Software as a Service Report

(Smart Attendance System with Face Recognition using OpenCV)

Shumbul Arifa, **Keerti Chaudhary**

Students of Department of Computer Science and Engineering, National Institute of Technology Karnataka, Surathkal, Karnataka, India

shumbul.181co152@nitk.edu.in , keerti2001.kc@gmail.com

Abstract

We seek to provide a valuable attendance service for both teachers and students and reduce the manual process errors by providing an automated and reliable attendance system that uses face recognition technology. The proposed solution is to develop an operating prototype of a system that may facilitate class attendance management for the lecturers within the lecture rooms by detecting the faces of scholars from an image taken in a classroom. The database can store the faces of scholars, once the face of the individual matches with one in all the faces held within the database then the attendance is recorded. In recent years, analysis has been dispensed and face recognition and detection systems are developed.

Keywords: SaaS, SOA, Face Recognition, Face Detection, OpenCV, User Interface, Attendance system, Architectural Design

1 Introduction:

Maintenance of attendance is incredibly necessary altogether at the institutes for checking the performance of students. Every institution has its method of taking student attendance Some institutes take attendance manually using paper or file-based approaches and a few have adopted ways of automatic attendance using some biometric techniques. The recent methodology for taking attendance is by calling out the name or roll number of the scholar to record attendance. It's a long and less efficient method of marking attending as a result of as we all know the info written within the paper typically may be lost or is less accurate as a result of students often mark every other's attending proxy. Therefore, to unravel these issues and avoid errors, we recommend computerizing this method by providing a system that records and manages student's attending mechanically with no need for lecturers' interference.

Every biometric system consists of the enrollment process during which distinctive features of someone are stored within the database and then there are processes of identification and verification. These 2 processes compare the biometric feature of someone with previously stored biometric details captured at the time of enrollment. Biometric templates are of the many varieties like Fingerprints, Eye Iris, Face, Signature, and Voice. Our system uses the face recognition approach for the automated Attendance Management System. This product gives many more solutions with accurate results in a user interactive manner rather than existing attendance management systems. Face recognition consists of two steps, in the first step faces are detected in the image and then these detected faces are compared with the database for verification. Face detection is employed to find the position of face region and face recognition is employed for marking the attendance.

2 Software as a Service architecture

In the SaaS model, cloud suppliers deliver application software packages as a service to cloud users. A group action management system may be a necessary tool for taking group action in any atmosphere wherever group action is essential. However, most of the prevailing approaches square measure long, intrusive, and need manual work from the users.

The following are five of the top advantages of using SaaS: (Source: IBM)
1. Reduced time to benefit
Software as a service (SaaS) differs from the traditional model because the software (application) is already installed and configured. You can simply provide the server for an instance in the cloud, and in a couple of hours, you'll have the application ready for use. This reduces the time spent on installation and configuration and can reduce the issues that get in the way of the software deployment.

2. Lower costs
SaaS can provide beneficial cost savings since it usually resides in a shared or multi-tenant environment, where the hardware and software license costs are low compared with the traditional model.
Another advantage is that you can rapidly scale your customer base since SaaS allows small and medium businesses to use software that otherwise they would not use due to the high cost of licensing. Maintenance costs are reduced as well since the SaaS provider owns the environment and it is split among all customers that use that solution.

3. Scalability and integration
Usually, SaaS solutions reside in cloud environments that are scalable and have integrations with other SaaS offerings. Compared with the traditional model, you don't have to buy another server or software. You only need to enable a new SaaS offering and, in terms of server capacity planning, the SaaS provider will own that. Additionally, you'll have the flexibility to be able to scale your SaaS use up and down based on specific needs.

4. New releases
With SaaS, the provider upgrades the solution and it becomes available for their customers. The costs and effort associated with upgrades and new releases are lower than the traditional model that usually forces you to buy an upgrade package and install it (or pay for specialized services to get the environment upgraded).

5. Easy to use and perform proof-of-concepts
SaaS offerings are easy to use since they already come with baked-in best practices and samples. Users can do proofs-of-concept and test the software functionality or a new release feature in advance. Also, you can have more than one instance with different versions and do a smooth migration. Even for large environments, you can use SaaS offerings to test the software before buying.

