Facial Attendance System using Flask

M.Bindu Sri[1], K.SRIHARI RAO[2], T.Anvitha[3], V.Anusha[4], N.Raj Kamal[5], T.Jayadweep[6]

[1] *Associate professor, Department of Electronics and Communication Engineering, Nri institute of technology, Perechala, Andhra Pradesh*,*India*

[2] *Associate professor, Department of Electronics and Communication Engineering, Nri institute of technology, Perechala, Andhra Pradesh*,*India*

[3]UG Students,*Department of Electronics and Communication Engineering, Nri institute of technology, Perechala, Andhra Pradesh*,*India*

---***---

Abstract - *Face recognition systems are essential in practically every industry in our digital age. One biometric that is frequently utilized is face recognition. It has several more benefits in addition to being useful for security, authentication, and identity. Despite being less accurate than fingerprint and iris recognition, it is nonetheless extensively utilized since it is a non-invasive and contactless technique. Additionally, facial recognition systems may be utilized in businesses, universities, schools, and other settings to indicate attendance. The goal of this system is to create a facial recognition-based class attendance system because the current manual approach requires a lot of time and effort to maintain. There's also a risk that proxies will show up. As a result, this mechanism becomes more necessary. The four stages of this system are database building, face detection, face recognition, and attendance updating. Images of pupils in class are used to develop databases. The local binary pattern histogram technique and the Haar-Cascade classifier are used, respectively, for face detection and recognition. From the classroom's live streamed footage, faces are identified and detected. After the session, an attendance report will be forwarded to the relevant faculty members.*

***Key Words*: Face Recognition, Face Detection, Haar-Cascade classifier, Local Binary Pattern Histogram, attendance system**

1. INTRODUCTION

The morning roll call, when the lecturers would personally call on our names to verify our presence, is one of the most memorable experiences that any college student has. In educational institutions, the routine is drawn out and tiresome. Despite being a crucial aspect of administration, keeping track of attendance can sometimes become a tedious and time-consuming task that is prone to errors. Keeping track of students' attendance during class has grown to be a difficult task. As manual calculating leads to errors and costs a lot of time, figuring out the attendance proportion becomes a crucial task. The computerization of the traditional attendance-taking method forms the foundation for the development of an automatic attendance management system.

1.1 Scope of this Project

With the use of their faces, pupils' attendance will be tracked in this initiative. Typically biometric systems or calling roll numbers are used to record attendance. They are simple to proxy. However, there is a very slim possibility of errors and proxies in our project. The attendance system is entirely automated, and reports are periodically stored.

1.2 A summary of the document, recommended reading, and intended audience

This text is meant to be distributed widely with its contents. But it's crucial to preserve information like the student's name and roll number. The document since it is their duty to make sure that the instructions inside is followed. All employees that handle student data in the course of their employment should also be made aware of these requirements.

2. Introduction to Anaconda

The Anaconda distribution comes with a desktop graphical user interface (GUI) called Anaconda Navigator, which lets you manage conda packages, environments, and channels without the need for command-line interfaces. Navigator has the ability to look for packages in a local Anaconda repository or on Anaconda Cloud.

2.1 Anaconda Distribution

Anaconda is a distribution for Python and R data science, an environment manager, a package manager, and a repository of more than 7,500 open-source programs. Anaconda provides free community assistance and is simple to install.After obtaining the Anaconda Cheat Sheet, launch Anaconda.Do you want to install conda and utilize it to install only the necessary packages? Acquire Miniconda. utilize Navigator if you would rather utilize a desktop graphical user interface (GUI) after installing Anaconda or Miniconda. Use conda and the Anaconda prompt (also known as the terminal on Linux or macOS) if that's how you like to work. Additionally, you may flip between them. After installing Anaconda, launch the command anaconda-navigator in the Anaconda prompt (or terminal on Linux or macOS) to try Navigator. Alternatively, select the Navigator icon on the application menu of your operating system.Take the 30-minute conda test drive and obtain a conda cheat sheet to give conda a try after installing Anaconda or Miniconda. Write your first Python project using Anaconda by following the instructions at Getting started with Anaconda if you're new to the program. To read the user guide from beginning to end, utilize the navigation tabs at the bottom of the page.To understand how to utilize Anaconda's graphical user interface, see the Navigator user manual. Take a look at the UI and learn how to use Navigator at Opening Navigator for the first time.Anaconda and Microconda both come with Conda, an open source package management and environment management system. Find out how to install and use conda.

