

Smart Attendance System with Face Recognition using OpenCV

Chunduru Anilkumar

Department of Information Technology
G M R Institute of Technology-
Rajam, Srikakulam(Dist.), AP, India
anilkumar.ch@gmrit.edu.in

B Venkatesh*

Department of Computer Science and
Engineering, BVRIT HYDERABAD
College of Engineering for Women-
Hyderabad, Telangana, India
venkatesh.b@bvrithyderabad.edu.in
orcid: 0000-0001-8116-9235

S Annapoorna

Department of Computer Science and
Engineering (AIML), BVRIT
HYDERABAD College of Engineering for
Women- Hyderabad, Telangana, India
annapoorna.s@bvrithyderabad.edu.in

Abstract— A face recognition-based automatic attendance scheme requires the deployment of a system to address the shortcomings of traditional manual attendance. It can also be used as an access control system by registering an organization's personnel or students with their faces, and then recognizing them by taking their photos with faces and marking their attendance in the database. The face recognition module is built upon the dataset “labelled faces in the wild” which got an accuracy 99%. While observing the face recognizer it is concluded that the less number of faces can be recognized very accurately and in less time. Open-CV is an open-source image processing package that can be used in a variety of applications. Flask is a python-based web application micro-framework enabling easy app management. The system is built using the Python Open-CV module and a Flask web application. It begins by detecting faces in images captured from a web camera and comparing them to images of pupils who have previously been enrolled. New users or students can register by providing accurate information and a photograph. A member's existing information can be managed by an administrator. The web application may retrieve detailed attendance reports for a class on a certain date from the database. For quick access, this system will combine face recognition technology with an automated web application sys-tem.

Keywords: *OpenCV, Face_Recognition, flask, Attendance system, MySQL*

I. INTRODUCTION

The attendance system is one of the major aspects in every educational institute and organization in order to keep track of students' class presence and the working hours of staff or an employee. Manually attendance-taking procedures may not be reliable sometimes because of human errors. An automated system will save time and money for the organization. Many institutes are switching to an automated attendance system by using digital technologies and biometric authentications. But these biometric systems also can be served only one person at a time. It is already known that face recognition models are improving a lot, by using many deep learning techniques. These face recognition models can be used as a feature in an attendance system. Face recognition technology is used by many public and private institutions for different purposes. It is used for authentication at entry gates, identifying the criminals in a crowd, Liveness detection and many more applications are there. In this proposed system face recognition model is used which built upon the “labelled faces in the wild” dataset which comprises of different faces with corresponding labels. This system uses state of art of deep metric learning and transform each image as a 128 bit encoded vector and then uses SVM model for classification task. The

literature review, full description of several stages in the proposed model, results and conclusions, and scope for improvement are the sections that follow in this study. The proposed system will have the benefits like time-saving, cost reduction of stationary materials, easy reporting, and reduced administrative load.

II. LITERATURE REVIEW

Face recognition is a computer vision technology that analyses facial feature information for identity recognition. Experimental data shows that the accuracy rate of the video face recognition system is up to 82%. Compared with the traditional check-in method, the face recognition attendance system can be reduced by about 60% [1]. Many face recognition systems needs the input image in a certain conditions such as controlled illumination, particular posing and view angle, and with no obstacles in between face and camera. These are known as face recognition under controlled conditions [2]. These rules restrict the uses of face recognition in many real time applications because they cannot satisfy these rules. Real time applications need techniques which does not need any strict control over the human beings for recognizing the face [3]. These types of systems need face recognition under uncontrolled conditions. So, this paper proposes one such system but the system needs an image as input and one person per image which is a drawback of the system and provides a hindrance in using it in real time applications like attendance systems [4]. Biometrics systems essentially consist of recording and comparing biometric characteristics. When a person first uses a biometrics system, his or her identifying features are enrolled as a reference for future comparison. This reference can be stored in laptop/ server for future use [5]. The Open source technology Open face was used to create this real-time GUI based facial recognition system. Open Facial is a face recognition Programme based on Open CV developed by Carnegie Mellon University. HOG is one of the most used ways for representing a face image [6]. This study discuss about the automated attendance monitoring system without human intervention in this work. In this method, a camera is installed in the classroom, which captures the image, detects the faces, compares them to a database, and then records the attendance. Eigen faces are a collection of Eigen vectors used in the face recognition problem in computer vision [7]. In the author incorporated NFC technology, Raspberry Pi and 3 Face Authentication to be implemented on the mobile device [8]. In authors proposed an automated attendance management system. This system, which is based on face detection and recognition algorithms, automatically detects the student when he enters the class room and marks the attendance by recognizing him [9].