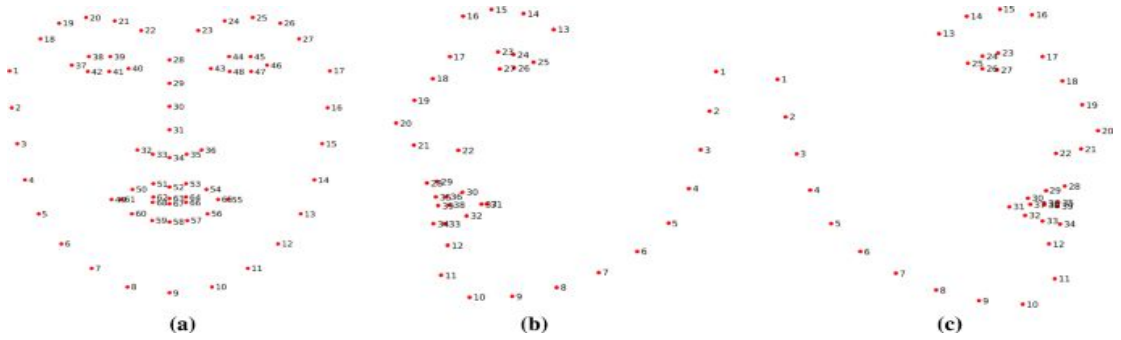
2. 1 System Design

To square measure seeking to settle on ought to host the net application, the windows service and will be user friendly. The specified internet application parts are:

- Face Enrolment: The enrollment method involves making face templates from a group of pictures and a digital camera. The face guide keeps within the information for face recognition.
- Image Storage: All pictures non-inheritable from the science camera video live feed similarly as student listed face templates square measure regenerate to binary and keep within the information.
- Image Enhancement to grayscale: Most times, non-inheritable pictures have darkness most likely as a result of poor lighting within the atmosphere. Before these pictures are used for enrollment, detection, or

recognition functions, it's to be normalized. standardization enhances the accuracy of face detection and recognition. This method begins by changing the RGB image into grayscale. bar graph standardization is then applied for distinction sweetening.

- Face Detection: this is often the power to recognize an individual's face in a picture. Facial Landmark Detector. During this step, the system detects 68-point facial landmarks on the face image then extracts this information for classification within the next step.
- Face Recognition: The server ought to be able to confirm if a listed face guide keeps within the information is found within the non-inheritable image.



Posing and Projecting Faces

The features are then compared with existing records to check if there is a match. If the face matches it is displayed and output is in the form of attendance being marked for the person whose face was recognized. In our model, we will be providing the users the options to choose a given function from the above possible functions, the pseudocode for the same is given below:

1. Check Camera
2. Capture Faces
3. Train Images
4. Recognize & Attendance
5. Auto Mail
6. Quit

```
Begin
   • if choice = 1
        call the function to check if the system/external camera is working
   • elif choice = 2
        call the function which captures faces
   • elif choice = 3
        call the function which trains images
   • elif choice = 4
         the function that recognizes the person from the set of saved images
     from the database is called
   • elif choice = 5
          send Mail to the admin
   • else exit
End
```

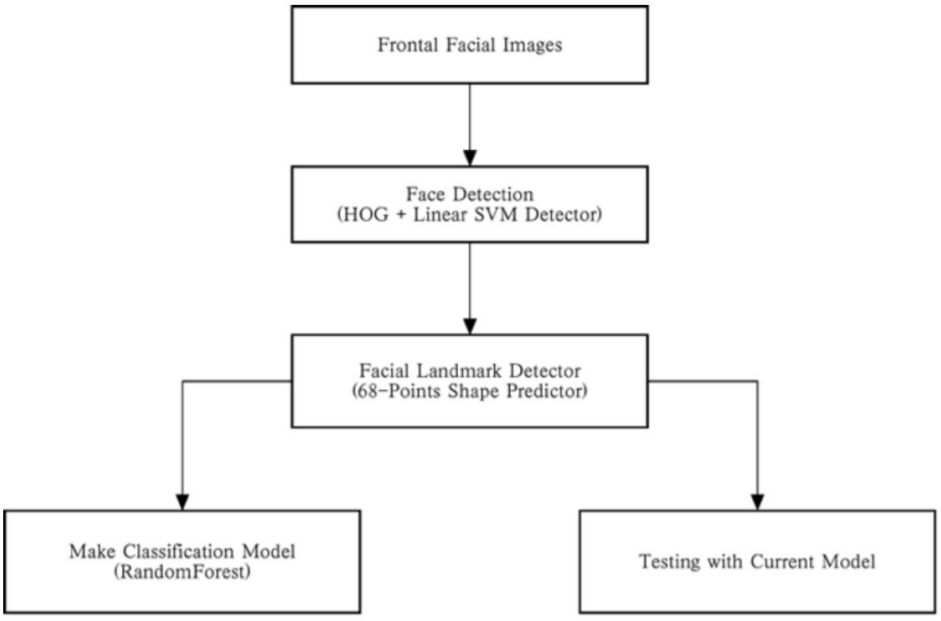
2.2 Software Oriented Architecture (SOA)

