International Journal of Environmental Sciences
ISSN: 2229-7359
Vol. 11 No. 6, 2025
https://theaspd.com/index.php

Face Detection and Recognition Student Attendance Systems

Naga Durga Saile. K[1], Dr. Sreenivas Mekala[2], Dr. T. Ravi[3], Dr. Ashish[4], Mr. Sachin Chandravadan Karad[5], Poreddy Jayaraju[6], Bhupesh Deka[7]
[1]Assistant Professor, Department of CSE-AIML & IoT, Vallurupalli Nageswara Rao Vignana Jyothi Institute of Engineering and Technology, Hyderabad, Telangana, INDIA.
[2]Associate Professor, Department of CSE, School of Agricultural Engineering &Technology, Kaveri University, Gowraram(v), Wargal(M), Siddipet(D), Telangana, INDIA.
[3]Professor, Department of Computer Science and Engineering, Vel Tech Rangarajan Dr. Sagunthala R&D Institute of Science and Technology, Avadi, Chennai, Tamil Nadu, INDIA.
[4]Associate Professor, Department of CSIT, Koneru Lakshmaiah Education Foundation (K L University), Vaddeswaram, AP, INDIA.
[5]Freelancer Coach & Consultant, Independent Researcher, IEEE Senior Member, Village: Rameshwar, Taluka & District - Latur, Maharashtra, INDIA
[6]Assistant Professor, Department of IT, GNITC, Hyderabad, Phd Scholar, IIT Bhilai, Computer Science Department, INDIA
[7]Assistant Professor, Department of CSE-AIML & IoT, Vallurupalli Nageswara Rao Vignana Jyothi Institute of Engineering and Technology, Hyderabad, Telangana, INDIA.
[1]saileknd3@gmail.com, [2]mekala.sreenivas@gmail.com, [3]drravit@veltech.edu.in,
[4]dr.aashishsinha@gmail.com, [5]sachin413510@gmail.com,
[6]jayarajudavid@gmail.com,[7]bhupesh_d@vnrvjiet.in

Abstract
Maintaining attendance manually is a challenging, time-consuming, and error-prone process. An intelligent and automated system for managing attendance can be implemented using various biometric technologies, with face recognition being one of the most influential and non-intrusive methods. In every organization, particularly educational institutions, accurate attendance tracking is essential to record the presence of individuals efficiently. Our face recognition project detects and identifies the faces of students, helping to save valuable time compared to traditional methods. Furthermore, the system generates a .csv file, allowing lecturers to maintain an organized, easily accessible daily attendance record. This not only ensures the accuracy of attendance tracking but also simplifies the analysis of attendance trends, promoting accountability and improving overall administrative efficiency. In addition, the automated system reduces the possibility of human error and tampering with attendance records. With its real-time tracking capabilities, the system can also provide instant reports and alerts, enhancing overall classroom management. By integrating face recognition technology, institutions can streamline operations, leaving more time for educational and developmental activities.
Keywords: *Image enhancement, MTCNN, P-Net, R-Net, O-Net, FaceNet, CSV file, Face detection, Face recognition, FAISS, image embeddings, attendance management*

1. INTRODUCTION

In a classroom setting, taking attendance can be one of the most time-consuming and tedious tasks for a lecturer, especially in a class of 70-80+ students. Manually keeping handwritten records makes it difficult to compute and analyse statistical data, and this method is more prone to errors such as false attendance or proxy entries. To address these issues, several alternative methods for identifying individuals have been proposed. Some of the most effective solutions include scanning ID cards, using fingerprint sensors, and implementing face recognition systems.

Face recognition technology can recognise or verify a person's identity by mapping their facial features from images or videos using advanced biometric techniques. Face recognition in today's digital world is proving to be an invaluable tool, particularly in streamlining processes like attendance verification. While many institutions have embraced technology to automate this process, some still rely on outdated, manual methods that are time-intensive and inefficient. Face recognition not only speeds up the process of recording attendance but also enhances accuracy, reducing the likelihood of errors or fraudulent entries. Moreover, the system can integrate seamlessly with databases to generate real-time reports, making it easier for educators to track attendance trends and monitor student participation. As technology continues to evolve, face recognition is expected to become a standard tool in classrooms, improving operational efficiency and minimizing manual effort.