2.2 Getting Anaconda Started

Python, hundreds of scientific programs, Anaconda Navigator, and conda are all included in the Anaconda Distribution. All of these were also installed when you installed Anaconda.Conda may be used using a command line interface, such as the Linux and macOS terminal or the Windows Anaconda Prompt. With Navigator, you can quickly manage conda packages, environments, and channels without needing to use command-line commands. Navigator is a desktop graphical user interface. To determine which is best for you to manage your packages and environments, you may test both Navigator and conda. The work you complete in one may be viewed in the other, and you can even flip between them.

2.3 The Anaconda Navigator

The Anaconda distribution comes with a desktop graphical user interface (GUI) called Anaconda Navigator, which lets you manage conda packages, environments, and channels without the need for command-line interfaces. It also lets you start programs. Navigator has the ability to look for packages in a local Anaconda repository or on Anaconda Cloud. It may be downloaded for Linux, macOS, and Windows.

3. LITERATURE SURVEY

Face-specific systems and object recognition systems both employed a variety of techniques to address disparities in pictures that were affected by variations in light. Gray-level data was utilized to extract a face or an object from the shading strategy as a way to cope with these differences.Because grayscale representations simplify the procedure and minimize computation overhead, they are preferred over directly working with color pictures for extracting descriptors.Color isn't really helpful in this situation, and adding extraneous information might make more training data needed to get decent results. These suggested answers assumed the object's form and reflectance characteristics or the lighting circumstances since the challenge was ill-posed. It didn't work well for facial recognition since the presumptions were too rigid for generic object recognition. The second method makes advantage of the image's edge map, a helpful characteristic for representing objects that is unaffected by changes in lighting or specific events.Images with edges might be used to recognize objects and attain accuracy levels comparable to those of grayscale images. The edge map information technique has low memory requirements and is feature-based approach's invariance to illumination. It groups the pixels of the face edge map into line segments, so integrating the structural information with the spatial information of a face picture. The edge map of a face is created by applying a polygonal line fitting procedure after the edge map has been thinned. Another method for handling picture discrepancies caused by variations in illumination is to use a model made up of many photos of the same face taken in different lighting scenarios. Here, the photos that were taken can be utilized as separate models or as part of a larger model-based recognition system.

4. OVERALL DESCRIPTION

4.1 Feasibility Report

4.1.1 Technical Feasibility

The most difficult aspect of a feasibility study is evaluating technical feasibility. This is because there aren't many in-depth system designs at this moment, which makes it challenging to access concerns like performance, pricing, etc. (due to the type of technology that has to be developed). Several factors need to be taken into account while doing a technical analysis.

4.1.2 Operational Feasibility

Only when proposed ideas can be developed into an information system that satisfies the operational needs of the company will they be beneficial.In other words, this feasibility test questions if the system will function when it is constructed and put into place.Exist significant obstacles to implementation?

4.1.3 Economic Feasibility

The goal of economic feasibility is to balance the advantages of having a new system in place with the expenses of creating and executing it. The feasibility study will provide the new system's senior management with an economic basis.

In this instance, a straightforward economic analysis that provides an accurate cost-benefit comparison is far more significant. Additionally, this serves as a helpful point of reference to compare actual costs as the project moves forward. There may be a variety of intangible benefits due to automation, such as improved customer satisfaction, higher-quality products, better decision-making, faster information retrieval, expedited activities, improved operational accuracy, better documentation and record-keeping, and higher employee morale.

5. Existing System

Everything is done on paper under the current system. The attendance record for each session is kept in the register, and reports are created at the conclusion of the semester, which requires additional processing time. Roll numbers will be called in order to manually record attendance in registers at first. A later feature is biometric attendance. The time required for attendance in the old fashion has decreased thanks to biometrics, but there are still some drawbacks, such as proxy and network connection difficulties. The current system has the following drawbacks:

1. To build the report, we need to perform further computations.
2. Manual labor is required, and the process takes time.
3. Report generation requires a lot of documentation.
4. Even though attendance time may be readily altered, it has been lowered through the use of Bio-Metric.
5. Prospective attendance by proxy.
6. We suggested a solution in order to get around the problems with manual and biometric.

