Real-Time Passenger Tracking System in an Airport Using Machine Learning

Oblisamy L[1], Shraddha Kadam[2], Seema Madurkar[3], Nikita Shendage[4], Rutika Ghaste[5]
[1]*Professor, Department of Aeronautical Engineering, Annasaheb Dange College of Engineering and Technology (ADCET), Ashta, Maharashtra, India*
[2,3,4,5]*Student, Department of Aeronautical Engineering, Annasaheb Dange College of Engineering and Technology (ADCET), Ashta, Maharashtra, India*

***Abstract*—Airports are highly dense areas that necessitate efficient passenger monitoring not only to ensure their security but also the efficient functioning of the airport itself. The current monitoring techniques are mainly based on sensing technologies, including RFID, Wi-Fi, and Bluetooth Low Energy. Although these techniques can provide the capability to perform extensive monitoring, they do not have the means to recognize passengers individually and require costly infrastructure installations, which can offer very little identification information. The presented paper discusses the design and implementation of the real-time passenger monitoring system that incorporates the facial recognition techniques implemented by deep learning techniques into the zone-based monitoring approach. To extract precise features, the DeepFace system with FaceNet embeddings was chosen as the main component of this work, whereas OpenCV library was chosen to detect faces in real-time video streams. Moreover, the number of people present in a certain zone will be determined in real-time as well. According to the experimental findings, the average recognition rate of proposed model equals 92% with a processing time of 100-150 ms per frame. In comparison with the Haar Cascade algorithm with LBPH, the presented model proves to be better in terms of cost-efficiency, recognition rate, and robustness.**

***Index Terms*—Facial Recognition, Passenger Tracking, Deep Learning, Computer Vision, Real-Time Surveillance, Smart Airports, Zone-Based Tracking**

I. INTRODUCTION

Airports represent complex settings that require close attention to monitoring passenger flows, in order to ensure safety and efficiency within the airport facility. In the course of the growing demand for flight services, it becomes impractical to monitor passengers manually. The practice results in various problems, including delays and errors made by people who perform monitoring. Consequently, automated monitoring systems have become popular among researchers. Most of the existing systems for passenger tracking utilize sensors like RFID, Wi-Fi, and Bluetooth Low Energy (BLE). Such systems are effective in monitoring crowds and analysing movements of a large number of people. However, most of these systems are not capable of recognizing individual identity since they do not have personal identification capabilities. Besides, they may require additional hardware infrastructure that might be costly to install and maintain [3].

On the other hand, there have been notable developments in the field of computer vision. Earlier techniques, such as Viola-Jones detector [5] and HOG descriptor [6], although prevalent in the domain of person detection, usually perform poorly under adverse conditions of varying illumination, pose, and occlusions. Deep learning techniques, on the other hand, have led to better performance. Techniques such as FaceNet [1] proposed embedding techniques for face recognition, which have since been modified to make the techniques more robust [2], [4]. Deep learning techniques have also been employed in estimating crowd density [7].

Despite these developments, most of the existing systems either concentrate on crowd tracking irrespective of recognizing individual faces or vice versa. This, therefore, raises the need for a system which integrates the two functionalities in the real-world scenario, which would be a modern airport environment. The proposed system in this paper is made up of deep learning-based face detection coupled with zone-based tracking. The aim is to track passengers by recognizing faces within the zones and estimate the total number of passengers within each zone in real-time. It should be noted that such a system will achieve its objectives without any external equipment.

II. LITERATURE REVIEW

There are extensive research studies regarding passenger monitoring and tracking technologies based on various methods such as sensors, conventional computer vision, and deep learning methods. RFID, Wi-Fi, and Bluetooth Low Energy (BLE) have been extensively applied to monitor the movements of passengers in large airports. These systems can be used to analyse crowd flow and estimate passenger density. For instance, the tracking system implemented in RFID technology uses tags and readers to monitor passenger movements within the airport [3]. Although these systems offer good coverage, they require extra hardware support and tend to trace the device instead of the individual, making them unsuitable for identification applications.

Regarding the area of computer vision, the initial approaches have been focused on the identification and tracking of humans using designed features. In terms of facial recognition, Viola-Jones [5] suggested a fast and effective approach, whereas the HOG approach [6] was used to improve the detection of people by making use of the gradient information. Although they are popular techniques, their performance can be hampered in real-life settings due to changes in illumination, occlusions, and camera angles.

Deep learning technology has enabled the creation of better solutions for this problem, as well as other problems of computer vision. For instance, the FaceNet model [1] provides an effective solution to the face recognition problem by embedding features into a high dimensional space. Additionally, new deep face recognition algorithms perform better and can be employed for real-life tasks [2]. Furthermore, current face detection models such as S3FD can improve the performance of this task [4].

