent i.e., re



Areas	Applications
Information Security	Access security (OS, data bases) Data privacy (e.g. medical records) User authentication (trading, on line banking)
Access management	Secure access authentication (restricted facilities) Permission based systems Access log or audit trails
Biometrics	Person identification (national IDs, Passports, voter registrations, driver licenses) Automated identity verification (border controls)
Law Enforcement	Video surveillance Suspect identification Suspect tracking (investigation) Simulated aging Forensic Reconstruction of faces from remains
Personal security	Home video surveillance systems Expression interpretation (driver monitoring system)
Entertainment - Leisure	Home video game systems Photo camera applications

CHAPTER-6

SYSTEM IMPLEMENTATION

6.1 MODULES

Image Capture Module

Functionality: Captures images of individuals entering the designated area (classroom, office, etc.).

Components:

- Camera integration
- Real-time image capture
- Handling multiple faces in a single frame

Face Recognition Module

Functionality: Recognizes and matches detected faces against a pre-existing database.

Components:

- Feature extraction using deep learning models (e.g., CNNs)
- Face encoding and matching
- Handling variations in lighting, angle, and expressions

Database Management Module

Functionality: Manages the database of enrolled users and attendance records.

Components:

- User data storage (images, personal details)
- Attendance records (timestamp, presence status)
- Database operations (CRUD - Create, Read, Update, Delete)

Logging and Monitoring Module

Functionality: Logs system activities and monitors performance.

Components:

- Activity logs (user actions, system events)
- Performance metrics (response times, accuracy rates)
- Error handling and alerting mechanisms

6.2 SAMPLE CODE

```
import cv2
import numpy as np
import face_recognition
import os
from datetime import datetime

# Define path to dataset
path = '/home/lohith/Desktop/face_recognition/dataset'
images = []
imgLabels = []
myList = os.listdir(path)
print(myList)
# Load images and their labels
for cl in myList:
curImg = cv2.imread(os.path.join(path, cl))
images.append(curImg)
imgLabels.append(os.path.splitext(cl)[0])
print(imgLabels)
def findEncodings(images):
encodList = []
for img in images:
```

```
img = cv2.cvtColor(img, cv2.COLOR_BGR2RGB)

encode = face_recognition.face_encodings(img)

if len(encode) > 0:

encodList.append(encode[0])

return encodList

# Get encodings of known faces

encoded_faces = findEncodings(images)

print('Encoding completed')

def markAttendance(name):

with open('/home/lohith/Desktop/face_recognition/attendance1.csv', 'r+') as f:

myDataList = f.readlines()

nameList = [line.split(',')[0] for line in myDataList]

if name not in nameList:

now = datetime.now()

dtString = now.strftime('%H:%M:%S')

f.writelines(f'\n{name},{dtString}')

else:

# Update attendance time if already in the list

now = datetime.now()

dtString = now.strftime('%H:%M:%S')

updatedList = [f'{name},{dtString}' if line.startswith(name) else line.strip() for line in myDataList]

f.seek(0)

f.writelines('\n'.join(updatedList) + '\n')

FACE_DISTANCE_THRESHOLD = 0.5

# Start webcam for face recognition

webcam = cv2.VideoCapture(0)

while True:
```

```
success, img = webcam.read()

imgS = cv2.resize(img, (0, 0), None, 0.25, 0.25)

imgS = cv2.cvtColor(imgS, cv2.COLOR_BGR2RGB)

faceCurFrame = face_recognition.face_locations(imgS)

encodeCurFrame = face_recognition.face_encodings(imgS, faceCurFrame)

for encodeFace, faceLocation in zip(encodeCurFrame, faceCurFrame):

matches = face_recognition.compare_faces(encoded_faces, encodeFace,
tolerance=FACE_DISTANCE_THRESHOLD)

faceDis = face_recognition.face_distance(encoded_faces, encodeFace)

matchIndex = np.argmin(faceDis)

if matches[matchIndex] and faceDis[matchIndex] < FACE_DISTANCE_THRESHOLD:

name = imgLabels[matchIndex].upper()

print(name)

y1, x2, y2, x1 = faceLocation

y1, x2, y2, x1 = y1 * 4, x2 * 4, y2 * 4, x1 * 4

cv2.rectangle(img, (x1, y1), (x2, y2), (0, 255, 0), 3)

cv2.rectangle(img, (x1, y2 - 35), (x2, y2), (0, 255, 0), cv2.FILLED)

cv2.putText(img, name, (x1 + 6, y2 - 6), cv2.FONT_HERSHEY_COMPLEX, 1, (255, 255, 255), 2)

markAttendance(name)

else:

print("Unknown face detected")

y1, x2, y2, x1 = faceLocation

y1, x2, y2, x1 = y1 * 4, x2 * 4, y2 * 4, x1 * 4

cv2.rectangle(img, (x1, y1), (x2, y2), (0, 0, 255), 3)

cv2.rectangle(img, (x1, y2 - 35), (x2, y2), (0, 0, 255), cv2.FILLED)

cv2.putText(img, "Unknown", (x1 + 6, y2 - 6), cv2.FONT_HERSHEY_COMPLEX, 1, (255, 255, 255), 2)

cv2.imshow('Frame', img)
```

```
if cv2.waitKey(1) & 0xFF == ord('q'):

break

webcam.release()

cv2.destroyAllWindows()
```

6.3 SYSTEM TESTING

The purpose of testing is to discover errors. Testing is the process of trying to discover every conceivable fault or weakness in a work product. It provides a way to check the functionality of components, subassemblies, assemblies and/or a finished product It is the process of exercising software with the intent of ensuring that the Testing is the process of trying to discover every conceivable fault or weakness in a work product.