6. Proposed System

Student photos are gathered in this attendance system so that students' faces may be identified. The student database, which consists of all of the students' faces, is used to train the algorithm at first. In order to optimize the user experience during training and testing—which involve using the system to take attendance and gather student photos—the system has an intuitive user interface. Numerous more applications where facial recognition is used for authentication might benefit from this study. At the conclusion of the semester, reports will be generated automatically by the system.The benefits of the suggested system are listed below.Our suggested method will collect attendance by identifying a person's face.

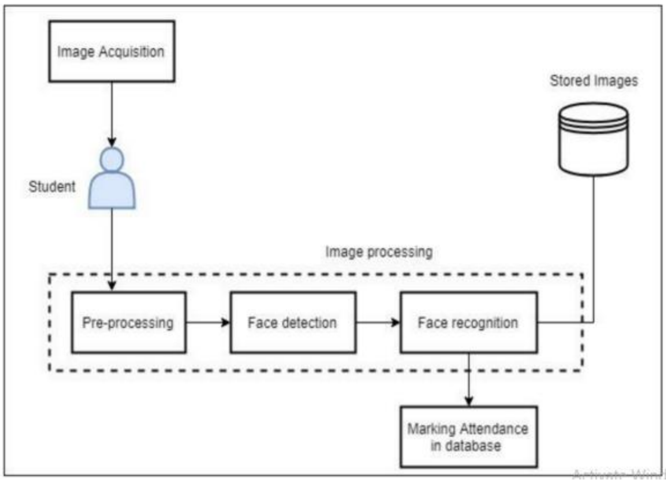


Fig -1: Architecture of the System

6.1 Use case Diagram

Utilization Case Using actors and use cases, the diagram encapsulates the functionality and needs of the system. Use cases simulate the functions, duties, and services that a system must provide. Use cases show high-level features and the way a user will interact with the system.

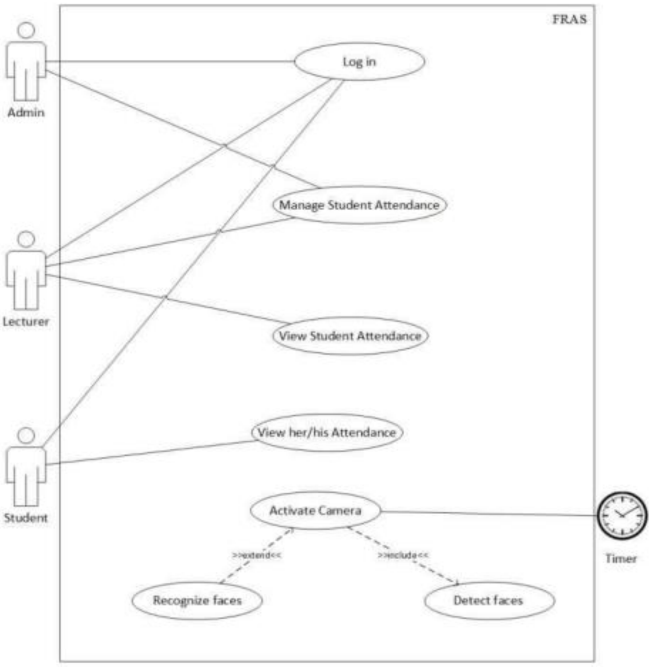


Fig -2: use case illustration

6.2 Sequence Diagram

UML Sequence Diagrams are interaction diagrams that show the steps involved in performing an operation. They depict how items interact with one another when working together.

Sequence diagrams, which have a time emphasis, use the vertical axis to represent time, the messages transmitted, and the timing of those messages to graphically depict the sequence of the interaction.