2. Related Work

2.1. Existing Methods

Traditional attendance systems used in educational institutes make use of a mix of technologies and approaches to track/manage student attendance. Let's take a standard case of such a system: Traditional Methods. Before modern technologies came into existence, most of the schools were following manual methods for taking attendance by calling roll, using sign-in sheets, or swipe cards. Although these methods of detection are still implemented in some locations, they have gradually become outdated due to their potential for error and long time consumption. RFID/NFC-Based Systems: In some institutions, attendance is tracked using RFID or NFC technology. Students are given RFID cards or tags, which they must swipe or tap on a reader when entering the classroom. This automatically records their attendance in the system.

Biometric Attendance Systems: This type of biometric attendance system identifies students by unique physiological features such as fingerprints, facial features, or patterns in the iris. They have to scan their biometrics at the school with dedicated machines for entry. They have great accuracy and are very difficult to be biased.

2.2. Literature Survey:

A PCA and eigenface-related attendance system is presented in this paper to tackle spurious attendance and proxy problems. It is claimed that the system is highly invariant to illumination changes, and it can help improve attendance management efficiency. But the system does not work well for other head poses and lighting conditions, also it is very sensitive to glass-wearing Entity [1]. The attendance system proposed in this paper is based on face recognition using Haar cascade for detecting the face, and a Local Binary Pattern Histogram (LBPH) algorithm for recognizing the face. The system is said to be less sensitive to changes in illumination and head pose, and could identify a person after they have grown a beard or put on glasses. The system has an approximately low identification rate (60%) of unfamiliar persons and may be dependent on the training set [2]. In this work, we evaluate the performance of various machine learning algorithms (Support Vector Machine (SVM), MLP, and CNN) for attendance systems based on face recognition. SVM and MLP extract features from PCA and LDA; on the contrary, CNN directly utilizes images as input. The results indicate that CNN achieves the best accuracy with approximately 98% followed by SVM and MLP, achieving an accuracy of 87% and 86.5% respectively. But the accuracy of the system may be influenced by camera quality and light conditions [3]. Abstract: The authors recommend a smart attendance system based on face recognition based Local Binary Pattern Histogram (LBPH), and DNN (Deep Neural Network) algorithms. It attempts to reduce proxy accounts, and it sends messages generated by GSM.UPRIC has compared the attendance from the teacher's mark, which can be used more easily for most of the applications. We apply the LBPH algorithm for face recognition and the DNN for face detection. LBPH can be quite slow on large datasets or high-res images, and it can be sensitive to noise in the images [4]. This paper describes an automatic attendance maintainer by implementing the techniques of image processing for face recognition and detection. The proposed system employs the Viola-Jones algorithm for face detection and Fisher Face for recognition. There is a system to avoid these manual and time taking and paper-wasting attendance systems. But, the system is only 45% to 50% accurate, which may not be reliable enough for attendance management purposes [5]. In this paper, we have designed an automatic attendance system based on face detection with the help of Haar cascade and AdaBoost classifiers. It attempts to minimize the precision mistakes due to manual sign systems and also aspires to save time and strain during attendance marking. It handles an immense database of images and is capable of capturing images accurately at 80cm. The system, though effective in tracking faces, may not have the ability to cope with the contrast of lighting or expression variation [6]. A Smart Attendance System based on Face Detection and Recognition using OpenCV. For attendance data analysis and storage, the system uses Python Integrated Development & Learning Environment (IDLE) for coding as well as Microsoft Excel. We designed the system to automate attendance tracking and minimize manual entry. However, the system's accuracy can be affected by factors like variations in lighting conditions, facial expressions, and occlusions [7]. The face recognition algorithms such as Eigenface, Fisherface, and LBPH are compared using OpenCV in this paper. These results indicate that Eigenface is more resistant to changes in lighting conditions, whereas LBPH excels at dealing with alterations in facial expressions and poses. Nevertheless, Eigenface needs more disk space [8]. Since many eigenfaces and coefficients must be stored. The proposed model represents a real-time implementation of an automatic attendance system by using YOLOv3 for face detection and Microsoft Azure Face API for the recognition. CMM Attendance, aimed at minimizing manual attendance in colleges using mobile phones. Cameras take pictures at the beginning and end of classes, logging both student presence and verifying faces -- known or unknown? Then it spits out spreadsheets and sends monthly emails to students, parents, and faculty, improving attendance tracking efficiency. However, the system relies on external services like YOLOv3 and Microsoft Azure, which might introduce limitations and dependencies beyond the institution's control [9]. This study describes a face recognition-based attendance system with the Fisherfaces method. This system is built with the goal of auto attendance, which also provides accuracy and efficiency to track better attendance by implementing real-time face detection and recognition for attendance. Nevertheless, Fisherfaces has its drawbacks as well and is not very effective when sharp transitions occur in the lighting of an image, which may be considered a distinguishing feature for face recognition [10].