In addition to identification, other approaches such as crowd analysis have recently gained considerable attention among researchers [7]. Techniques based on deep learning have been used to predict the density of crowds as well as analyze the population distribution in dynamic settings. Even though much progress has been made in this area, most systems currently rely on crowd monitoring on one hand and on individual identification on the other hand. Very few works have developed an integrated system that performs both crowd analysis as well as the identification and tracking of individuals.

III. PROPOSED METHODOLOGY

This chapter describes the design and implementation process of the proposed real-time passenger monitoring system. The approach taken here aims at achieving clarity, repeatability, and practicality in the application of the proposed solution in airport contexts.

A. System Architecture

The proposed real-time monitoring system has been designed as an online system that makes use of facial recognition technology and crowd tracking for zones. The system works through processing live video footage in order to detect and recognize faces. After that, it matches these faces against the already existing database in order to recognize the people through facial recognition. Simultaneously, it is also able to count the number of people within different zones.

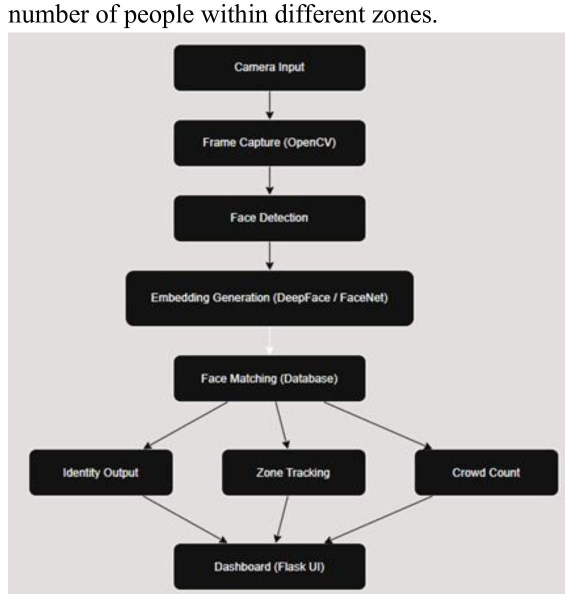

Fig. 1. System Architecture of the Proposed Passenger Monitoring System

B. Data Gathering and Data Set Construction

Passenger data is gathered using a registration process that entails collecting data related to each passenger and facial images. Multiple facial images are captured from each passenger in several views (front, left side view, and right side view). The images are organized in an orderly data set, while the metadata is stored in a CSV file.The images in the database must first be turned into facial embeddings before launching the application using the DeepFace library.

C. Face Detection and Preprocessing

The process of face detection takes place by using the OpenCV framework. The faces in each frame of the video feed are detected by making use of a trained classifier. The detected faces are then cropped and resized into a common format before proceeding with further processing. In order to enhance the performance of the algorithm, any frame with poor-quality data is filtered out.

D. Facial Recognition and Matching

The system uses the DeepFace framework with FaceNet embeddings for facial recognition. Each detected face is converted into a high-dimensional embedding vector. Recognition is performed by comparing this embedding with stored embeddings using Euclidean distance:

$$d = \sqrt{\sum_{\{i=1\}}^{n} (x_i - y_i)^2}$$

where E_detected is the embedding of the detected face and E_stored represents stored embeddings in the database.A match is determined based on a threshold value T. Instead of using a fixed threshold, the system employs an adaptive threshold mechanism that adjusts based on confidence levels and environmental conditions, improving recognition accuracy under varying lighting and pose conditions.

E. Zone-Based Tracking

The environment at the airport is separated into several predefined zones (like Zone A, Zone B). Each camera covers a particular zone. Once a passenger has been detected and identified, the passenger's zone along with the time of detection are logged.

Passenger movements are monitored by means of transitioning between zones:

$$\mathrm{P} = \{(\mathrm{z1}, \mathrm{t1}), (\mathrm{z2}, \mathrm{t2}), \ldots, (\mathrm{zn}, \mathrm{tn})\}$$

where zi is the identified zone and ti is its associated time stamp. This allows for monitoring of passenger mobility within different zones on a continuous basis. In order to prevent duplication of identities, caching is performed, whereby changes in the zone of a person can only be updated after caching his/her recent identity.

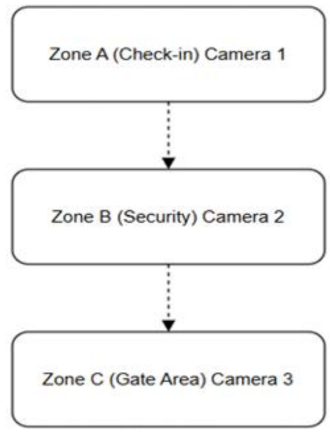

Fig. 2.Zone-Based Passenger Tracking Across Multiple Airport Areas

F. Real-Time Crowd Monitoring

Apart from identity recognition, the crowd surveillance system also makes use of real-time crowd monitoring that involves face detection. A tally of the total number of faces recognized in each image frame is made in order to get an estimate of the number of people within a particular zone. This tally is continuously kept up-to-date and is available in the dashboard.