Currently, the attendance tracking system for education sectors based on biometric recognition is underutilized, presenting an excellent chance to do fascinating research in this field [10]. The web is the most widely used networking tool that meets the needs of all sorts of users and gives a solution to any problem definition. To build and develop a web that is well-structured and attractive, we must use the appropriate technologies [11]. An image of a face is divided into patches, which are then concatenated in raster scan order to create an ordered sequence. DICW calculates the image-to-class distance between a query face and those of an enrolled subject using this order information by determining the best alignment between the query sequence and all sequences of that subject [12].

TABLE1: COMPARISON OF DIFFERENT TECHNIQUES USED IN FACE RECOGNITION BASED ATTENDANCE SYSTEM

Ref No.	Author & Title	Concept used	Advantage	Limitation
1	H. Yang and X. Han, "Face Recognition Attendance System Based on Real-Time Video Processing"	Real time video processing	More stable and accurate in different conditions	Large computations and hardware equipment required
2	Xin Geng, Zhi-Hua Zhou, & Smith-Miles, "An Approach to Face Recognition Under Uncontrolled Conditions."	Face Recognition under controlled conditions	Better result in real time conditions	Video sequences verification and multiple person images recognition at real time is difficult
3	Patil, Pratima, Ajit Khachane, and Vijay Purohit. "A wireless fingerprint attendance system."	Finger print authentication based attendance system	Secure authentication	Time consuming
4	D. M. Prasanna and C. G. Reddy, "Development of Real Time Face Recognition System Using OpenCV"	access control system by face recognition using Histogram of gradients	More accurate on local images data	Inappropriate for large no. of samples in dataset.
5	E Varadharajan, "Automatic attendance management system using face detection"	Background subtraction of image and eigen weights for face detection	The face detected after background subtraction is accurate as compared to an image which is not background subtracted.	Face detections is very accurate but Face recognition task is difficult

6	Siti Umami Masruroh "NFC Based Mobile Attendance System with Facial Authorization on Raspberry Pi and Cloud Server"	Iris Recognition	Real time face detection and efficient	Difficult to work with low illumination
---	---	------------------	--	---

III. METHODOLOGY

The proposed system consists of the following functionalities:

1. Taking attendance based on Face Verification.
2. New Student Registration into the System.
3. Retrieving previous Attendance Report

The system is developed in 3 layers and each layer is responsible for different functionality

Application Layer: For an easily accessible interface.

System Layer : The functionality of the system.

Database layer : For storing and maintaining persistent data.

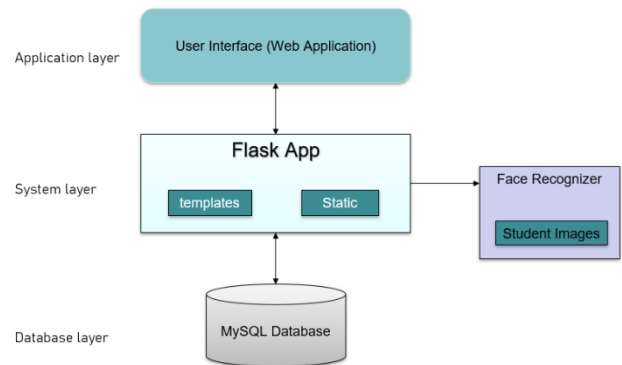


Fig. 1. System Overview.