2.3. Research Gap Identified:

However, some technical challenges are observable during the implementation phases, like lighting conditions, image quality, Target detection Algorithm Selection, and external service integration.

1. Sensitivity to Environmental Factors: These systems are sensitive to variations in light intensity, head poses, facial expressions, occlusions, and background clutter.
2. Accuracy and Performance Limitations: The accuracy of these systems may be limited, with recognition rates ranging from 45% to 60%. Additionally, algorithms like LBPH may suffer from slow processing times, especially with large datasets or high-resolution images.
3. Range Limitations: Systems may have limitations in terms of the distance at which they can accurately capture facial images. For example, accurate capture may be limited to a distance of up to 80cm.

3. Proposed Method

Our proposed system creates embeddings of each student's face and stores them in a database for their attendance. There is no need for the teacher to manually take attendance as the system automatically detects and recognizes the faces of students in the image. Attendance details are then stored in a spreadsheet automatically. Extended Range: We will be improving the range limitations. It can able to detect students at various distances (close to moderate distances). It can also be able to handle Minor tilts or Rotations well.

3.1. Dataset Preparation

The dataset for our project was collected manually by us and consists of 80-100 images of each person from different angles to ensure that the system can detect faces with minor tilts.

The Structure of the dataset is:

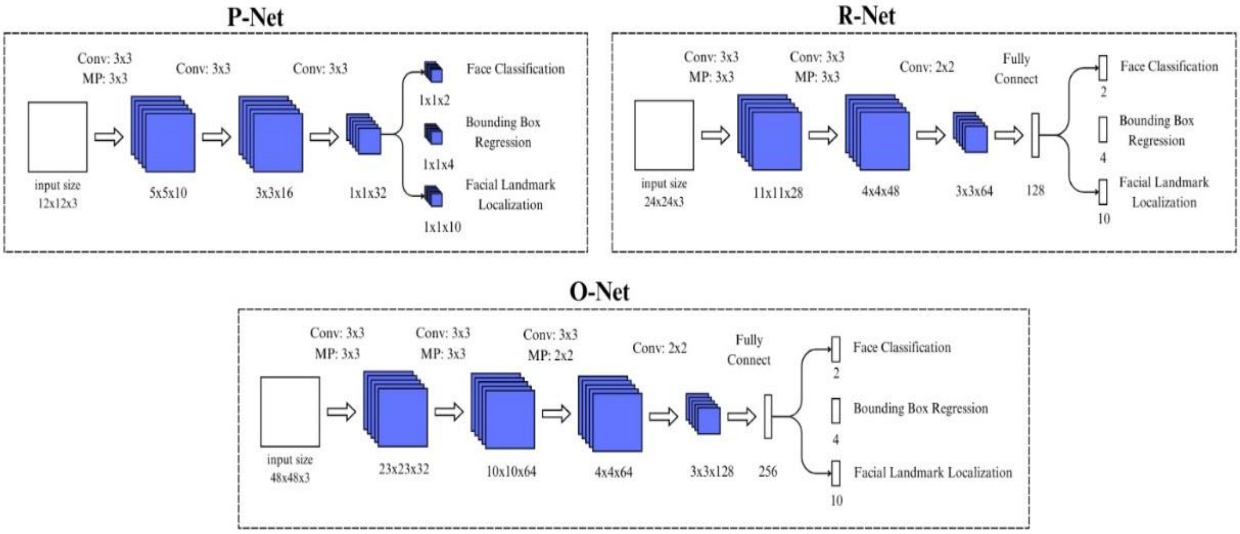
```
-face images: (folder name)
    -student1name:
        -img1
        -img2......
    -student2name:
        -img1
        -img2.......
    ...........
    ...........
```

3.2. Model Architecture

The model we used for face detection is MTCNN, and for face recognition, we used FaceNet.

