Deep Vision Crowd Monitor: AI for Density Estimation and Overcrowding Detection

Intern Name: Nithya Shree S

Table of Content

S. No.	Title	Page No.
1	Introduction	3
2	Objective of Milestone 4	4
3	System Architecture	5
4	Web-Based Dashboard Development	6
5	Enhanced Alert System using SMTP	7
6	Tools and Technologies Used	8
7	Implementation Details	9
8	Results and Observations	10
9	Conclusion	11

1. Introduction

Crowd management is a critical aspect of public safety in places such as railway stations, malls, temples, and event venues. With increasing population and urbanization, manual monitoring of crowds using CCTV cameras has become inefficient and unreliable. Continuous human supervision is difficult and prone to errors, especially during peak hours.

To address these challenges, automated crowd monitoring systems using artificial intelligence have gained importance. This project focuses on an AI-based crowd density monitoring system that estimates the number of people present in a scene using deep learning techniques.

The system analyzes video input and generates accurate crowd counts along with visual density representations. By integrating the model into a web-based dashboard and alert system, authorities can monitor crowd situations in real time.

This milestone emphasizes deployment, visualization, and alert mechanisms to make the system practically usable.

2. Objective of Milestone 4

The main objective of Milestone 3 is to transform the trained crowd counting model into a functional real-time monitoring application. This phase focuses on integrating the model with a web interface and an automated alert system.

The milestone aims to display live crowd counts and density maps through a user-friendly dashboard. Another key objective is to implement an email-based alert system that notifies authorities when crowd levels exceed safe limits.

This milestone ensures that the project moves beyond theoretical implementation and demonstrates real-world usability. It also validates the effectiveness of the system in handling continuous video input.

Overall, Milestone 3 bridges the gap between AI model development and practical deployment.

3. System Architecture

The system architecture defines how different components interact to perform real-time crowd monitoring. The architecture begins with a video input source, which can be a laptop webcam or a prerecorded CCTV-style video.

The video frames are preprocessed and passed to the deep learning model for analysis. The trained CSRNet model processes each frame to generate a density map and estimate the total crowd count.

The output is then sent to the web-based dashboard, where results are displayed visually. Simultaneously, the alert module checks whether the crowd count exceeds a predefined threshold. If the threshold is crossed, the alert system is triggered.

This modular architecture ensures scalability, flexibility, and efficient real-time performance.

4. Web-Based Dashboard Development

The web-based dashboard serves as the main interface for monitoring crowd density in real time. It is developed using Streamlit, which allows rapid creation of interactive and dynamic web applications.

The dashboard displays live video frames along with estimated crowd counts. A density heatmap is also shown to visually represent crowded regions within the frame.

An alert status indicator is included to show whether the crowd condition is normal or overcrowded. The dashboard updates continuously as new frames are processed.

This visual representation helps authorities quickly understand the crowd situation without manually analyzing video footage. The dashboard improves accessibility and ease of monitoring.

5. Enhanced Alert System using SMTP

The alert system is a crucial component designed to notify authorities during overcrowding situations. Manual observation of dashboards is not always reliable, making automated alerts necessary.

The system uses the Simple Mail Transfer Protocol (SMTP) to send email notifications. When the crowd count exceeds a predefined threshold, an alert email is automatically generated.

The email contains details such as crowd count and alert status. It is sent to registered email addresses stored in the system.

This alert mechanism ensures timely communication and quick response during emergencies. It enhances the safety and effectiveness of the crowd monitoring system.

6. Tools and Technologies Used

Various tools and technologies are used to implement the crowd monitoring system. Python is used as the primary programming language due to its simplicity and strong AI support.

PyTorch is used to implement and run the CSRNet deep learning model. OpenCV is used for video processing and frame extraction.

Streamlit is used to build the web-based dashboard for real-time visualization. SMTP is used to implement the email alert system.

Additional libraries such as NumPy, Pandas, and Matplotlib support data handling and visualization. These technologies together ensure efficient system performance.

7. Implementation Details

The implementation involves integrating multiple modules into a unified system. The trained CSRNet model is loaded and used for predicting crowd density from video frames.

Video input is processed frame-by-frame to ensure real-time analysis. The predicted crowd count and density map are sent to the Streamlit dashboard for display.

A predefined threshold value is set to identify overcrowding situations. When the threshold is crossed, the SMTP alert function is triggered.

The modular implementation allows easy debugging and future enhancements. The system runs efficiently on standard hardware without requiring specialized equipment.

8. Results and Observations

The system successfully displays real-time crowd counts on the web dashboard. Density maps accurately highlight congested areas within the video frame.

The alert system reliably sends email notifications whenever the crowd count exceeds the defined threshold. The dashboard updates smoothly without noticeable delay.

The system performs well with both webcam input and prerecorded video files. These results demonstrate that the system is suitable for real-world deployment.

Overall, the implementation meets the objectives of Milestone 4 and validates the effectiveness of the integrated system.

9. Conclusion

Milestone 4 successfully completes the deployment phase of the AI-based crowd monitoring project. The integration of the deep learning model with a web-based dashboard provides real-time visualization of crowd density.

The SMTP-based alert system enhances safety by automatically notifying authorities during overcrowding situations.

This milestone demonstrates how artificial intelligence can be applied to practical surveillance and crowd management problems.

The project achieves its goal of creating a scalable and efficient crowd monitoring solution. The system can be further extended for real-world applications.