Journal of **N**onlinear **A**nalysis and **O**ptimization
Vol. 15, Issue. 1 : 2024
ISSN : **1906-9685**

FACE RECOGNITION-BASED ATTENDANCE SYSTEM WITH MACHINE LEARNING AND LOCATION DETECTION

[#1]A.SHARADA , ***Assistant Professor, Department of CSE (AI&ML),***

[#2]K.VIDYA, ***Assistant Professor, Department of CSE,***

VAAGESWARI COLLEGE OF ENGINEERING, KARIMNAGAR, TELANGANA.

Abstract: According to this concept, teachers waste too much time and energy keeping attendance records the old-fashioned manner. Taking attendance by going around the room and shouting out names or taking them down on paper is a slow and straightforward method that can be hacked. The findings imply that these issues could be remedied by implementing a computer-based attendance tracking system driven by computer vision. It may be easier to learn photographs in school with the help of computer vision. Cameras, sensors, and algorithms can be used to process and study visual data such as images of people. This is known as "computer vision." This would enable the use of facial recognition technology in attendance management, allowing attendance to be automatically tracked and recorded. Using this method to track attendance instead of the traditional method, which was done by hand, provides numerous advantages. Automatic attendance management systems may appear to be a good idea at first glance since they eliminate the need for individuals to keep track of attendance, which takes a lot of time and effort. This device provides real-time attendance statistics, allowing you to keep track of pupils who arrive late or leave early. The approach accurately collects daily attendance data, which administrators can utilize to identify and correct attendance and participation issues. The primary purpose of this study was to demonstrate the utility of an automated technique for tracking attendance using computer vision. This technology could help students receive better scores if it is utilized to make attendance data easier to interpret.

Keywords*: Python; OpenCV and Google API; Student attendance; Face recognition*

1. INTRODUCTION

There are a few options for dealing with physical presence, but the majority of them either prohibit remote data storage or require you to join a private cloud service. A low-cost attendance system is created by integrating free cloud storage services (such as Google Sheets) with Python tools (such as OpenCV) that detect faces and track attendance. We can make it much easier to find people and keep track of attendance by using these simple but powerful Python features. Using the Google Application Programming Interface (API), we were able to connect Google Sheets to our time and attendance system. This made cloud attendance tracking simple and inexpensive. If the proposed strategy is implemented, businesses of all sizes may be able to be more effective and save money on costs associated with maintaining attendance. Using Python-based tools and inexpensive cloud storage, it is possible to make attendance tracking easier and less expensive.

2. LITERATURE SURVEY

Because of all the technologies in today's

classrooms, kids' attendance is becoming increasingly vital. Names are no longer spoken aloud and signatures are no longer gathered as part of the registration process. Naveen Raj composed the letter. Making a good Monday paper is not easy. This is why the "Python-based Student Attendance System with Facial Recognition" was created. The computer's camera captures a picture of the student, and an algorithmic analysis of the image assists in determining who the student is. When face recognition works properly, data is transmitted to an Excel table and attendance is captured immediately. The GUI was created using Python Tkinter, which allows users to easily navigate around and interact with the system. The new method of keeping track of numbers is more accurate and less prone to error. Because of this significant divergence from the usual, the value of normal schooling could skyrocket. Shivangi Awasthi and 2Shubhangi Awasthi published "Facial Recognition Attendance System Using Python" in the 2022 edition of the International Journal of Research Publications and Reviews.

Face recognition (AFR) technology has brought about numerous improvements in our ever-changing culture. This strategy can be utilized in the real world for things like attendance tracking. Students could be identified via a facial recognition-based attendance system that uses biometric traits and high-definition camera video. My goal is to use deep learning to build an automatic facial recognition system capable of finding and identifying people in still photographs and video surveillance data. Face recognition has improved because to various algorithms and methodologies. I want to apply deep learning to convert video frames into visually appealing images so that attendance records may be updated instantly. This technology has the potential to make tracking which students are present in classrooms considerably easier, faster, and more precise. (Divya Pandey will demonstrate her Python-based Face Recognition Attendance System at JETIR in October 2020.