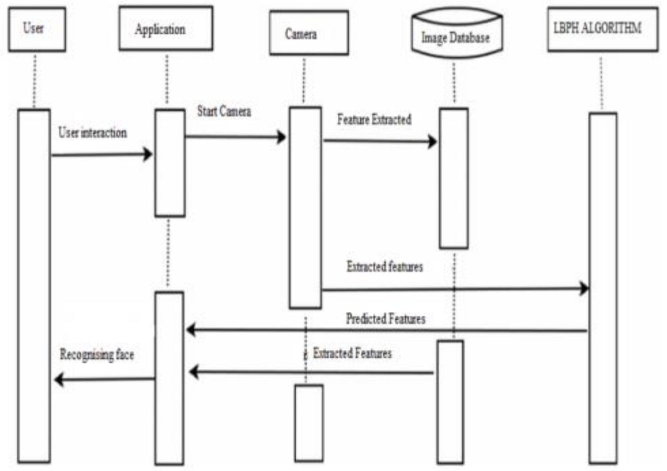


Fig -3: schematic of the series

6.3 Activity Diagram

An activity diagram is a type of UML diagram where the execution and flow of a system's behavior are the main emphasis rather than implementation. Another name for it is an object-oriented flowchart. Activities that are composed of actions that are applicable to behavioral modeling technologies make up activity diagrams.

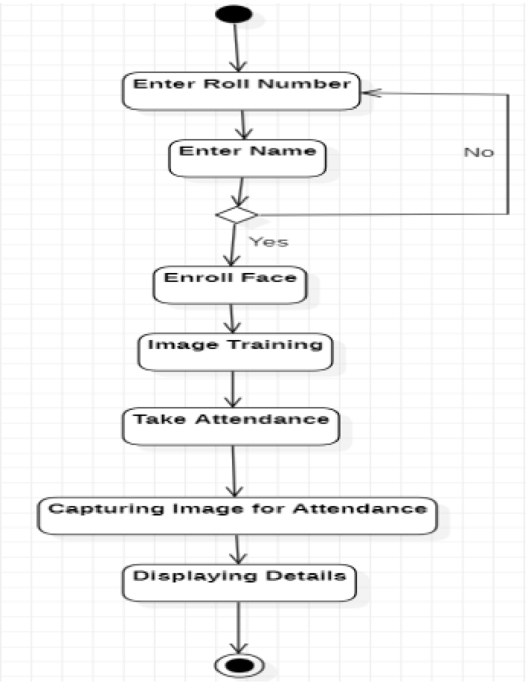


Fig -4: Activity Diagram

7. IMPLEMENTATION

Python is a robust and simple-to-learn programming language. Its object-oriented programming methodology is straightforward but efficient, and its high-level data structures are efficient. Python's interpreted nature, dynamic typing, and beautiful syntax make it a perfect language for scripting and quick application development across a wide range of platforms. From the Python website, https://www.python.org/, the large standard library and the Python interpreter are freely distributable in source or binary form for all major systems. Distributions of and links to several free third-party Python modules, tools, and applications, as well as more documentation, may be found on the same website. It is simple to add new functions and data types to the Python interpreter that are implemented in C or C++ (or other languages that may be called from C). Python may also be used as an extension language to create programs that can be customized. See The Python Standard Library for a description of standard modules and objects.

A more formal definition of the language may be found in the Python Language Reference. Read Extending and Embedding the Python Interpreter and the Python/C API Reference Manual before writing any C or C++ modifications. Python is also thoroughly covered in a number of books.

7.1 Introduction to OpenCV

Gary Bradsky founded OpenCV at Intel in 1999, and the first version was released in 2000. Vadim Pisarevsky took over Gary Bradsky's position as manager of Intel's OpenCV software team for Russia. Stanley, the vehicle that won the 2005 DARPA Grand Challenge, was equipped with OpenCV. Later, with Gary Bradsky and Vadim Pisarevsky in charge of the project, Willow Garage supported its ongoing active development. Currently, OpenCV supports a large number of machine learning and computer vision methods, and its list is growing daily.

7.2 Face Recognition

In order to retake attendance in this module, we must recognize a student's face in the picture and display a rectangle box around it. We'll utilize the haarcascade file for that once more. The Traineer.yml file contains the characteristics that the LBPH Algorithm extracts from the photos. It will extract the features from the current image at attendance time and compare them with preexisting features; if it discovers a match, attendance will be recorded for that specific student.