G. Experimental Setup

The system has been developed utilizing Python language, where the web interface is powered by Flask. For video processing, OpenCV is utilized, while DeepFace is applied to perform face recognition.The experiments are performed under laboratory conditions, using a webcam. The system performs real-time processing of video frames, which takes around 100 to 150 milliseconds per frame on average.

H. Evaluation Metrics

Performance is assessed on the basis of the following measures:

1. Accuracy: This measure gives the accuracy of detection in percentage terms.
2. Time Taken: This measure shows the time taken for processing of each frame.
3. Detection Measure: This measure gives the number of persons detected per frame.

All these measures ensure a complete assessment of performance and speed of processing of the system.

IV. RESULTS

The proposed Real-Time Passenger Tracking System was evaluated in a controlled experimental environment to analyze its performance in terms of identification accuracy, tracking efficiency, and real-time responsiveness. The system was tested using a dataset of registered passengers under varying conditions such as lighting variations and multiple zone transitions.

A. Experimental Setup

The system was implemented using a webcam-based setup simulating multiple airport zones. Passenger data was collected during registration, including facial images captured from multiple angles (front, left, and right profiles). These images were used to generate embeddings for recognition. The evaluation was conducted across multiple test cases to ensure consistency and reliability of system performance.

B. System Output Visualization

The implementation of the proposed system was validated using a real-time dashboard interface. The output of different modules is presented through figures for better understanding.

1. Dashboard Overview

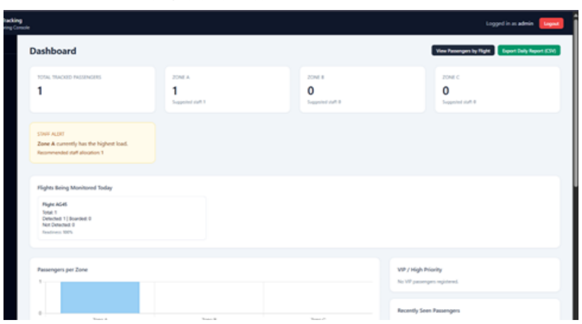

Fig. 3. Dashboard Overview of the Proposed System

In Fig 3, we can see the main dashboard of the system, which displays total tracked passengers, zone-wise distribution, and system-generated alerts. The dashboard also provides staff allocation recommendations based on passenger density.

2. Live Zone Monitoring

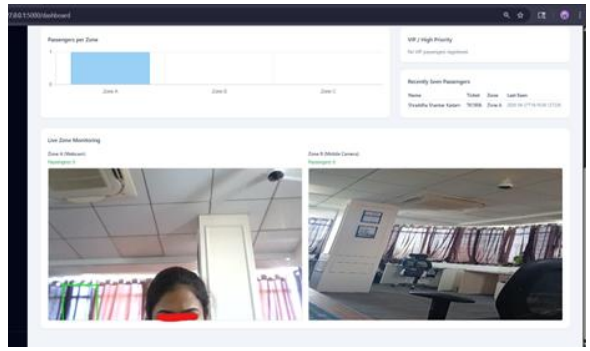

Fig. 4. Real-Time Face Detection and Zone Monitoring

In Fig 4, we can observe the live monitoring system where video feeds from different zones are processed in real time. Face detection is performed using bounding boxes, and identified passengers are tracked dynamically.

3. Passenger Registration Module

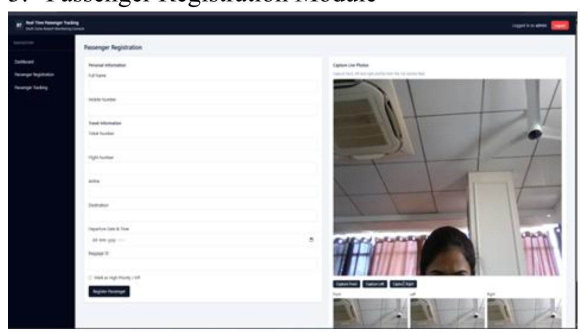

Fig. 5. Passenger Registration and Face Capture Interface

In Fig 5, we can see the passenger registration interface where user details such as name, ticket number, and flight information are entered. The system captures multiple facial images from different angles to generate accurate embeddings.