The camera image is used to compare the student's face to a set of other faces. If a student's face can be identified, their appearance is recorded in the attendance record; otherwise, their absence is recorded. Face recognition software can generate a biometric identifier based on each person's distinct and easy-to-remember facial traits.

Monitoring attendance strategies are critical for schools, colleges, and universities to keep the peace and ensure that all students get the most out of their education. Calling "roll" or collecting signatures on paper are two inefficient and time-consuming methods of taking attendance. Today's tools, on the other hand, enable you to create a system that does not require outside assistance and is quick and simple to use. This project includes an automatic attendance system constructed with a Raspberry Pi 3B+, OpenCV/Python tools, and a recognizer algorithm. Teachers can take roll without being distracted in this manner. Using face recognition technology saves time and eliminates the possibility of proxy votes. This strategy could be useful in any location that wants to retain precise attendance data. It is critical to the project's goals and aesthetics. This engineering solution's success is largely due to its ability to assist schools in keeping track of and adhering to attendance policies. Furthermore, the suggested procedure ensures precision, eliminates all potential of error, and reduces the time required. Ghalib Al-Muhaidhri and Javeed Hussain are among the writers mentioned. Have a discussion at IJERT 2019: "Smart Attendance System Using Face Recognition" The Internet of Things (IoT) is transforming the way colleges keep track of who is attending by using fast and cheap new technologies. Using the Internet of Things, a prototype of a Smart Attendance System that can track users' attendance, send out records, and keep a watch on them was created. The mobile device allows for rapid attendance by passing it from one student to the next. Many testing have been conducted on the technology, and preliminary findings indicate that it will save time for teachers and workers. Using this technology allows us to explore and advance in this area in novel ways. It also demonstrates how using IoT technologies

may improve and streamline reporting methods in any setting. It was written by V. Yadav and G. P. Bhole.

3. METHODOLOGY

Cmake

Before creating native build files, a native build system can be configured to create libraries, compile source code, build containers, compile executable binaries, and store data that can be modified by the user. This provides consumers more control over the building process and allows them to fine-tune and adjust their built settings more easily.

Dlib

Using the well-known machine learning framework Dlib, we can determine 68 face coordinates (x, y) that correspond to different facial landmarks. Many people use this toolkit because it offers simple Python bindings that allow you to create high-quality machine learning and data analysis tools. When I need to recognize someone by their face, this is the first thing I look for. The individuals and domains referenced were clearly real to me.

Face Recognition

The Dlib program for detecting faces is simple to use and effective. Face recognition systems at the leading edge of technology enable this. Because it works 99.38% of the time, Labeled Faces in the Wild is one of the most effective face recognition libraries. Dlib is not only very accurate, but it also offers a command-line utility that can be used to find specific people in a large number of images. Anyone, regardless of technology knowledge, can utilize this resource to include facial recognition into their projects. Dlib is widely used in facial recognition software because it performs well and is easy to use.

OpenCV

OpenCV is an open-source machine learning and computer vision tool that includes everything you need. This site provides users with access to over 2500 highly helpful algorithms that may be used for a variety of tasks, such as discovering related photographs inside a larger one. Since its release, more than 18 million individuals have downloaded OpenCV. As a result, many people who are interested in AI employ it in their profession. Because it is so adaptable and has so many capabilities, many people use it as the standard for computer vision and machine learning jobs in a variety of fields.

Google API

The Google APIs are a set of computer languages developed by Google to allow data movement between Google services and those of other firms easier. This is evident in Google's search engine, email, translation tools, and maps. These application programming interfaces (APIs) provide access to technologies for machine learning, analytics, and user data to developers working on Google Maps. One of the nicest features of Google APIs is the RESTful interface for reading and updating data to Google Sheets. Programming languages such as JavaScript, PHP, and Python can be used in real time to access and modify data in Google Sheets.

Google API - Authentication and authorization

You must be permitted and authenticated even while accessing the Google API. Many people utilize OAuth 2.0 for this because it is dependable and simple to use. Before they can utilize a Google API, developers must first sign up for keys in the Google Developers Console. Once the client app has gathered all of the necessary information, it can request an access pass from the Google Authorization Server. After verifying the client application's access code, the client is permitted to utilize the Google API service. Because OAuth 2.0 is simple to implement, writers may find it straightforward to incorporate it into their applications.