8. RESULTS

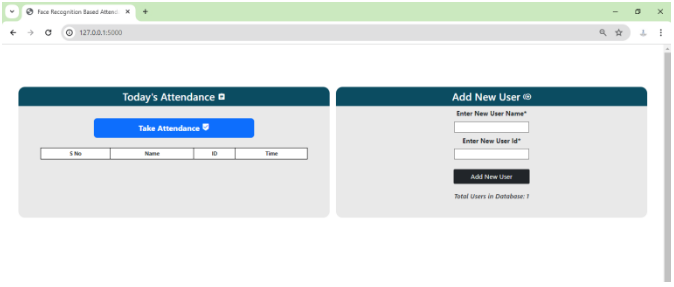
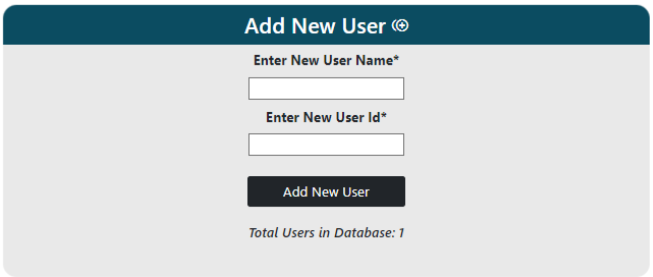
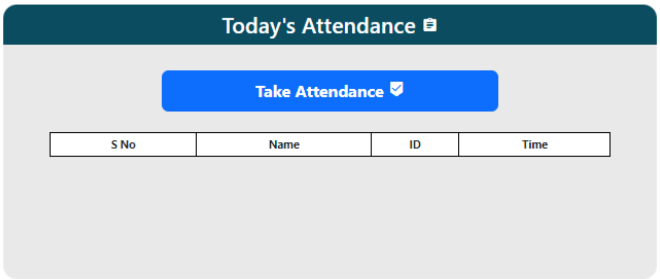


Fig -5: Home page



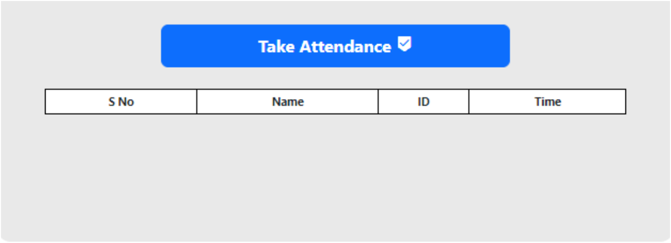
Total Users in Database: 1

Fig -6: User Registration



S No	Name	ID	Time

Fig -7: Take Attendance



S No	Name	ID	Time

Fig -8: Attendance Data

3. CONCLUSIONS

This automated attendance system has been put into place with the aim of lowering the mistakes that arise in the conventional method of taking attendance. This research presents a deep learning-based facial recognition system that is resilient and accurate, recognizing users with 98.3% accuracy. The outcome demonstrates the system's capacity to adapt to changes in face projection and posing. According to research on deep learning face recognition, the lighting issue is resolved during face identification as the original image is converted into a HOG representation that effectively captures the main characteristics of the image independent of its brightness. Local facial landmarks are taken into consideration for subsequent processing in the face recognition algorithm. Faces are then encoded, producing 128 measures of the photographed face. The best way to recognize a face is to decode the encoding to get the subject's name. An Excel sheet is then created using the outcome.

REFERENCES

B.K.P. Horn and M. Brooks, Seeing Shape from Shading. Cambridge, Mass.: MITPress, 1989.

A.F. Abate, M. Nappi, D. Riccio, and G. Sabatino, "2D and 3D face recognition: A survey", Pattern Recognition Letters, vol.28, issue 15, pp.1885-1906, Oct 2007.

Yael Adini, Yael Moses, and Shimon Ullman, “Face Recognition: The Problem of Compensating for Changes in Illumination Direction”.

Kanan C, Cottrell GW (2012) Color-to-Grayscale: Does the Method Matter in Image Recognition? https://doi.org/10.1371/journal.pone.0029740

T. Kanade, Computer Recognition of Human Faces. Basel and Stuttgart: Birkhauser Verlag 1997.

ABOUT THE AUTHOR

K.SRIHARI RAO.
DESIGNATION: PROFESSOR.
QUALIFICATION: pH.D
EXPERIENCE: 34 yrs.