It provides a way to check the functionality of components, subassemblies, assemblies and/or a finished product It is the process of exercising software with the intent of ensuring that the Software system meets its requirements and user expectations and does not fail in an unacceptable manner. There are various types of tests. Each test type addresses a specific testing requirement. Testing is the process of trying to discover every conceivable fault or weakness in a work product. It provides a way to check the functionality of components, subassemblies, assemblies and/or a finished product It is the process of exercising software with the intent of ensuring that the test case are executed .

6.4 TYPES OF TESTS

1. Unit testing:

Unit testing involves the design of test cases that validate that the internal program logic is functioning properly, and that program inputs produce valid outputs. All decision branches and internal code flow should be validated. It is the testing of individual software units of the application .it is done after the completion of an individual unit before integration. This is a structural testing, that relies on knowledge of its construction and is invasive. Unit tests perform basic tests at component level and test a specificbusiness process, application, and/or system configuration. Unit tests ensure that each unique path of a business process performs accurately to the documented specifications and contains clearly defined inputs and expected results.

2.Integration testing:

Integration tests are designed to test integrated software components to determine if they actually run as one program. Testing is event driven and is more concerned with the basic outcome of screens or fields. Integration tests demonstrate that although the components were individually satisfaction, as shown by successfully unit testing, the combination of components is correct and consistent. Integration testing is specifically aimed at exposing the problems that arise from the combination of components.

3.Functional test:

Functional tests provide systematic demonstrations that functions tested are available as specified by the business and technical requirements, system documentation, and user manuals.

Functional testing is centered on the following items:

Valid Input : identified classes of valid input must be accepted.

Invalid Input : identified classes of invalid input must be rejected.

Functions : identified functions must be exercised.

Output : identified classes of application outputs must be exercised.

Systems/Procedures: interfacing systems or procedures must be invoked. Organization and preparation of functional tests is focused on requirements, key functions, or special test cases. In addition, systematic coverage pertaining to identify Business process flows; data fields, predefined processes, and successive processes must be considered for testing. Before functional testing is complete, additional tests are identified and the effective value of current tests is determined.

Systems/Procedures: interfacing systems or procedures must be invoked. Organization and preparationof functional tests is focused on requirements, key functions, or special test cases. In addition, systematic coverage pertaining to identify Business process flows; data fields, predefined processes, and successive processes must be considered for testing. Before functional testing is complete, additional tests are identified and the effective value of current tests is determined

6.5 TESTING METHODOLOGIES

1 Unit Testing.

Unit testing involves the design of test cases that validate that the internal program logic is functioning properly, and that program inputs produce valid outputs. All decision branches and internal code flow should be validated. It is the testing of individual software units of the application .it is done after the completion of an individual unit before integration. This is a structural testing, that relies on knowledge of its construction and is invasive. Unit tests perform basic tests at component level and test a specific business process, application, and/or system configuration. Unit tests ensure that each unique path of a business process performs accurately to the documented specifications and contains clearly defined inputs and expected results.

2 Integration Testing.

Integration tests are designed to test integrated software components to determine if they actually run as one program. Testing is event driven and is more concerned with the basic outcome of screens or fields. Integration tests demonstrate that although the components were individually satisfaction, as shown by successfully unit testing, the combination of components is correct and consistent. Integration testing is specifically aimed at exposing the problems that arise from the combination of components.

3 Functional Testing.

Functional tests provide systematic demonstrations that functions tested are available as specified by the business and technical requirements, system documentation, and user manuals.

Functional testing is centered on the following items:

Valid Input	: identified classes of valid input must be accepted.
Invalid Input	: identified classes of invalid input must be rejected.
Functions	: identified functions must be exercised.
Output	: identified classes of application outputs must be exercised.
Systems/Procedures	: interfacing systems or procedures must be invoked.

Organization and preparation of functional tests is focused on requirements, key functions, or special test cases. In addition, systematic coverage pertaining to identify Business process flows; data fields, predefined processes, and successive processes must be considered for testing. Before functional testing is complete, additional tests are identified and the effective value of current tests is determined.

4 SYSTEM TEST

System testing ensures that the entire integrated software system meets requirements. It tests aconfiguration to ensure known and predictable results. An example of system testing is the configuration-oriented system integration test. System testing is based on process descriptions and flows, emphasizing pre-driven process links and integration points.

5 white Box Testing:

White Box Testing is a testing in which in which the software tester has knowledge of the inner workings, structure and language of the software, or at least its purpose. It is purpose. It is used to test areas that cannot be reached from a black box level. Software integration testing is the incremental integration testing of two or more integrated software components on a single platform to produce failures caused by interface defects. The task of the integration test is to check that components or software applications, e.g., components in a software system or – one step up – software applications at the company level – interact without error. The task of the integration test is to check that components or software applications, e.g., components in a software system or – one step up – software applications at the company level – interact without error The task of the integration test is to check that components or software applications, e.g., components in a software system or – one step up – software applications at the company level – interact without error.

6 Black Box Testing:

Black Box Testing is testing the software without any knowledge of the inner workings, structure or language of the module being tested. Black box tests, as most other kinds of tests, must be written from a definitive source document, such as specification or requirements document, such as specification or requirements document. It is a testing in which the software under test is treated, as a black box .you cannot “see” into it. The test provides inputs and responds to outputs without considering how the software works.

7 Unit Testing:

Unit testing is usually conducted as part of a combined code and unit test phase of the software lifecycle, although it is not uncommon for coding and unit testing to be conducted as two distinct phases. Unit testing is usually conducted as part of a combined code and unit test phase of the software lifecycle, although it is not uncommon for coding and unit testing to be conducted as two distinct phases.