Google Sheets API

- The Google Sheets API is RESTful in design and allows users to view and modify spreadsheets. You can edit existing information, count the number of cells, and create new papers.
- To perform computations, numbers are added

to and subtracted from cells in a worksheet.

- Feel free to alter the appearance of the paper. Spreadsheets with a lot of linkages are simple to utilize.

Work flow

This is the suggested approach for a facial recognition project that entails acquiring information about coworkers, as shown in Figure 1. Certain applications must be installed in order to complete the task as specified. OpenCV activates the camera and records live images of faces for further processing. The Face Recognition Python software will search for identifying features in a given image. Python stores all of this information as a list. The embedded face is then compared to a list of known pupil sizes by the Python application. When a person is discovered, a CSV file is saved locally. It will keep attendance data in the cloud for free. A Key file can be used to link the Python software to Google servers using the Google API. Connect Google Sheets to the Google Sheets API after that. A Google Sheet contains a list of everyone's names and expected arrival times. Every day, there are file deletions. The daily attendance log is kept on-site in case it is needed in the future. The date that a face was discovered is appended to the end of the new CSV file's name. This strategy works well and is convenient because storing data locally or online is free.

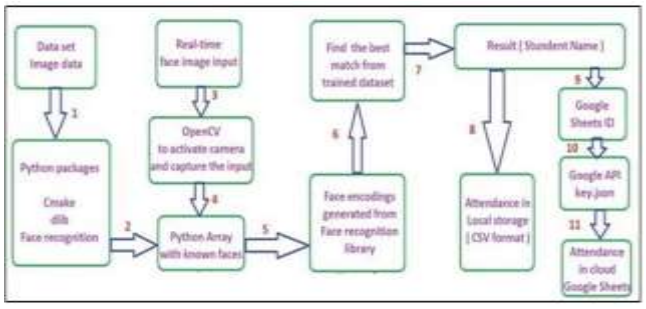

Figure 1 Python, OpenCV, and the Google API can be used to create a face tracking system.

Implementation

Figure 2 depicts the procedures required to locate a photograph and preserve it in a database so that it can be seen.

Figure 2 A current picture of each student should be kept in their personal folder.

Figure 3 shows how OpenCV activates the camera so that it may receive a live image of a face as input.

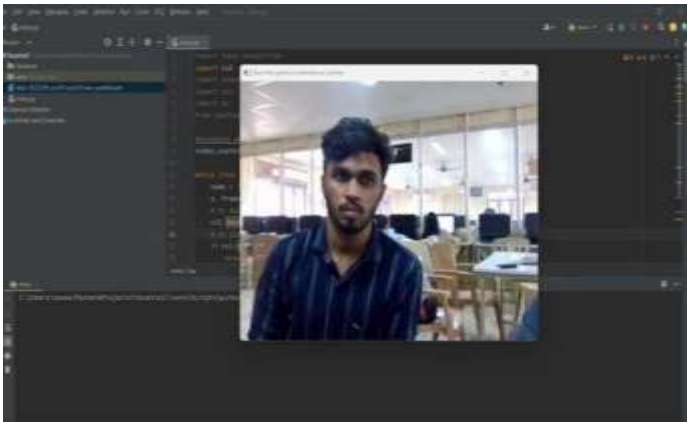

Figure 3 Tap the record button.

We can select the pupil that best matches the face using the Face detection module's face encodings and a Python array with a list of known pupils (Figure 4).