A. Application Layer

The user interface is made by using html along with styling applied by CSS and Bootstrap. The interface is mainly consisting of three sections. Attendance, Enrolment and Attendance Report. The index page is made by using HTML and styling applied by using CSS and Bootstrap framework. The application layer is made for a user friendly access of the system for the user. User is not concerned with what actually happening in the back, he will use the application easily from this interface.

a. Take attendance

Here the face recognition model that is built pre-trained is used to detect and classify faces in front of camera. This section is responsible for taking attendance of a class students by face verification and send the recognized faces data to the flask runtime which will store the data in database. This will take 4 inputs from the user those are Branch, Year of study, Section, Subject. After those four inputs the recognizer will be called and starts verifying faces.

b. Enrolment of New Student

This section is responsible for adding new student's

information to the system with given details in the registration form. The form data will be sent to the flask application management.

c. Attendance Report

A hyper link is displayed in the home page to get the attendance report of a class on a given date. This hyperlink will redirect to the "report" page. The report page has a form which consists of 4 inputs such as Branch, Year, Section, Subject and Date. The attendance report of the class based on given data will be displayed in this page.

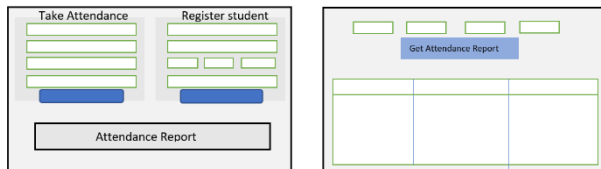


Fig. 2. Basic layout of home page and search page.

For front end web page design we used the HTML, CSS and Jinja2 template coding.

a. HTML & CSS

Hypertext means the links provided in a page and markup text is used for structuring the content of the web page by using Tags that are available in the HTML. The information is visually shown to the user by different elements on the web page [13]. These elements are then styled by using different Style sheets and frameworks.

Styling can be applied to the HTML by using CSS which provides attractive styles and transformation for a more user friendly interface through the web page. Cascading Style Sheets are applied to html pages by including the css files as meta links the head tag of the html.

b. BOOTSTRAP

Bootstrap is a front-end web development framework which comprises of CSS predefined classes and JavaScript validations and extensions. It allows us to make a web interface look stylish and proper validations. Bootstrap allows Responsive web design for different platforms like mobile, Laptops, PCs. It uses a grid system which facilitates the programmers to make web site responsive according to viewport [14].

Bootstrap is used by using CDN meta links in head of the html page or we can also use the vendor files for applying bootstrap. However in this project we are using the CDN links and bootstrap classes of Forms and Cards and buttons required for the website styling design.

c. JINJA2

Jinja2 is a powerful templating language popular in the Python community. It may be used directly in Python programs, and it's the template rendering engine for many larger projects [15].

Templating languages enable the development of text-based

documents with dynamically created content. The resulting files can be HTML, JSON, XML, or any other text-based format. The goal is to embed business logic in code while giving template designers control over the flow and layout of the final document.

B. System Layer

a. Face Recognizer

The face recognizer is made by using OpenCV-python and face_recognition library in python. It starts capturing faces and if the face is recognized then it will add the id to a list. And then finally it returns the list of verified ids.

Following libraries are used in the task of recognizing face.

1. OpenCV
2. face_recognition
3. dlib

- OpenCV is a programming package geared mostly at real-time computer vision. It was created by Intel and then sponsored by Willow Garage and Itseez. Under the open-source Apache 2 License, the library is cross-platform and free to use.
- We'll be utilizing the face recognition library in our code because it's quite straightforward to use. Before we use face recognition, we must first install dlib.
- The dlib library provides our implementation of 'deep metric learning,' which is used to build the face embeddings that are employed in the recognition process.