SOA is a way of seeing development, breaking down development into specific and generic features, has a significant advantage; that is, so-called generic (and autonomous) services are, in essence, “reusable” by other departments, divisions, and other services. Thus, they will save time and costs in terms of development costs for the company in the future.
Besides, if they are reusable by other services, as in the case of image recognition services, these services will also have good commercialization potential, revenues. Finally, since these are services developed with generic features, they should theoretically create no obstacle in terms of updates since they did not require custom development, unlike “domain-driven” features. This can represent potential substantial savings.

3 OVERVIEW OF ARCHITECTURE

Smart Attendance System with Face Recognition has the architecture as the following step. The camera needs to install in the front which can capture an entire face of the student inside the class. In the first phase after the camera has been captured; the captured image is transferred into the system as an input. The image capture from the camera sometimes comes with the darkness or brightness which needs to do an enhancement on it such as convert to a gray image.

Face detection performs locating and extracting face image operations for face recognition systems using the 68-points shape predictor. The model design of this system will form a blueprint for the implementation that will be put together to achieve the project objectives and best performance for the final product. This system design consists of activities that fit between software requirements analysis and software construction.



The framework of a basic Face Detector

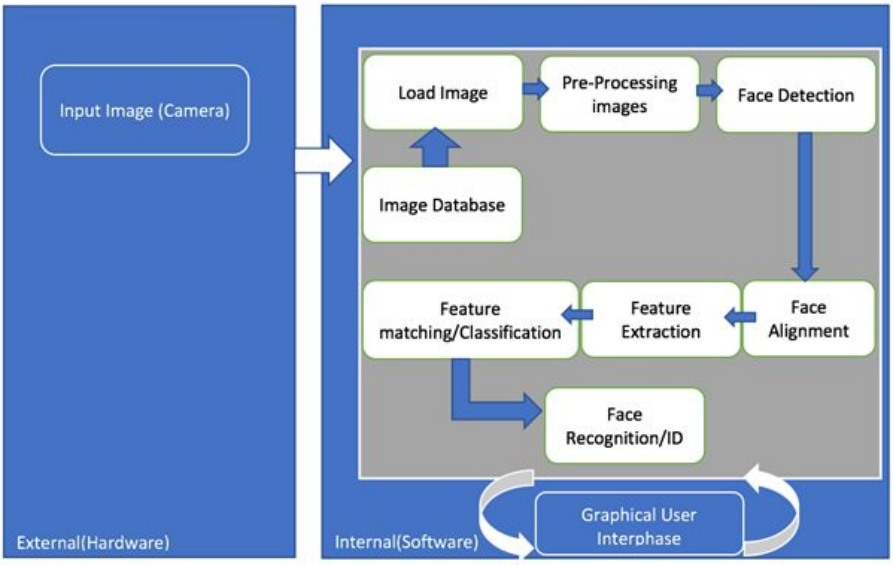
Given below is the python code for recognizing the 68-points in a person’s face using the face-recognition library.

```
import face_recognition
image = face_recognition.load_image_file("your_file.jpg")
face_locations = face_recognition.face_locations(image)
```

This feature allows the model to train the model if the face is not already known to the model, and if it's some known face, the 68-points shape predictor helps to recognize and determine the person. This detail is updated on the .csv file of attendance and thus the attendance is marked.

4 CONCEPTUAL ARCHITECTURE DESIGN

This shows the interaction between software (Internal) components and hardware (External) components with an interface to establish a framework to achieve system objectives. Both external and internal components have been considered. The internal component incorporates all the functionalities with a Graphical User Interface to allow the user to interact with the system.



