

PROJECT ABSTRACT

Queue Length Estimation from CCTV using CNN

Project ID: k-1645 | projectech.in/projects/queue-length-estimation-cctv-cnn/

Canteens, banks and clinics share one problem: queues build silently until customers are already angry. This project turns existing CCTV cameras into queue managers — a density-map CNN counts people inside each service-line ROI, counts convert to wait-time estimates, and a dashboard raises alerts when a queue crosses its threshold.

Suitable for B.E./B.Tech final-year projects in Computer Science, IT and AI & Machine Learning.

Project type: Software

Availability: Built to order

Suitable branches: Computer Science, IT and AI & Machine Learning

Technologies: Python 3, PyTorch, CSNet, OpenCV, ShanghaiTech dataset, NumPy, Matplotlib, SQLite, HTML/JS dashboard

Price: Request a quotation

Prepared by Projectech · projectech.in — for academic project reference.

1. Title

Queue Length Estimation from CCTV using CNN — a built-to-order software project delivered with a working demo application, complete source code, project report, presentation and viva preparation kit.

2. Introduction / Background

Counting a queue is harder than detecting people: perspective shrinks distant figures and crowds occlude each other, so bounding-box detectors undercount exactly when it matters. The density-map approach from crowd-counting research regresses per-pixel density whose integral is the count — robust precisely in dense, occluded queues.

3. Problem Statement

Counting a queue is harder than detecting people: perspective shrinks distant figures and crowds occlude each other, so bounding-box detectors undercount exactly when it matters. The density-map approach from crowd-counting research regresses per-pixel density whose integral is the count — robust precisely in dense, occluded queues.

4. Objectives

- Train a CSNet-style density-map CNN on the ShanghaiTech crowd-counting dataset.
- Fine-tune on site queue footage with ROI annotations.
- Convert counts to wait-time estimates via calibrated per-counter service rates.

- Reach the design targets: count MAE ≤ 1.8 persons, wait-time MAE ≤ 1.5 min.
- Deliver a multi-camera dashboard with ROI configuration and threshold alerts.

5. Proposed Methodology

ShanghaiTech Parts A/B (about 330k annotated heads) train the density-map CNN with Euclidean loss on Gaussian density ground truth. The model is fine-tuned on site footage with queue ROI polygons drawn once per camera. Per frame, the density integral inside each ROI gives the count; a calibrated service rate converts counts to wait times. Evaluation reports count MAE and wait-time MAE on annotated site footage.

6. System Architecture / Working

A Python service (frame capture, ROI masking, density inference, alerting) feeds a single-file HTML/JS dashboard: a live camera grid with overlaid counts and wait times, threshold alert banners, hourly trend charts, day-by-hour heatmaps and a timestamped alert log stored in SQLite.

7. Hardware / Software Requirements

- Hardware: any machine that runs Python 3 (GPU recommended for training; CPU suffices for demo).
- Software: Python 3, PyTorch, OpenCV, NumPy, Matplotlib, SQLite.
- Data: ShanghaiTech crowd dataset (free for research); site footage collected at build time.
- Cameras: existing CCTV or webcam feeds for the live view.

8. Expected Outcomes

- A density-map counter hitting the design MAE targets on site footage.
- A live dashboard that makes queue state visible at a glance.
- Wait-time estimates grounded in calibrated service rates, shown as estimates.
- An alert log and analytics supporting real staffing decisions.

9. Applications

- Academic final-year project in Computer Science, IT or AI & Machine Learning programs.
- Teaching aid for crowd counting, density maps and applied computer vision.
- Pilot deployment for a canteen, bank branch or clinic reception.

10. Future Scope

- Automatic counter-opening recommendations from predicted peaks.
- Dwell-time and abandonment analytics per queue.

- Privacy-preserving on-camera inference (edge deployment).
- Multi-site aggregation for chain operations.