8 Integration Testing

Software integration testing is the incremental integration testing of two or more integrated software components on a single platform to produce failures caused by interface defects.

The task of the integration test is to check that components or software applications, e.g. components in a software system or – one step up – software applications at the company level – interact without error.

Test Results: All the test cases mentioned above passed successfully. No defects encountered.

9 Acceptance Testing

User Acceptance Testing is a critical phase of any project and requires significant participation by the end user. It also ensures that the system meets the functional requirements.

Test Results: All the test cases mentioned above passed successfully. No defects encountered.

To study the car pooling problem, we formulate the problem to match passengers with drivers in this paper. As the car pooling problem formulated in this paper above is not a standard matching problem in computer science, existing algorithms for matching problems cannot be applied directly.

Instead of applying existing algorithms for matching problems, we have proposed

a heuristic method to solve the above mentioned problem. In our approach, we collect the requests of passengers and the trajectory data

it is a testing in which the software under test is treated, as a black box .you cannot “see” into it. The test provides inputs and responds to outputs without considering how the software works.

CHAPTER-7

SCREEN SHOTS

Web attendance is a one of the efficent way to mark the attendance , its implimentation and deployment on the project work with the connection of both the machine learning and deep learning librarys .it process image the help of face_recognition and opencv to open the web came and encoding the images from the data set with the help of haar - cascades and PCA .

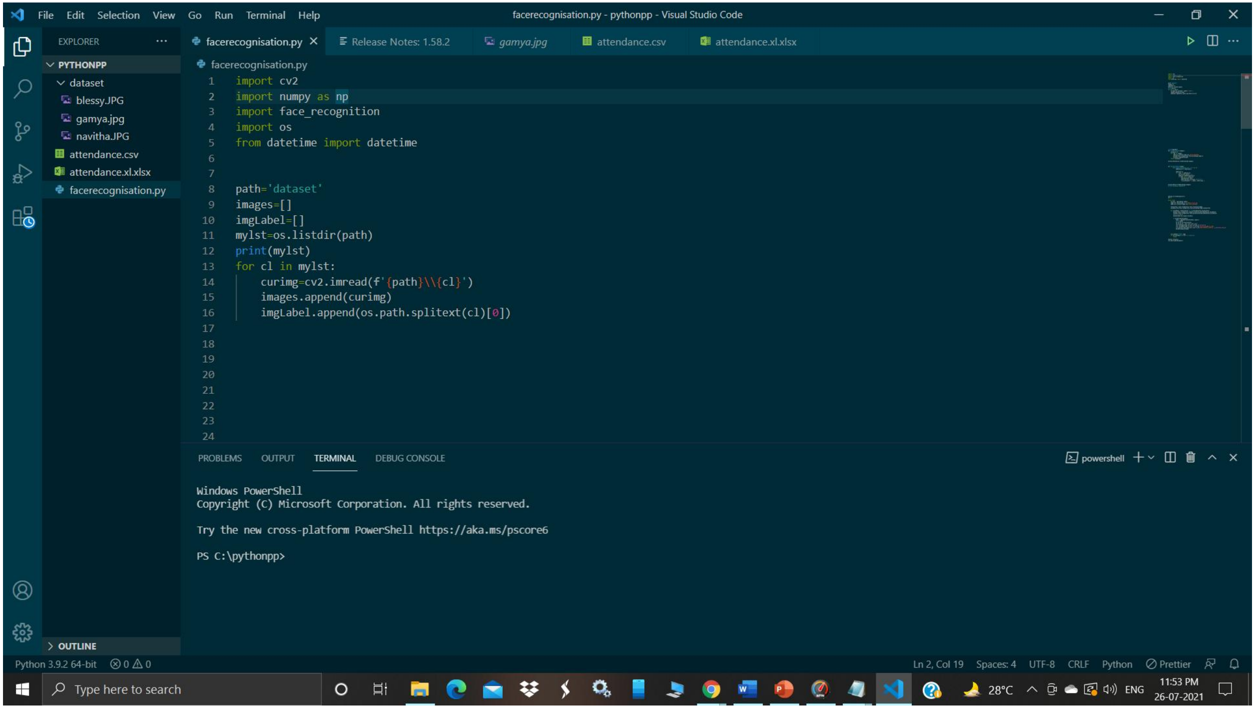
To overcome from above issues we are employing Blockchain based carpooling services where Blockchain support distributed data access which means data can be access and stored at multiple node and if one node then services will be access from other working node. Blockchain has internal support for data verification which means Blockchain store each data as block/transaction and associate each block with unique hashcode and before storing any new data then Blockchain verify hashcode of all nodes and if data tamper at any data then it will result into different hashcode and verification will be failed and data tamper will be detected.

To store and access data from Blockchain we need to make use of smart contract by using SOLIDITY code. This code contains functions to store and access data from Blockchain. Below screen showing Smart contract code.

Automatic attendance using machine learning The detected face is extracted and compared with those in the database. Upon a successful match, the name associated with the face is then displayed on the upper edge of the rectangular box. The number of faces in the scene as well as their corresponding names areal so on the highlighted boxnumber2. TheFaceAdderbox3canalso be used to add faces to the database

The Project Interface

First, we should run the code from our Python CV 2 software to open our Attendance System. The interface of our system is as below.