The face_recognition module has the following functions:

1. face_locations

Returns an array of human face bounding boxes in an image.

2. face_encodings

Return the 128-dimension face encoding for each face in an image given an image. On these encodings, we perform a comparison.

3. face_distances

Compare a list of face encodings against a known face encoding and calculate the Euclidean distance between each comparison. The distance between the faces indicates how similar they are.

4. compare_faces

To see if they match, compare a list of face encodings to a candidate encoding. Based on the comparison, return True or False.

Working of recognizer

- i. Go to directory where all known images are available.
- ii. Store all the names available in the directory as class Names.
- iii. Find encodings of all known images available and store them in a known Encodings list.

- iv. While True:
- Read each frame from source video and scale down by resizing
 - Find all face locations in the frame.
 - Find face encodings for all faces in the frame and store them in current Encodings
 - Compare known Encodings with current Encodings
 - Find face distances between known Encodings and current Encodings.
- v. When the user press on 'S' key close the window and return all the recognized names list.

Flask Application

Flask is a Python Application Programming Interface that allows us to create web-based applications. A web application can be easily managed by using flask.

It will bind different URLs of a web site and defines each functionality to the each page with python programming. It integrates the python language with web programming. We can process the data in python and send them to the web interface. Flask is built using WSGI toolkit and Jinja2 Template engine.

Routing:

A route decorator in flask is used to redirect to target URL. The flask will define each function in python is to this URL.

Data retrieval through HTTP requests:

For data retrieval from the supplied URL, Flask supports a variety of HTTP protocols.

GET: This is used to deliver data to the server in a form that is not encrypted.

POST: This method sends the form data to the server. The server does not cache data received via the POST method.

The post request method returns post data, such as information entered into a web form. This information is organized as an associative dictionary, with form input names serving as keys and entered values serving as dictionary values.

File-Uploading in Flask:

Uploading files in Flask is a breeze. It requires an HTML form with an enctype attribute and a URL handler to retrieve the file and save it to the specified location. Files are temporarily kept on the server before being sent to the desired location.

Template Inheritance:

We can re-use the code that is written in some base html template.

Before running flask application it is better to create a virtual environment which isolates the project.

This system consists of following end points

1. Root ('/')

2. Take Attendance ('/take attendance')

3. Add Student ('/addstudent')

4. Attendance Report ('/search')

home (/):

This will be the base URL that is opened directly when we open the web server (localhost). On this page Index.html template is rendered. This consists of basic interface without any messages.

/takeattendance :

This function is responsible for calling a recognizer and perform actions on database. At the end this function will render the index.html with the message whose attendance has been marked. Those ids will be displayed on index page when attendance marking is done.

/addstudent

This function is responsible for adding new student into the system by performing SQL queries and it uploads the student pictures in a given directory.

/search

This URL call the function that is responsible for retrieving attendance details on given date and finally render the search.html page with the retrieved data from database.

The run() function is used to start the Flask application. For any code changes, the method should be manually restarted. To combat this, debug support is turned on, allowing you to track out any errors.

Database Layer:

MySQL is a Relational database management system (RDBMS) based on structured query language (SQL).

This project's database system consists of totally 2 Relations:

Students Table:

Students table is responsible for storing the details of students with Student ID as primary key. It is accessed by the enrolment functionality of the system to insert new student's data. Students (Id, Name, Branch, Year, Section)

StudentID	Name	Branch	Year	Section
19341A1277	M V Ram Kumar	IT	3	B
19341A1284	P Hari Shankar	IT	3	B
19341A1295	Routhu RamaKrishna	IT	3	B
19341A1296	Routhu Sathish	IT	3	B
19341A1297	G Sai Ram Prudhvi	IT	3	B

Fig 3: Students table in MySQL server

Attendance Table:

Attendance table is used for storing the data of attendance of a class. The list of student's ids will be stored as a long Text value

in the database. This will have a primary key as current **TIMESTAMP**. It is accessed by the “take attendance” functionality of the system. The structure of the table is:

