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

Intern Name: Nithya Shree S

TABLE OF CONTENT

ABSTRACT

Crowd monitoring is an essential component of modern surveillance systems used in public places such as malls, railway stations, airports, and events. Manual monitoring using CCTV cameras is inefficient and prone to errors due to human fatigue. This project focuses on integrating a deep learning-based crowd counting model, CSRNet, into a real-time video processing pipeline.

In this milestone, a laptop webcam is used as an alternative to CCTV hardware to simulate real-world surveillance conditions. The system captures live video, processes each frame, estimates crowd count, and displays results using a Gradio-based interface. Alerts are generated when crowd density exceeds a predefined threshold. This milestone demonstrates the practical applicability of CSRNet in real-time crowd monitoring systems.

1.INTRODUCTION

Introduction

With the increasing population and frequent mass gatherings, crowd management has become a major challenge in urban environments. Overcrowding can lead to safety risks such as stampedes, accidents, and security threats. Traditional surveillance systems rely heavily on manual observation through CCTV footage, which is time-consuming and unreliable.

Recent advancements in deep learning have enabled automatic crowd analysis using computer vision techniques. Crowd counting models estimate the number of people in an image or video frame by learning spatial patterns and density distributions. CSRNet is one such powerful convolutional neural network designed for highly congested scenes.

This milestone focuses on implementing a real-time crowd monitoring system using CSRNet. Instead of using physical CCTV cameras, a laptop webcam is used to simulate a live video feed. This approach makes the system cost-effective and accessible while still maintaining real-world relevance.

2.OBJECTIVES

Objectives

The main objectives of Milestone 3 are as follows:

- To integrate the trained CSRNet model into a real-time video processing pipeline
- To simulate a CCTV-style video feed using a laptop webcam
- To capture and process live video frames continuously
- To perform real-time crowd estimation on each frame
- To visualize crowd count results using a Gradio web interface
- To trigger alerts when crowd count exceeds a predefined threshold

This milestone aims to bridge the gap between offline model training and real-time deployment, demonstrating how deep learning models can be used in practical surveillance applications.

3. TOOLS AND TECHNOLOGIES USED

Tools and Technologies

The following tools and technologies were used to implement Milestone 3:

Programming Language:
Python was used for implementing the entire pipeline due to its extensive support for deep learning and computer vision libraries.

Deep Learning Framework:
PyTorch was used to load and run the trained CSRNet model.

Model:
CSRNet (Congested Scene Recognition Network) for crowd density estimation.

Video Processing:
OpenCV was used to capture live video from the laptop webcam and process frames.

User Interface:
Gradio was used to create a real-time dashboard for displaying video feed, crowd count, and alerts.

Development Environment:
Visual Studio Code (VS Code)

Version Control:
Git and GitHub were used to maintain and submit the project.

4. SYSTEM OVERVIEW

System Overview

The real-time crowd monitoring system follows a structured pipeline to process live video input. The laptop webcam acts as a continuous video source similar to a CCTV camera. Each frame captured from the webcam is processed individually and passed through the CSRNet model.

The model generates a density map for every frame, representing crowd distribution. The total crowd count is calculated by summing the density values. The results are then displayed in real time using a Gradio interface.

The system also includes an alert mechanism that activates whenever the crowd count exceeds a predefined threshold. This architecture closely resembles real-world CCTV surveillance systems used in smart cities.

5.METHODOLOGY (LIVE VIDEO CAPTURE & PREPROCESSING)

5. Methodology

5.1 Live Video Capture

The laptop webcam is accessed using OpenCV's VideoCapture functionality. This allows continuous capture of video frames in real time. The webcam feed serves as a replacement for CCTV hardware.

5.2 Frame Preprocessing

Each captured frame undergoes preprocessing steps such as:

- Conversion from BGR to RGB format
- Resizing to match CSRNet input size
- Normalization of pixel values
- Conversion to PyTorch tensors

These steps ensure accurate inference by the CSRNet model.

5.3 CSRNet Inference

The preprocessed frames are passed to the trained CSRNet model. The model outputs a density map where pixel intensity represents crowd density. The sum of the density map values gives the estimated crowd count for the frame.

5.4 Gradio Interface

Gradio is used to create a user-friendly web interface that displays:

- Live webcam feed
- Real-time crowd count
- Warning messages when threshold is exceeded

This interface simplifies monitoring without requiring advanced frontend development.

6. **ALERT MECHANISM**

6. Alert Generation

A crowd threshold value is predefined (for example, 50 people). When the estimated crowd count exceeds this limit, an alert message is displayed on the Gradio interface.

This feature is useful for:

- Crowd control
- Public safety
- Event management

(Optional email alert functionality can be added for real-world deployment.)

7.RESULTS AND OBSERVATIONS

The real-time crowd monitoring system implemented in Milestone 3 was tested using the laptop's built-in webcam as a simulated CCTV video source. The system demonstrated stable and continuous performance while processing live video frames in real time. The webcam feed was successfully captured and streamed without significant latency, allowing smooth frame-by-frame analysis.

One of the key observations is that the crowd count updates dynamically as people move in and out of the camera's field of view. The CSRNet model effectively generated density maps for each frame, and the total crowd count was accurately estimated by summing the density values. This confirms that the trained CSRNet model is capable of handling continuous video input rather than only static images.

The alert mechanism functioned as expected throughout the testing phase. Whenever the estimated crowd count exceeded the predefined threshold value, alert messages were triggered immediately on the Gradio interface. This real-time alert generation is critical for crowd management and safety applications, as it enables timely intervention during overcrowding situations.

The Gradio dashboard provided a clear, simple, and user-friendly visualization of the system's output. The live video feed, real-time crowd count, and alert messages were displayed simultaneously, making the system easy to monitor even for users without technical expertise. The web-based interface also allows easy deployment and access through a browser.

Overall, the system effectively simulates a CCTV-based surveillance environment without requiring actual CCTV hardware. The results confirm that the integration of CSRNet into a real-time pipeline is successful and reliable. The observations from this milestone validate the feasibility of deploying deep learning-based crowd monitoring systems in real-world scenarios such as public spaces, events, and smart city surveillance infrastructures.

8.CONCLUSION AND FUTURE SCOPE

Milestone 3 successfully demonstrates the practical deployment of the CSRNet model in a real-time crowd monitoring pipeline, effectively bridging the gap between theoretical model development and real-world application. The objective of this milestone was to validate the performance of the trained CSRNet model on continuous visual data streams, and this was achieved by simulating a CCTV surveillance environment using a laptop webcam.

By utilizing the laptop's built-in webcam as a live video source, the system was able to capture continuous frames and process them in real time. Each frame was preprocessed and passed through the CSRNet model to generate a density map, from which the crowd count was accurately estimated. This approach closely mimics the functioning of real CCTV-based surveillance systems while eliminating the need for specialized hardware, making the solution cost-effective and easily deployable.

The integration of the Gradio interface further enhanced the usability of the system by providing a simple yet effective dashboard for real-time monitoring. The live display of crowd count, along with instant alert messages when the predefined threshold was exceeded, demonstrates the system's suitability for practical applications such as public safety monitoring, crowd control at events, and smart city surveillance.

Overall, this milestone proves that deep learning-based crowd counting models like CSRNet can be successfully integrated into real-time systems. The implemented solution is scalable, flexible, and adaptable to various real-world scenarios. Milestone 3 lays a strong foundation for future enhancements, including integration with real CCTV cameras, multi-camera support, and advanced alert mechanisms, thereby contributing to the development of intelligent and automated crowd management systems.