File Edit Selection View Go Run Terminal Help
facerecognisation.py - pythonpp - Visual Studio Code
EXPLORER
PYTHONPP
dataset
blessy.JPG
gamya.jpg
navitha.JPG
attendance.csv
attendance.xl.xlsx
facerecognisation.py
facerecognisation.py
Release Notes: 1.58.2
gamya.jpg
attendance.csv
attendance.xl.xlsx
PROBLEMS OUTPUT TERMINAL DEBUG CONSOLE
powershell
OUTLINE
Python 3.9.2 64-bit ⊗ 0 ⚠ 0
Ln 2, Col 19 Spaces: 4 UTF-8 CRLF Python Prettier
Type here to search
28°C ENG 11:53 PM 26-07-2021

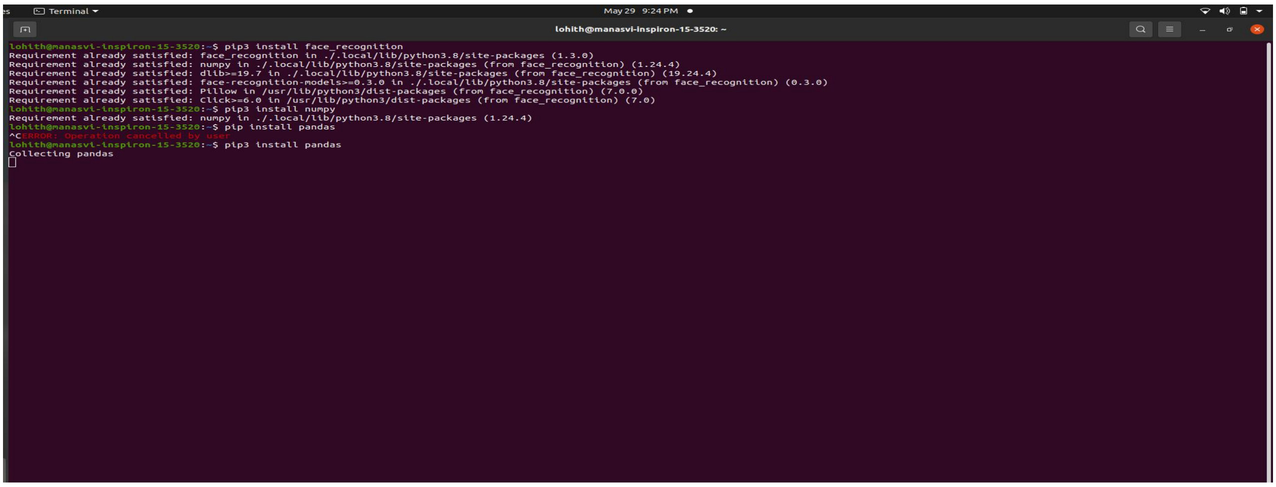
```python
import cv2
import numpy as np
import face_recognition
import os
from datetime import datetime


path='dataset'
images=[]
imgLabel=[]
mylst=os.listdir(path)
print(mylst)
for cl in mylst:
    curimg=cv2.imread(f'{path}\\{cl}')
    images.append(curimg)
    imgLabel.append(os.path.splitext(cl)[0])
```

```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6

PS C:\pythonpp>
```

If we observe the above picture of our program interface, we have to run the code here.

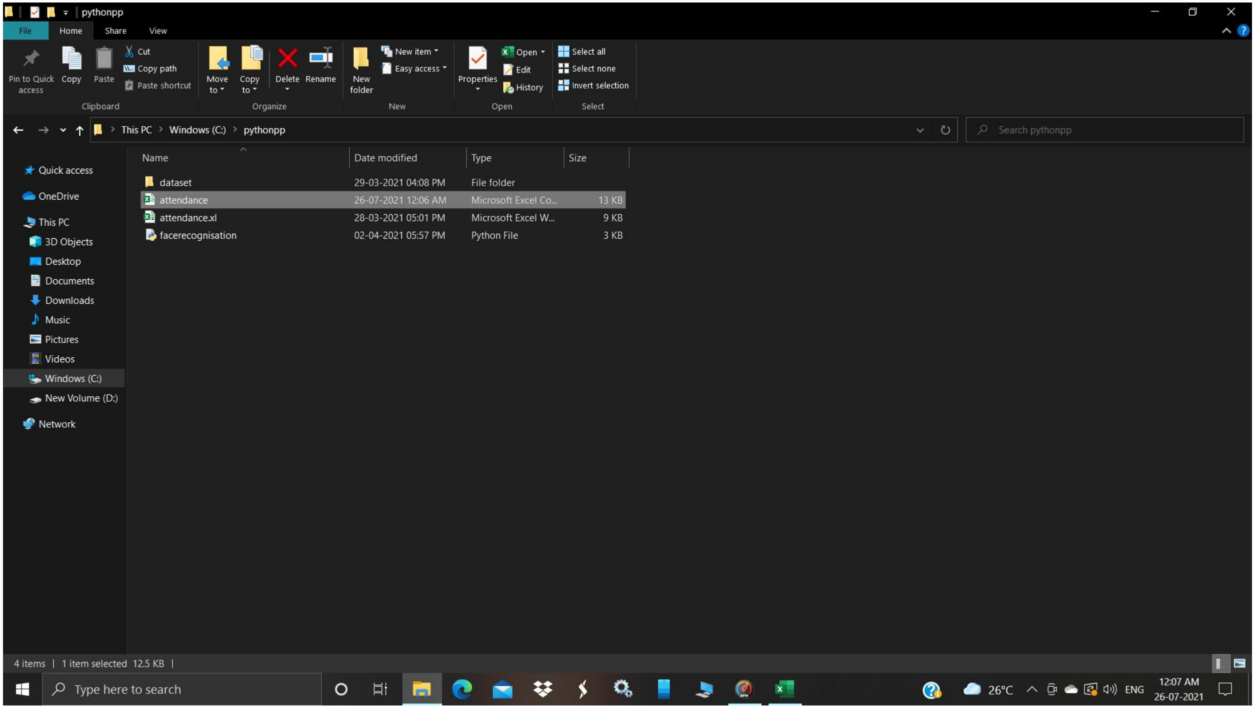
1) First go to the face_recognition file and open the the face_recognition(2).py file at the desktop