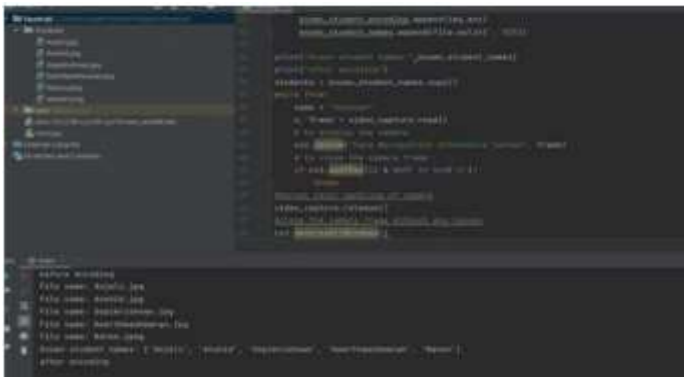

Figure 4 Face Recognition Techniques

As soon as the rear imaging procedure is completed (see Fig. 5), real-time face monitoring and image verification can begin.

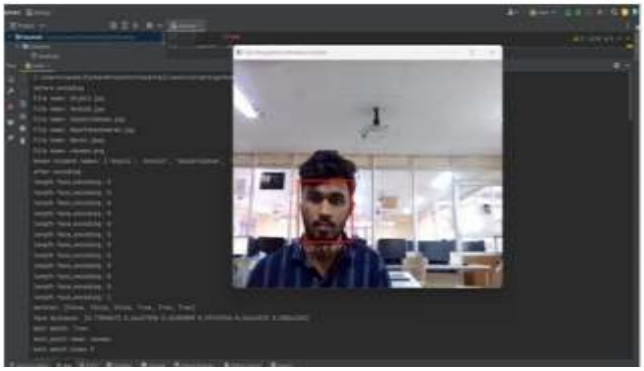

Figure 5 Having an up-to-date CSV file on your computer containing the person's current information.

4. CONCLUSION

The Google API is what distinguishes the application because it will bring back the old-fashioned paper and pencil method of keeping numbers up to date. As a result, the system operates more precisely and swiftly. This was created using pre-existing Python software. These apps are incredibly popular because they operate effectively and are simple to use.

One of the numerous advantages of Python-based systems is that software may be created more quickly. Workers can concentrate more on the key aspects of their jobs because they don't have to worry about the specifics of how it will be done. If we don't squander as much time, we can get more done in less time.

It is critical to ensure that the project is cost effective. Google Sheets can be used to keep track of attendees instead of spending a lot of money on cloud storage. Google Sheets is a cloud-based chart program that allows users to collaborate and share data in real time. Because of this, small enterprises and organizations can use the system without having to hunt for cheaper cloud storage choices.

Face recognition software that works well and is inexpensive was created using the Google API and other Python-based technologies. This solution maintains accurate attendance data in real time and is simple to integrate into existing processes..

REFERENCES

[1] Shivangi Awasthi, 2Shubhangi Awasthi Facial Recognition Attendance System Using Python at 2022 International Journal of Research Publication and Reviews

[2] Divya Pandey, 2Priyanka Pitale, 3Kusum Sharma Face Recognition Based Attendance System using Python at JETIR October 2020, Volume 7, Issue 10

[3] Sudhir Bussa, Shruti Bharuka, Ananya Mani, Sakshi Kaushik Smart Attendance System using OPENCV based on Facial Recognition at 2020 International Journal of Engineering Research & Technology (IJERT)

[4] Ghalib Al-Muhaidhri [1], Javeed Hussain[2] Smart Attendance System using Face Recognition at 2019 International Journal of Engineering Research & Technology (IJERT)

[5] Yadav, V., & Bhole, G. P. (2019). Cloud based smart attendance system for educational institutions. In 2019 international conference on machine learning, big data, cloud and parallel computing (Com-IT-Con)

[6] Varadharajan, E., Dharani, R., Jeevitha, S., Kavinmathi, B., & Hemalatha, S. (2016). Automatic attendance management system using face detection. In Online international conference on green engineering and technologies (IC-GET).

[7] Arsenovic, M., Sladojevic, S., Anderla, A., & Stefanovic, D. (2017). FaceTime—Deep learning based face recognition attendance system. In IEEE 15th international symposium on intelligent systems and informatics. University of Novi Sad, SISY 2017, September 14–16, Subotica, Serbia.

[8] Shah, K.; Bhandare, D.; Bhirud, S. Face recognition-based automated attendance system. In International Conference on Innovative Computing and Communications; Springer: Berlin/Heidelberg, Germany, 2021; pp. 945–952