4. Detected Passenger Output

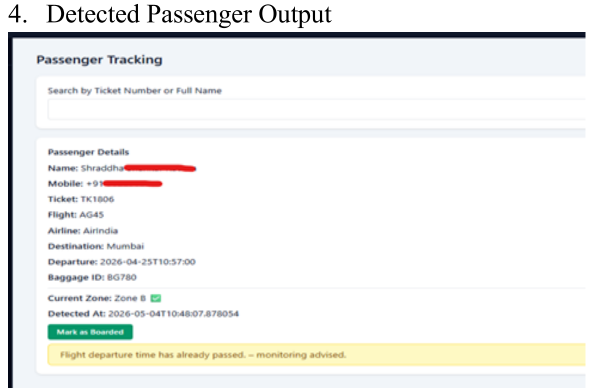

Fig. 6. Passenger Detection and Tracking Output

In Fig 6, we can observe the system successfully identifying a passenger and displaying detailed information including name, ticket ID, flight details, current zone, and detection timestamp.

5. Not Detected / Alert Case

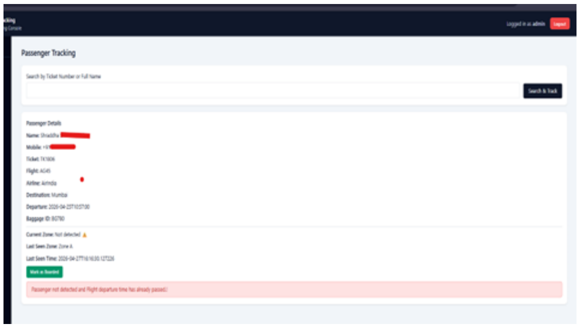

Fig. 7. Alert for Undetected Passenger

In Fig 7, we can see the alert generated when a passenger is not detected within the expected time frame. The system flags the passenger and provides a warning indicating a possible missed flight.

C. Real-Time Performance Analysis

The system processes video frames continuously and performs detection and recognition in near real time. The average processing time per frame was observed to be approximately 100–150 milliseconds, ensuring low latency and smooth system operation.

Table 1: Frame-wise Processing Time

Frame	Processing Time (ms)
F1	110
F2	120
F3	100
F4	140
F5	130

Table 1 represents the processing time per frame, showing that the system maintains low latency suitable for real-time applications.

V. CONCLUSION

The current paper provides a Real-Time Passenger Tracking System intended to improve airport monitoring with the help of using artificial intelligence methods. Using the DeepFace technique for face recognition together with real-time video analysis allows this system to track the movement of passengers from different areas without any significant lags. As shown by the experimental studies, the proposed method works effectively in terms of precision as well as speed. Thus, it can be used in highly dynamic environments, especially, in the airports. Apart from passenger detection, the system is capable of tracking the movement within several zones and providing notifications depending on the passing time of each individual. Nevertheless, despite high efficiency and precision, some limitations associated with the impact of different environmental factors on system performance should be taken into account in the further research. Thus, further development of the presented technology seems necessary to enhance its effectiveness and enable its mass implementation.

ACKNOWLEDGMENT

Hereby, we would like to express our sincere gratitude to our project guides Prof. Oblisamy L and Prof. Sanoj P. Suresh, Department of Aeronautical Engineering, ADCET, Ashta for their constant help, support, and patience during this project. The valuable insights provided by them played a vital part in making us keep focused on the topic and proceed with confidence.
We would also like to extend our sincere thanks to our institution for the provision of all necessary resources. Last but not the least, we thank all our team members for their valuable participation in this project and their dedication and hard work.

REFERENCES

[1] F. Schroff, D. Kalenichenko, and J. Philbin, “FaceNet: A Unified Embedding for Face Recognition and Clustering,” in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2015, pp. 815–823.
[2] O. M. Parkhi, A. Vedaldi, and A. Zisserman, “Deep Face Recognition,” in Proceedings of the British Machine Vision Conference (BMVC), 2015.
[3] J. J. Garau, B. Cameron, and E. Crawley, “System Architecture for Tracking Passengers Inside an Airport Terminal Using RFID,” in Proceedings of the IEEE Aerospace Conference, 2018.
[4] S. Zhang et al., “S3FD: Single Shot Scale-Invariant Face Detector,” in Proceedings of the IEEE International Conference on Computer Vision (ICCV), 2017.

[5] P. Viola and M. Jones, “Rapid Object Detection Using a Boosted Cascade of Simple Features,” in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2001.
[6] N. Dalal and B. Triggs, “Histograms of Oriented Gradients for Human Detection,” in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2005.
[7] C. Zhang, H. Li, X. Wang, and X. Yang, “Cross-Scene Crowd Counting via Deep Convolutional Neural Networks,” in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2015.
[8] Q. Cao et al., “VGGFace2: A Dataset for Recognising Faces Across Pose and Age,” in Proceedings of the IEEE International Conference on Automatic Face and Gesture Recognition (FG), 2018.
[9] M. A. Khan et al., “Smart Airport: A Review of Technologies and Challenges,” IEEE Access, vol. 8, pp. 109561–109580, 2020.