2) After opening the code in visual studio we have to import the librarys in shell terminal or command prompt



```
lohith@manasvi-inspiron-15-3520:~$ pip3 install face_recognition
Requirement already satisfied: face_recognition in ./.local/lib/python3.8/site-packages (1.3.0)
Requirement already satisfied: numpy in ./.local/lib/python3.8/site-packages (from face_recognition) (1.24.4)
Requirement already satisfied: dlib>=19.7 in ./.local/lib/python3.8/site-packages (from face_recognition) (19.24.4)
Requirement already satisfied: face-recognition-models>=0.3.0 in ./.local/lib/python3.8/site-packages (from face_recognition) (0.3.0)
Requirement already satisfied: Pillow in /usr/lib/python3/dist-packages (from face_recognition) (7.0.0)
Requirement already satisfied: Click>=6.0 in /usr/lib/python3/dist-packages (from face_recognition) (7.0)
lohith@manasvi-inspiron-15-3520:~$ pip3 install numpy
Requirement already satisfied: numpy in ./.local/lib/python3.8/site-packages (1.24.4)
lohith@manasvi-inspiron-15-3520:~$ pip install pandas
^CERROR: Operation cancelled by user
lohith@manasvi-inspiron-15-3520:~$ pip3 install pandas
Collecting pandas
```

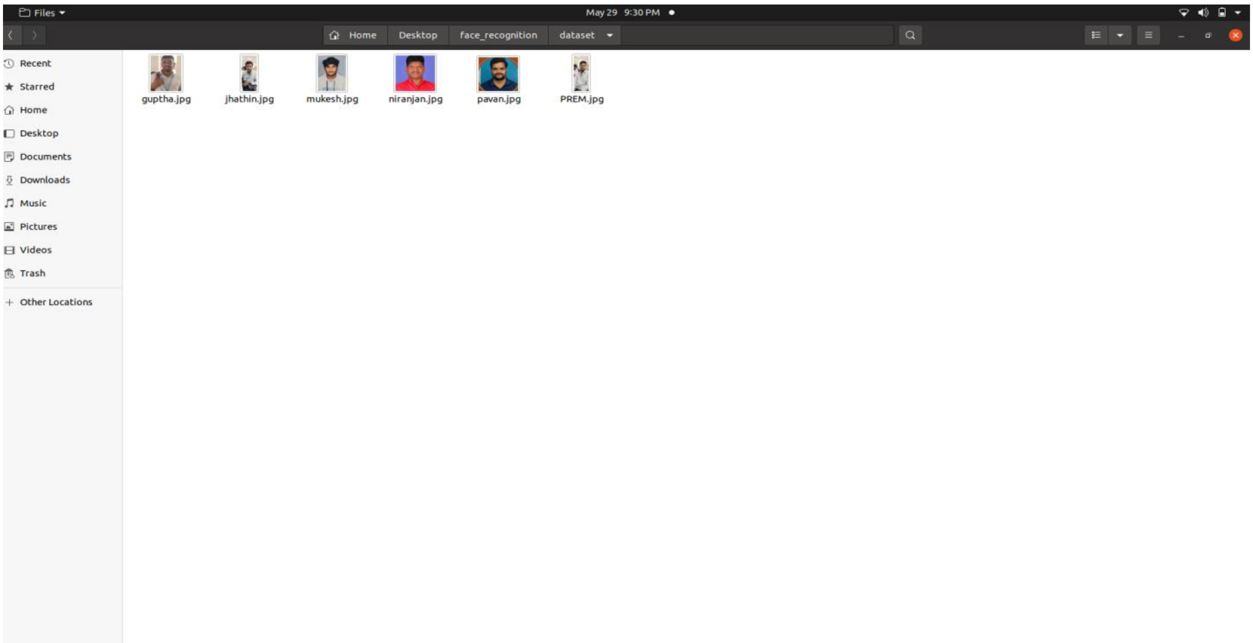
3) After installing the librarys we required we have to create the dataset .