Attendance (time, date, year, branch, section, sub, presents)

TIME	DATE	YEAR	BRANCH	SECTION	SUB	PRESENTS
2022-05-09 10:59:43	2022-05-09	3	IT	B	6	19341A1266 19341A1276 19341A1277 19341A1278 19341A...
2022-05-09 19:59:43	2022-05-09	3	IT	B	1	19341A1297 19341A1298 19341A1299 19341A12A0 19341A...
2022-05-10 12:12:05	2022-05-10	3	IT	B	3	19341A1296 19341A1284
2022-05-12 22:25:48	2022-05-12	3	ECE	B	1	19341A04D8 19341A04D9 19341A04E2
2022-05-12 22:29:23	2022-05-12	3	ECE	B	3	19341A1296 19341A12B7 19341A1295 19341A1277

Fig 4: Attendance table in MySQL server

SQL Queries:

Enrolment of new students can be done by adding data into students table with given details of student and executing following SQL Query.

Insert Into Students Values (Student Id, Name, Branch, Year, Section)

For taking attendance of a class students, the flask application will invoke the recognizer and get the ID's of students whose face has been recognized. This list of ID's is converted into a String format and inserted into the database.

Insert Into Attendance (Year, Branch, Section, Sub, Presents) Values (3, 'It', 'B', 'Sub-5', '19341a1296 19341a1297 19341a1277')

To retrieve the attendance of a date the ids presents list is taken from the database on that date. In the web page all the student's details are displayed with their ID, Name and present/absent Status. The present or absent status is determined by the list of IDs taken from attendance table.

This data displaying process will be following way:

Let the Students List is the details of all students taken from the Students table in database and Presents List is list of attended IDs taken from the Attendance table.

for each student in Students List:

If student.id in Presents List:

SET Status = present

Else:

SET Status = absent

Display Student details: ID, Name, Status

Work Flow:

The workflow of the system can be described as follows:

The user can run the flask application using python commands.

The flask applications tries to connect to the MySQL database Server located in Xampp if it is connected then the interface is displayed on the home page. The user have to choose which function he want to use now. For example if a new student has to be registered in the system we have to go to the section of New Enrollment and then fill the necessary details to submit details. The internal working will not be depicted to the user.

This way the user able to take the attendance also by giving necessary details regarding the class room. If the user wants to get the report he can click on the hyperlink provided in the home page. After redirecting to the home report page user has to give the details about which class and on which date he wants the report.

IV RESULTS

The proposed system has developed as a web application and several testing are made by using different faces placing in front of camera. The correctness of the system results are given below:

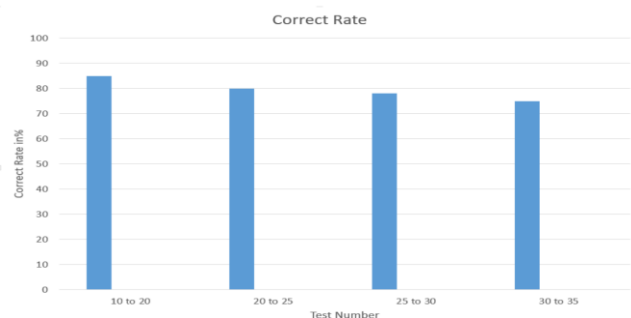


Fig 5: Correctness of face_recognition

We observed that there are mismatch results when the no. of people in front of camera increases.