- MTCNN - Multi-Task Cascaded Convolutional Networks:

MTCNN (Multi-task Cascaded Convolutional Networks) is a deep learning model designed for face detection and alignment. It is made up of three connected CNNs working in a cascaded fashion. The initial network generates potential facial areas, followed by a region-of-interest (ROI) warping image aligned with the face, and lastly conducts facial landmark detection. Thanks to MTCNN's orientation and lighting face variation acceptance, MTCNN is very robust for face detection. It addresses hundreds of thousands of apps and delivers accurate and Trustworthy face detection. MTCNN (Multi-task Cascaded Convolutional Neural Networks) is a face detection algorithm through which an image containing one person will go through three steps: P-Net, R-Net, and O-Net. This is what happens at each level:



1. **P-Net (Proposal Network):**

- It would first generate preliminary candidate regions, which would be some bounding boxes. P-Networks on a lower resolution, and thus, the input images are down-sampled at multiple scales.
- The PNet scans the input image first to search for facial features in every area of the image. For each candidate region, P-Net outputs bounding boxes and regression vectors for fine-tuning the locations. However, there are overlaps or redundancies between some of them, so NMS is applied to them.
- The P-Net proposals for the candidate face regions are forwarded to the R-Net to refine the bounding box.

2. **R-Net (Refinement Network):**

- This suppresses R-Net, which eliminates false positives by processing each candidate face region at a higher resolution and doing further checks.
- The R-Net processes each candidate face region at a higher resolution and performs more detailed checks, filtering out false positives.
- It outputs refined bounding boxes and regression vectors to adjust the boxes' size and location. Again, NMS is applied to remove overlapping boxes.

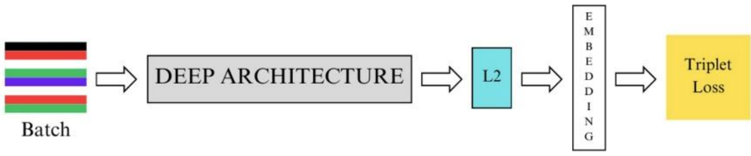
3. **O-Net (Output Network):**

- Finally, the candidate regions from the R-Net are passed to the O-Net, which performs the most detailed and precise face detection.
- The O-Net refines the bounding boxes and also outputs five facial landmarks (eyes, nose, and mouth corners) for more accurate localization of facial features.
- After further NMS, the final bounding box representing the face is produced, along with the associated landmarks.

By the end of these stages, the MTCNN model detects the face in the image and returns an accurate bounding box along with vital facial landmarks optimized for the input image size.

- Facenet

FaceNet is a CNN-based approach to face recognition. It constructs representations of face images as 128-dimensional vectors in deep networks. FaceNet computes real-valued embeddings in such a way that the distances between faces across different images reflect the amounts of change in their appearances. FaceNet makes use of the triplet loss function while training the faces of the same people close to each other in the face space.
Such areas where the faces of different individuals are far away from each other. Model architecture, which consists typically of network structures similar to ResNet or Inception, provides the ability to be insensitive to variations of lighting, pose, and
The FaceNet has been widely adapted in verification, face clustering, and real-time face recognition tasks.



Batch: Images are processed in lots for efficiency.
Deep Architecture: It extracts features from the images using a stack of layers.
L2 Normalization: Rescale the feature vectors or embeddings so that each has a length of unit.
Embedding: It provides unique vectors for each face.
Triplet Loss: trains a model ensuring that embeddings of the same person are close together, but people are galaxies apart.

- System Architecture

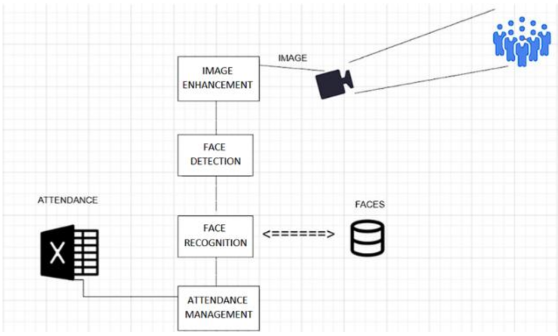


Fig 1: System Architecture

Based on the above figure, it has demonstrated a proposed workflow for image enhancement, face detection, face recognition, and Attendance Management has been demonstrated. In our work, this was implemented using Python 3.The process, termed the image Enhancement, rescales an input image to a size that is targeted and defined by the user, while maintaining its aspect ratio. This approach introduces padding around an image so that this could fit into a target dimension.
Face Detection: Face detection is defined as a finding and locating human faces within images or video. It's a pretty useful step in facial applications, such as recognition, security, and photography. Techniques range from easy algorithms to complex deep learning techniques.