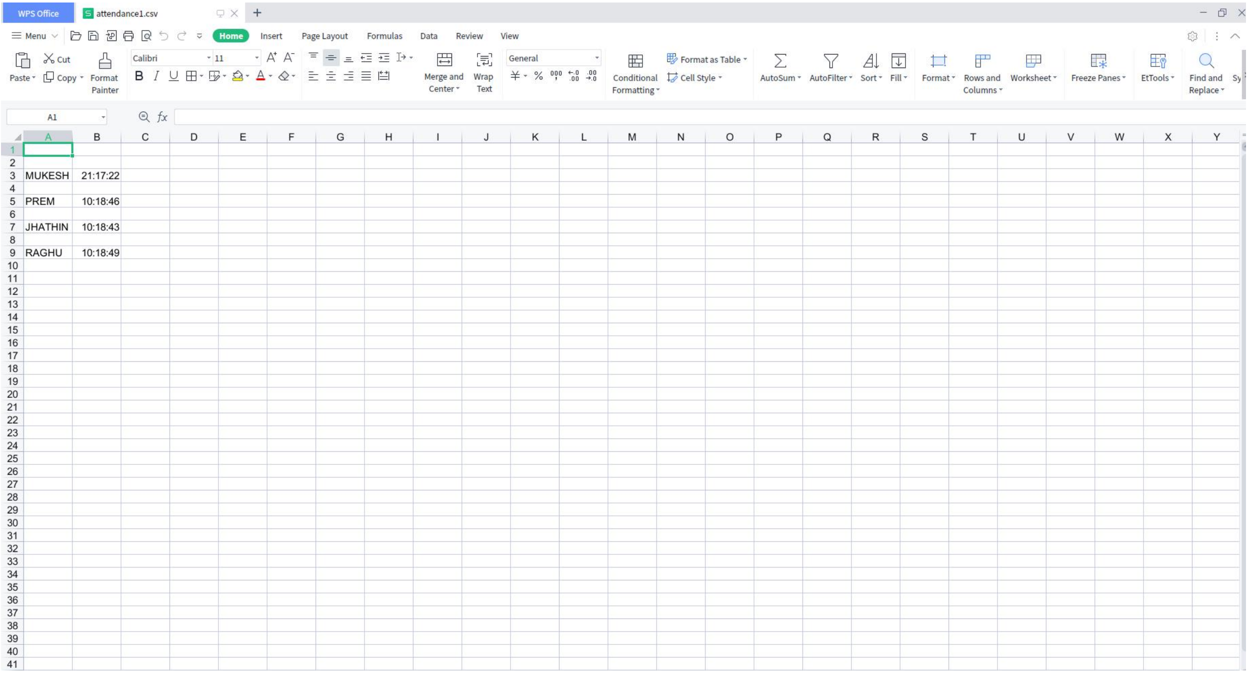
Name	Date modified	Type	Size
dataset	29-03-2021 04:08 PM	File folder	
attendance	26-07-2021 12:06 AM	Microsoft Excel Co...	13 KB
attendance.xl	28-03-2021 05:01 PM	Microsoft Excel W...	9 KB
facerecognisation	02-04-2021 05:57 PM	Python File	3 KB

pythonpp
This PC > Windows (C:) > pythonpp
4 items | 1 item selected 12.5 KB
Type here to search
26°C
12:07 AM
26-07-2021

4) The Dataset is the type of image dataset consist of all the students images and the name in the form of jpeg or jpg format

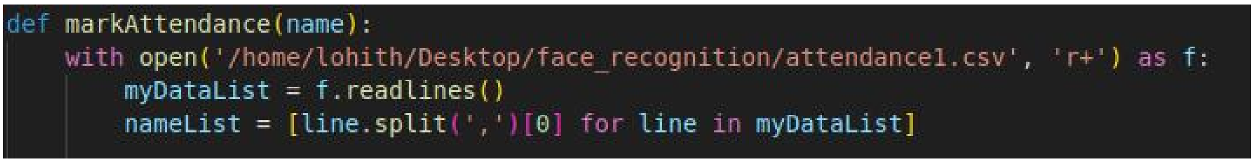


5) we have to create the excel file in ms office with the name of attendance.csv or xlsx format



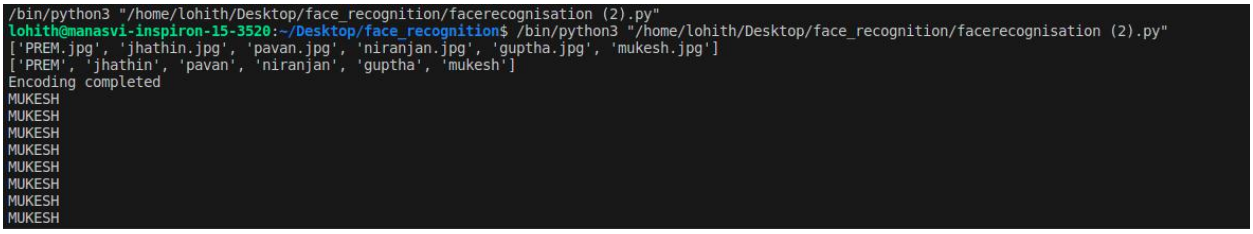
	A	B
1		
2		
3	MUKESH	21:17:22
4		
5	PREM	10:18:46
6		
7	JHATHIN	10:18:43
8		
9	RAGHU	10:18:49

6) we have to provide the path of excel sheet in the code with the root of the file