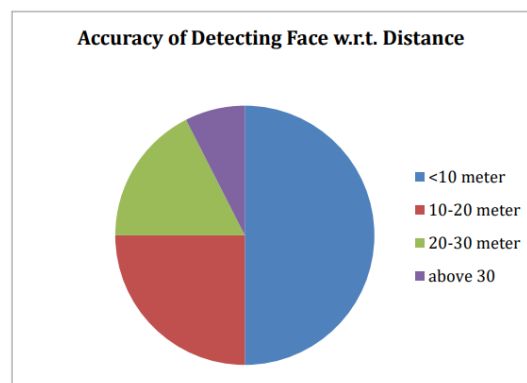


Fig :6 Accuracy w.r.t distance between face and camera

The web application results are shown below:

Home page:

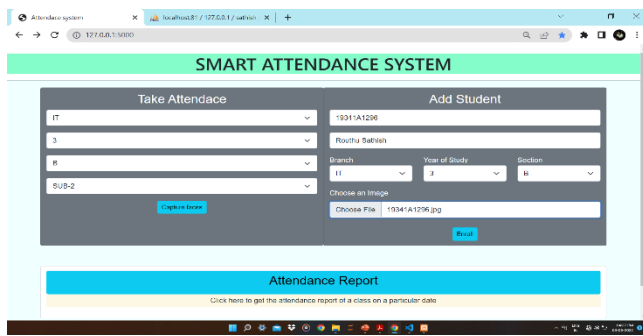


Fig 7: Homepage of the System

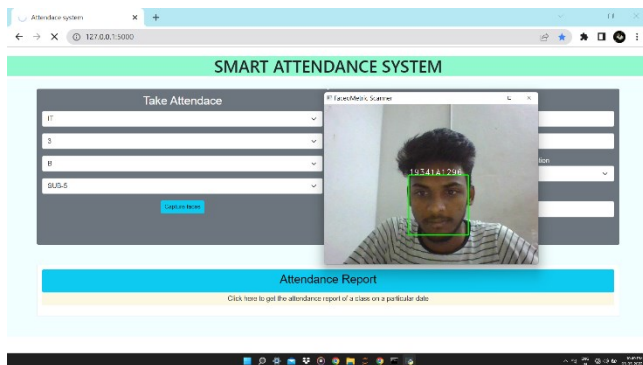


Fig 8: Recognizer verifying face

There can be more than one face in front of the camera. All those students attendance will be marked one they were recognized.

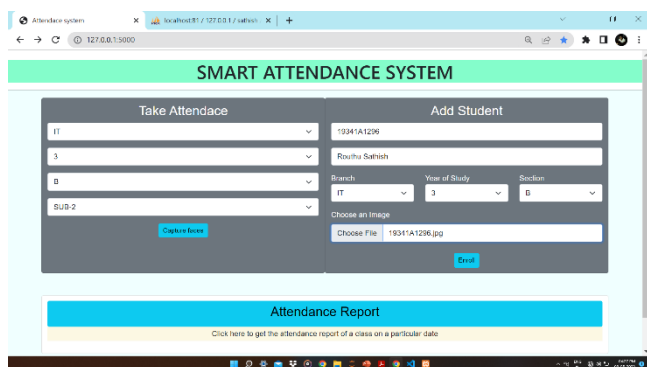


Fig 9: New Student Registration

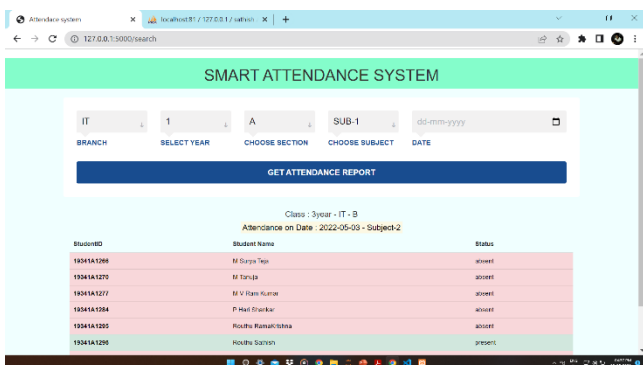


Fig 10: Attendance report on a date

V.CONCLUSION

This study has integrated a face recognition system with a web application so that it can be useful in the process of the attendance management system. The web interface provides easy accessibility to the user. This automated system can replace traditional attendance procedures by overcoming many problems occurred during the manually attendance-taking procedure and saves a great amount of time. This system can reduce the paperwork and stationary material cost. The attendance records of previous are available immediately from the database. This system will also ensure that the attendance is reliable and under the supervision of the administrator. There are still limitations in this system such as low light conditions and posing of the face by student who is in front of camera. One more disadvantage is to the recognizer cannot differentiate a real human and a picture of a user.