Face recognition: Technology that can identify or authenticate a person through comparison and analysis of facial features by algorithms on detected faces to compare with stored facial data. Examples include common applications in security systems, authentication systems, and even attendance systems.

This virtual comprehensive project regarding face detection and recognition works under the aim of using the power of DL techniques to detect and recognize faces appropriately. It involves a thorough system study that covers several important components. First, the dataset was gathered manually by us and included 80-100 images of each person from different angles to ensure that the system can detect faces with minor tilts. A well-defined system architecture will be established, with DL algorithms such as MTCNN (Multi-task Cascaded Convolutional Networks) and FaceNet. Image processing includes resizing the input image to a target size specified by the user while maintaining the aspect ratio. The method adds padding to fit the image into the target dimensions. Face detection and recognition algorithms require a constant input image size for processing. Face detection identifies faces in images, saves bounding box information, optionally crops faces, and annotates images with bounding boxes. Face recognition extracts facial embeddings using the ResNet-Inception model, normalizes these embeddings, and stores them in a FAISS index for efficient similarity search. Finally, it saves recognition results and metadata, including timestamps, to a CSV file.

3.3. Training Procedure

In this project, a complex facial recognition system that works with raw images, which were stored at first in the Face_images directory. It all starts at the tools.py script that resizes the image using padding to keep the aspect ratio of the original image and saves it in the preprocessed_data directory. Next, we have the image_to_embedding.py script, which uses the MTCNN model to find faces and extract them from these images. These faces are then cropped and stored in the post_processed_data directory. Concurrently, we create a labels.txt file storing the bounding box coordinates in YOLO format. Those cropped images are then processed through a FaceNet model, which converts them into embeddings. These embeddings are subsequently catalogued in a FAISS vector database, paired with a `metadata.pkl` file that holds corresponding label information for each embedding.

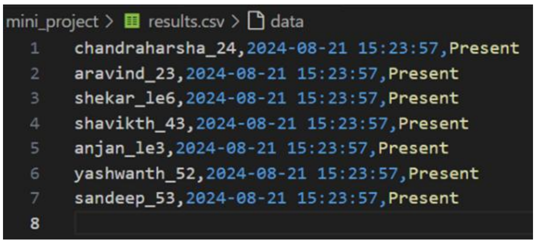
The model was trained on images of individuals, with 100 images per person captured at different angles and under varying quality and brightness conditions. Each image underwent the process above to convert it from an image to an embeddings.

3.4. Inference Process

Inference operations are handled by the `infer_mtcnn.py` script, which not only resizes images but also employs the MTCNN model to detect faces in test images. Cosine similarity metrics are then used to match these faces with the most similar counterparts in the FAISS database. Moreover, the system integrates a Flask-based web framework that streams live webcam feeds through a webpage using the `gen_frames` function. When users engage the attendance button, the system captures a snapshot, stores it temporarily, and processes it via `infer_mtcnn` to verify the identity of the person depicted. Attendance data is efficiently recorded and maintained in a CSV file. This comprehensive framework effectively facilitates real-time facial recognition and attendance tracking.

4. RESULTS

After completing each part mentioned in the previous section, the system is tested on a group of people. The inputs and outputs are listed below.


mini_project > results.csv > data

```
chandraharsha_24,2024-08-21 15:23:57,Present
aravind_23,2024-08-21 15:23:57,Present
shekar_le6,2024-08-21 15:23:57,Present
shavikth_43,2024-08-21 15:23:57,Present
anjan_le3,2024-08-21 15:23:57,Present
yashwanth_52,2024-08-21 15:23:57,Present
sandeep_53,2024-08-21 15:23:57,Present
```

In the first step, a single image with seven students was fed to the face detection and recognition pipeline. The model was able to take each student accurately, and with the help of the photos we had at hand, a CSV file was created, which could record recognized names, and the time duration when they attended effectively. We have added a table showing the accuracy for each student, which is how much confidence there is that the input image corresponds with its data in the training process. The accuracy scores shown above demonstrate the effectiveness of the model in separating each student by their facial features, as a high matching score indicates good performance. It recognized all of the people in that input image with strong results and supplied a confidence level for each identification. This result confirms the model's potential for automated attendance tracking in real-world scenarios where accuracy and speed are essential.

