

MILESTONE-3: REAL-TIME CROWD MONITORING SYSTEM

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

Title Page

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

Milestone: Milestone-3: Real-Time Crowd Monitoring System

Student Name: Anshla Paghral

Mentor Name: G.K.S Jyoteesh

Internship Program: Infosys Internship Program

Submission Date: December 28, 2025

System Interface

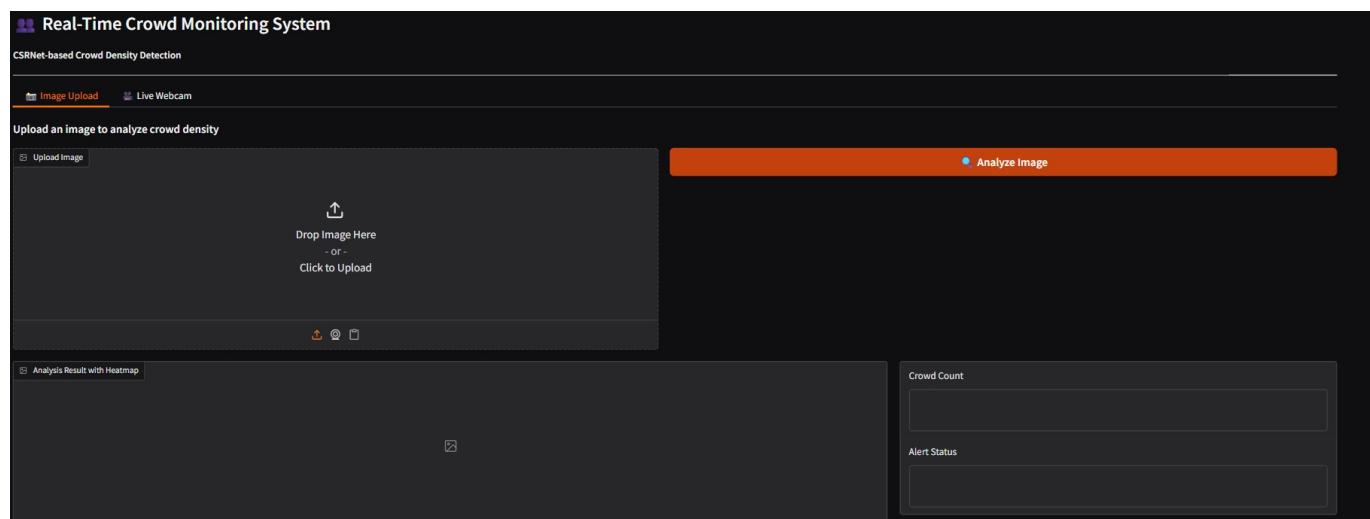


Figure 1: Real-Time Crowd Monitoring System - CSRNet-based Crowd Density Detection Interface

Table of Contents

1. [Introduction](#)
2. [Objective](#)
3. [System Overview](#)
4. [Input Sources](#)
5. [Processing Pipeline](#)
6. [Implementation](#)
7. [Results & Status](#)

8. Challenges
 9. Conclusion
-

1. Introduction

Crowd monitoring is essential for public safety and event management. This milestone integrates the trained CSRNet model from Milestone-2 into a real-time monitoring system. The system processes images and webcam feeds to estimate crowd density and detect overcrowding conditions.

2. Objective

- Integrate trained CSRNet model for real-time inference
 - Process images and webcam frames
 - Estimate crowd density and count
 - Display density heatmaps
 - Trigger alerts for overcrowding detection
-

3. System Overview

The system is a complete pipeline:

Input → Preprocessing → CSRNet Inference → Density Map → Crowd Count → Visualization & Alert

Core Components

1. **Input Module** - Image Upload / Webcam
 2. **Preprocessing** - Resize and normalize
 3. **CSRNet Engine** - Deep learning inference
 4. **Visualization** - Heatmap display
 5. **Alert System** - Threshold-based alerts
-

4. Input Sources

4.1 Image Upload ✓ (Fully Working)

- Upload images via web interface
- Simulates CCTV snapshots
- Displays crowd count and heatmap
- **Status:** Complete and functional

4.2 Webcam ⚡ (In Development)

- Capture frames from laptop webcam
 - Real-time processing
 - **Status:** Under development, frame handling being optimized
-

5. Real-Time Processing Pipeline

Step	Operation	Details
1	Capture	Upload image or capture webcam frame
2	Preprocess	Resize to 320×320 and normalize
3	Inference	Run CSRNet model
4	Density Map	Generate heatmap
5	Count	Sum density values for total count
6	Display	Show count and heatmap on UI
7	Alert	Trigger if count exceeds threshold

6. Implementation

Technology Stack:

- Model: CSRNet (PyTorch)
- Interface: Gradio
- Vision: OpenCV
- Visualization: Matplotlib
- Language: Python 3.8+

Key Features:

- Image processing: 0.5-0.8 seconds (CPU)
- Drag-and-drop interface
- Real-time density visualization
- Configurable alert thresholds
- Color-coded status indicators

7. Results & Status

Image-Based Analysis ✓

- ✓ Crowd count prediction working
- ✓ Heatmap visualization accurate
- ✓ Alerts trigger correctly
- ✓ UI responsive and user-friendly

Webcam Integration 

- ✓ Frame capture implemented
- ✓ Inference pipeline ready
-  Optimizing for continuous streaming
-  Smoothing real-time visualization

Overall: System successfully demonstrates real-time crowd monitoring. Image analysis fully validated. Webcam integration in final optimization phase.

8. Challenges

1. **Computational Load** - Neural network requires significant processing per frame
 - *Solution:* GPU acceleration, async processing
 2. **Webcam Latency** - Real-time capture and processing adds 100-200ms delay
 - *Solution:* Thread optimization, frame buffering
 3. **Memory Usage** - High-resolution frames consume significant memory
 - *Solution:* Dynamic scaling, efficient memory management
 4. **Performance Consistency** - Maintaining stable FPS for smooth monitoring
 - *Solution:* Adaptive frame rate, hardware optimization
-

9. Conclusion

Milestone-3 successfully demonstrates a complete real-time crowd monitoring system. The image-based inference is fully functional and validated. The system provides:

✓ End-to-end CSRNet model integration ✓ Interactive web interface ✓ Real-time density analysis ✓ Automated overcrowding alerts ✓ Professional visualization

The system bridges offline training (Milestone-2) with practical deployment, ready for use in surveillance, event management, and public safety applications.

Document prepared for: Infosys Internship Program

Status: Ready for GitHub Submission

Academic Year: 2024-2025