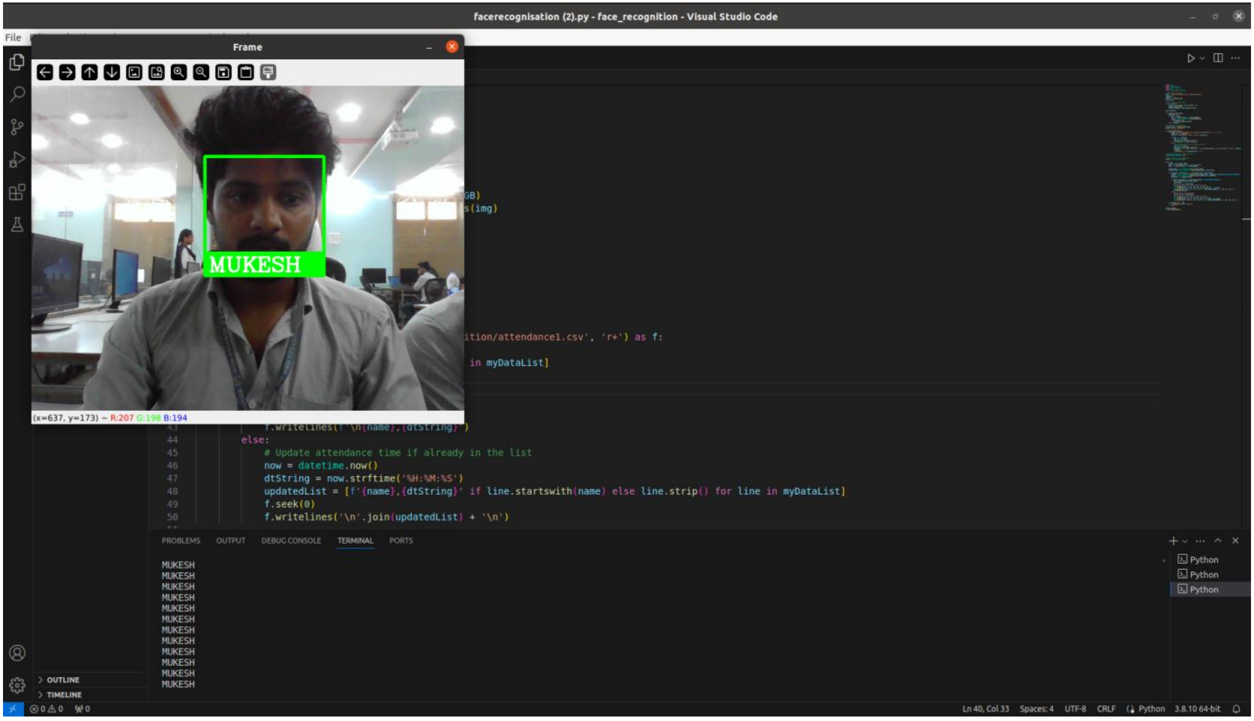
```
def markAttendance(name):
    with open('/home/lohith/Desktop/face_recognition/attendance1.csv', 'r+') as f:
        myDataList = f.readlines()
        nameList = [line.split(',')[0] for line in myDataList]
```

7) Excute the code with the command line



```
/bin/python3 "/home/lohith/Desktop/face_recognition/facerecognisation (2).py"
lohith@manasvi-inspiron-15-3520:~/Desktop/face_recognition$ /bin/python3 "/home/lohith/Desktop/face_recognition/facerecognisation (2).py"
['PREM.jpg', 'jhathin.jpg', 'pavan.jpg', 'niranjan.jpg', 'guptha.jpg', 'mukesh.jpg']
['PREM', 'jhathin', 'pavan', 'niranjan', 'guptha', 'mukesh']
Encoding completed
MUKESH
MUKESH
MUKESH
MUKESH
MUKESH
MUKESH
MUKESH
MUKESH
```

8) Code is execute by the encoding and webcam opens and identys the person in front of camera


facerecognisation (2).py - face_recognition - Visual Studio Code
Frame
MUKESH
(x=637, y=173) ~ R:207 G:198 B:194
PROBLEMS
OUTPUT
DEBUG CONSOLE
TERMINAL
PORTS
Python
Python
Python
Ln 40, Col 33 Spaces: 4 UTF-8 CRLF Python 3.8.10 64-bit

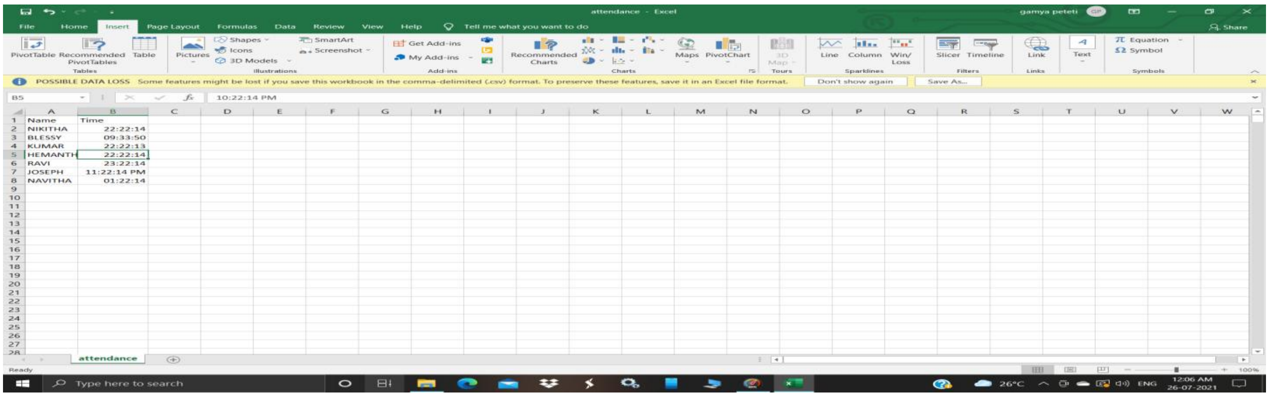
```
GB)
s(img)

ition/attendance1.csv', 'r+') as f:
 in myDataList]

        f.writelines(f'\n{name},{dtString}')
    else:
        # Update attendance time if already in the list
        now = datetime.now()
        dtString = now.strftime('%H:%M:%S')
        updatedList = [f'{name},{dtString}' if line.startswith(name) else line.strip() for line in myDataList]
        f.seek(0)
        f.writelines('\n'.join(updatedList) + '\n')
```

```
MUKESH
MUKESH
MUKESH
MUKESH
MUKESH
MUKESH
MUKESH
MUKESH
MUKESH
MUKESH
MUKESH
MUKESH
```

ATTENDANCE SHEET:

In this attendance sheet we can enter our names which are required to take the attendance. Here by default, we can take it as absent. Later by observing the attendance final they automatically printed as present when both captured image and data base image comparison are equal. When the face of the particular student is recognized, attendance is marked for that student for that particular day.