Label	Score
chandraharsha_24	0.81
aravind_23	0.82

shekar_le6	0.85
shavikth_43	0.82
anjan_le3	0.85
yashwanth_52	0.77
sandeep_53	0.80

5. CONCLUSION

We have created an attendance system that is automated, easy to use, and saves the time needed for tracking by reading mindlessly figures. It leverages tools such as MTCNN for face detection, FaceNet for extracting the embeddings from detected faces, and FAISS for fast matching of recent identities to deliver real-time accuracy. Flask provides a web interface that facilitates processing a live webcam stream and user interactions for attendance tracking as the scene happens. Ideal for classrooms, offices, or security systems, this is a practical and reliable solution. It manages tons of data seamlessly, minimizes errors, and avoids proxy attendance. Its modular design also allows it to be easily upgraded with future technologies. It is an advancement on biometric-based contact-free attendance monitoring and unique identification.

6. REFERENCES

[1] Bharath Tej Chinimilli, Anjali T, Akhil Kotturi (2020). Face Recognition-based Attendance System using Haar Cascade and Local Binary Pattern Histogram Algorithm. 4th International Conference on Trends in Electronics and Informatics (ICOEI). https://doi.org/10.1109/ICOEI48184.2020.9143046.

[2] Mayur Surve, Priya Joshi, Sujata Jamadar, Minakshi Vharkate (2020). Automatic Attendance System Using Face Recognition Technique. International Journal of Recent Technology and Engineering (IJRTE), 105–114. https://www.ijrte.org/portfolio-item/A2644059120/

[3] Sudhir Bussa, Shruti Bharuka, Ananya Mani, Sakshi Kaushik. (2020). Smart Attendance System using OPENCV based on Facial Recognition. Proceedings of the International Journal of Engineering Research & Technology. https://pdfs.semanticscholar.org/c38c/43363313ecf6f70ac3194de711a861503e76.pdf

[4] Priyanka Wagh, Jagruti Chaudhari. (2016). Attendance System based on FaceRecognition using Eigen face and PCA Algorithms. Proceedings of the International Conference on Green Computing and Internet of Things (ICGCIoT). https://doi.org/10.1109/ICGCIoT.2015.7380478

[5] Radhika C. Damale, Prof.Bageshree, V. Pathak. (2019). Face Recognition Based Attendance System Using Machine Learning Algorithms. Proceedings of the Second International Conference on Intelligent Computing and Control Systems (ICICCS). https://doi.org/10.1109/ICCONS.2018.8662938

[6] Arjun Raj, Mahammed Shoheb, Chethan K. (2020). Face Recognition Based Smart Attendance System. Proceedings of the International Conference on Intelligent Engineering and Management (ICIEM). https://doi.org/10.1109/ICIEM48762.2020.9160184

[7] Smit Hapani, Nandana Prabhu, Nikhil Parakhiya, Mayur Paghdal. (2019). Automated Attendance System using Image Processing. Proceedings of the Fourth International Conference on Computing Communication Control and Automation (ICCUBEA) https://doi.org/10.1109/ICCUBEA.2018.8697824

[8] SudhaNarang, Kriti Jain, MeghaSaxena, AashnaArora. (2018). Comparison of Face Recognition Algorithms Using OpenCV for an Attendance System. International Journal of Scientific and Research Publications. https://masters.donntu.ru/2018/fknt/makogon/library/article09.pdf

[9] Sikandar Khan, Adeel Akram & Nighat Usman. (2020). Real-Time Automatic Attendance System for Face Recognition Using Face API and OpenCV. Wireless Pers Commun 113, 469–480. https://doi.org/10.1007/s11277-020-07224-2

[10] Smitha, Pavithra S Hegde, Afshin (2020). Face Recognition-Based Attendance using Fisherfaces. Proceedings of the International Journal of Engineering Research & Technology. https://doi.org/10.17577/IJERTV9IS050861

[11] MTCNN Repository https://github.com/ipazc/mtcnn

[12] Facenet Model https://stackoverflow.com/questions/74556149/load-facenet-model

[13] Flask- A Micro Web framework documentation https://flask.palletsprojects.com/en/3.0.x