Conceptual Architecture Design

4.1 Architecture Design:

The Image input and image dataset are all dependencies for the excellent performance of the system. The system will perform best if the images are of good resolution. A good resolution image will enhance face detection to a wider range. Also, a good resolution of the image will have nearly all the pixels required for training the images which will boost the matching of images on the dataset. The accuracy of the system will very much rely on the resolution and quality of the image and how it is trained for recognition.

4.2 Data Storage

Although many methods have been proposed to detect and recognize human faces, developing a computational model for a large database is still a challenging task. That is why face recognition is considered as a high-level computer vision task in which techniques can be developed to achieve accurate results. Few popular methods known for face recognition are neural network group-based trees, neural nets,

artificial neural networks, and principal component analysis. The recognition of the face from an image has numerous applications in Open Computer Vision.

The main challenge of detecting face images is the pose and the illumination variations and sudden changes in the movement of the object. The main challenges of designing robust face recognition algorithms are pose variation, self-occlusion of facial features. The use of Multi-view data to handle the pose variation and its challenges. The proposed system analyzes and recognizes the exact face image from the picture even though there is pose variation and illumination variation while the existing system deals with the recognition of the face images.

4.3 Choice of Methods

The software component that performs this same task is a local database (folder) of images. The image is clicked using an internal functionality and then trained using facial-recognition python library. When the image is processed, the faces in the image are detected and aligned into suitable sizes that are required for feature extraction. The features are then classified and matched to the faces corresponding to that requested by the system or the user. This is then output to the GUI which is part of the internal software components. These functionalities will be used in the algorithms described below for the design and implementation of the system.

4.4 User Interfaces

The design of applications with which the user interacts is called the user interface design. User Interface Design should be designed in such a way that it becomes easy and simple for the user to use. Besides that, it should have a simple feel and look. As a user, the client wants a system where they can load an image that will automatically detect the number of faces on the image. The system should have the option to capture an image using an inbuilt webcam on a laptop. As a user, the system should be able to crop the faces on an image after detection and store them on a folder/dataset that will be used for recognition purposes in the second phase of the system. The system should be able to automatically count the number of faces detected on the image.

The client requests the second phase of the system to be able to match faces stored on a dataset against input images which are either detected from the first phase or captured by an input device (camera). When the face is detected while taking attendance, it is recognized and details of name and time of attendance taken are entered in the database.

5 Conclusion

The proposed algorithm is implemented in Python, utilizing the OpenCV face recognition framework for training and deploying the final models. Since the proposed method requires as input only facial images, a face detector needs to be employed to provide the face regions. In this paper, we've seen the benefits of cloud-based service SaaS (Software as a service). SaaS is a way of delivering services and applications over the Internet that provides a complete software solution that you purchase on a pay-as-you-go basis from a cloud service provider. We've also got a rough idea of how exactly the software is going to work, the steps involved, and the user interfaces. An attendance management system is a necessary tool for taking attendance in any environment where attendance is critical. The attendance system would be able to detect the faces in the image and compare it with the enrolled faces in the database. On identification of a registered student face on the acquired image collections, the attendance register is marked as present otherwise absent.

References

[1] IBM Cloud Team, IBM Cloud, “Advantages of Software as a Service”, 18 September 2020.
[2] Ms. SK. AYESHA, M.Tech, Assistant Professor of E.C.E dept “Face Recognition System With Face Detection ” a Project Report is submitted to Jawaharlal Nehru Technological University Kakinada, (2013-2017).
[3] Ofualagba Godswill and Ise Anderson Orobor, “Automated Student Attendance Management System Using Face Recognition ”, Vol. 5, No. 4, 2018, pp. 31-37, September 13, 2018.
[4] Majid O. Al-Shezawi, Jabar H. Yousif, Ibtisam A. AL-Balushi, “Automatic Attendance Registration System based Mobile Cloud Computing”, International Journal of Computation and Applied Sciences IJOCAAS, Volume 2, Issue 3, June 2017, ISSN: 2399-4509.
[5] Prasanna. J, Senthilkumar. MKS, “Implementation of Biometric Attendance Management System on Cloud Environment”, International Journal of Modern Engineering Research (IJMER), Vol.2, Issue.3, May-June 2012 pp-1052-1054 ISSN: 2249-6645.
[6] Akshay A. Pawle, Vrushsen P. Pawar, “Face Recognition System (FRS) on Cloud Computing for User Authentication”, International Journal of Soft Computing and Engineering (IJSCE) ISSN: 2231-2307, Volume-3, Issue-4, September 2013.