If some students are absent then no attendance is marked for them. For attendance marking logica1 is written. in the 9×1 cell each cell contains name of the student, and attendance is marked for the student shown in figure15. The attendance sheet is stored in the variable name data. Consider the second name stored in the cell array i.e., him a Shu. The particular case is shown in figure16. Like this other student attendance is marked according to their presence in that particular day. stored in the database


attendance - Excel
File
Home
Insert
Page Layout
Formulas
Data
Review
View
Help
Tell me what you want to do
POSSIBLE DATA LOSS Some features might be lost if you save this workbook in the comma-delimited (.csv) format. To preserve these features, save it in an Excel file format.
Don't show again
Save As...
B5
10:22:14 PM
attendance
Ready
Type here to search
26°C
ENG
12:06 AM
26-07-2021

Name	Time
NIKITHA	22:22:14
BLESSY	09:33:50
KUMAR	22:22:13
HEMANTH	22:22:14
RAVI	23:22:14
JOSEPH	11:22:14 PM
NAVITHA	01:22:14

CHAPTER-8
CONCLUSION

It can be concluded that a reliable, secure, fast and an efficient class attendance management system has been developed replacing a manual and unreliable system. This face detection and recognition system will save time, reduce the amount of work done bythe administration and replace the stationery material currently in use with already existent electronic equipment.

There is no need for specialized hardware for installing the system as it only uses a computer and a camera. The camera plays a crucial role in the working of the system hence the image quality and performance of the camera in real time scenario must be tested especially if the system is operated from a live camera feed.

The system can also be used in permission-based systems and secure access authentication (restricted facilities) for access management, home video surveillance systems for personal security or law enforcement.

The major threat to the system is Spoofing. For future enhancements, anti-spoofing techniques like eye blink detection could be utilized differentiate live from static images in the case where face detection is made from captured images from the classroom. From the overall efficiency of the system i.e., 83.1% human intervention could be called upon to make the system fool proof. A module could thus be included which lists all the unidentified faces and the lecturer is able to manually correct them.

Future work could also include adding several well-structured attendance registers for each class and the capability to generate monthly attendance reports and automatically email them to the appropriate staff for review.

APPENDIX-A

BIBLIOGRAPHY

1. K.Susheel Kumar, S.Prasad, V.BhaskarSemwal, and R.C.Tripathi,“Real Time FaceRecognition Using AdaBoost Improved Fast PCA Algorithm,” Int. J. Artif. ntell. Appl., vol. 2, no. 3, pp. 45–58, Jul.2011.
2. Prof. P.K Biswas, Digital Image Processing.
3. S. Z. Li and A. K. Jain, Eds., Handbook of face recognition. New York:Springer,2005.
4. N. Mahvish, “Face Detection and Recognition,” Few Tutorials, 2014...
5. Anil K Jain, Lin Hong, Sharath Pankanti, and Ruud Bolle, Biometric Identification,IEEE,2004.
6. V. Shehu and A. Dika, “Using Real Time Computer Algorithms in Automatic Attendance Management Systems.” IEEE, pp. 397 – 402, Jun.2010.
7. N. Tom, Face Detection, Near Infinity - Podcasts, 2007.
8. T. Kanade, Computer recognition of human faces. Basel [etc.]: Birkhäuser,1977.
9. A. L. Rekha and H. K. Chethan, “Automated Attendance System using face Recognition through Video Surveillance,” Int. J. Technol. Res. Eng., vol. 1, no. 11,pp. 1327–1330,2014
10. I. Kim, J. H. Shim, and J. Yang, “Face detection,” Face Detect. Proj. EE368 Stanf.Univ., vol. 28,2003.
11. E. Shervin, “OpenCV Computer Vision,” 03-Oct-2010.
12. T. Matthew and A. Pentland, Eigenfaces for Recognition, vol. 3, Volume 3, Number 1 vols. Vision and Modelling Group, The Media Laboratory, MIT: Journal of CognitiveNeuroscience,1991.
13. Y.-Q. Wang, “An Analysis of the Viola-Jones Face Detection Algorithm,” Image Process. Line, vol. 4, pp. 128–148, Jun.2014.
14. Y. Freund, R. Schapire, and N. Abe, “A short introduction to boosting,” J.-Jpn. Soc.Artif.Intell., vol. 14, no. 771–780, p. 1612,1999.
15. P. Viola and M. J. Jones, “Robust real-time face detection,” Int. J. Comput. Vis., vol. 57,no. 2, pp. 137–154,2004.
16. P. Viola and M. J. Jones, “Robust real-time face detection,” Int. J. Comput. Vis., vol. 57,no. 2, pp. 137–154,2004.

17. M.Fuzail,H.M.F.Nouman,M.O.Mushtaq,B.Raza,A.Tayyab,andM.W.Talib,"FaceDetection System for Attendance of Class 'Students."
18. Y. FreundandR.E.Schapire, "Decision-theoretic generalization of on-line learning and an application to boosting," in Computational learning theory, 199

TECHNOLOGY USED

Technologies :

Python,python librarys , webcams(cv), ms excel,numpy, face_recognition,os module, datetime modules .

Code Workflow:

1. Loading Images
2. Encoding Faces
3. Face Recognition and Attendance Marking
4. Displaying Results

Data Management: The code assumes a structured dataset and an existing CSV file for attendance, which must be managed appropriately.

Threshold Setting: The FACE_DISTANCE_THRESHOLD value determines the sensitivity of face recognition. A lower value means stricter matching.