VI FUTURE SCOPE

The system can be further developed by including other features in the system. The face recognition module in the system can be improved by training it on the own dataset of college students.

VII REFERENCES

- [1]. H. Yang and X. Han, "Face Recognition Attendance System Based on Real-Time Video Processing," in *IEEE Access*, vol. 8, pp. 159143-159150, 2020.
- [2]. Xin Geng, Zhi-Hua Zhou, & Smith-Miles, K. (2008). Individual Stable Space: An Approach to Face Recognition Under Uncontrolled Conditions. *IEEE Transactions on Neural Networks*
- [3]. Patil, Pratima, Ajit Khachane, and Vijay Purohit. "A wireless fingerprint attendance system." *International Journal of Security, Privacy and Trust Management* 5.4 (2016).
- [4]. D. M. Prasanna and C. G. Reddy, "Development of Real Time Face Recognition System Using OpenCV," *International Research Journal of Engineering and Technology*.
- [5]. E. Varadharajan, E., et al. "Automatic attendance management system using face detection." *international Journal of green engineering and technologies* IEEE, 2016.
- [6]. Masruroh, Siti Umami et al. "NFC Based Mobile Attendance System with Facial Authorization on Raspberry Pi and Cloud Server." 2018 6th International Conference on Cyber and IT Service Management (CITSM) (2018): 1-6.
- [7]. Chintalapati, Shireesha, and M. V. Raghunadh. "Automated attendance management system based on face recognition algorithms." 2013 *Computational Intelligence and Computing Research*. IEEE Access, vol 12, 2013.
- [8]. Rathod, H., Ware, Y., Sane, S., Raulo, S., Pakhare, V., & Rizvi, I. A. (2017, January). Automated attendance system using machine learning approach. *Journal of Nascent Technologies in Engineering (ICNTE) (2017) IEEE Access – vol-8 , (pp. 1-5).*
- [9]. Lokhande, P. S., Fankar Aslam, Nabeel Hawa, Jumal Munir, and Murade Gulamgaus. "Efficient way of web development using python and flask." (2015). Volume 6, No. 2, March-April 2015 *International Journal of Advanced Research in Computer Science*.
- [10]. Wei, Xingjie, et al. "Dynamic image-to-class warping for occluded face recognition." *IEEE Transactions on Information Forensics and Security* 9.12 (2014): 2035-2050.
- [11]. H. J. Seo and P. Milanfar, "Face verification using the LARK representation," *IEEE Trans. Inf. Forensics Security*, vol. 6, no. 4, pp. 1275–1286, Dec. 2011.

- [12]. G. Sharma, S. ul Hussain, and F. Jurie, "Local higher-order statistics (LHS) for texture categorization and facial analysis," in Proc. 12th Eur. Conf. Comput. Vis. (ECCV), vol. 7578. 2012, pp. 1–12
- [13]. J. Ruiz-del Solar, R. Verschae, and M. Correa, "Recognition of faces in unconstrained environments: A comparative study," EURASIP J. Adv. Signal Process., vol. 2009, pp. 1:1–1:19, Jan. 2009
- [14]. D. Keysers and W. Unger, "Elastic image matching is NP-complete," Pattern Recognit. Lett., vol. 24, nos. 1–3, pp. 445–453, Jan. 2003.
- [15]. X. Wei and C.-T. Li, "Fixation and saccade-based face recognition from single image per person with various occlusions and expressions," in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW), Jun. 2013, pp. 70–75