# Person Counting & Tracking System for Colleges, Schools, and Offices

EURON

**Person Counting & Tracking System for Colleges, Schools, and Offices**

This system leverages computer vision, edge AI, and IoT to count and track people in various locations like entry/exit gates, classrooms, office spaces, hallways, and common areas.

---

# 1. System Overview

The person counting and tracking system will:

- Detect and count people in real-time.
- Track individuals anonymously (without storing personal data).
- Identify congestion and optimize space utilization.
- Integrate with dashboards for analytics and reporting.
- Send alerts in case of anomalies or policy violations.

---

# 2. High-Level Architecture

The system consists of the following components:

- **Edge Devices** (CCTV Cameras, IoT Sensors, Edge AI Devices)
- **Data Processing Layer** (Edge AI + Cloud Processing)
- **Storage & Databases** (Time-series DB, NoSQL DB, Relational DB)
- **Analytics & Dashboarding** (BI Tools, Web Dashboard)
- **Notifications & Alerts** (Kafka Event Streaming, Alerting System)

---

# 3. Detailed Architecture Flow

## 3.1 Data Acquisition Layer (Input Sources)

This layer captures real-time data using various sensors and cameras.

### (A) Camera-Based Detection

- **CCTV/IP Cameras**: Installed at entry/exit points, hallways, classrooms, and common areas.
- **Depth Cameras (e.g., Intel RealSense, Kinect)**: Used for improved detection in crowded areas.
- **Thermal Cameras**: Used for detecting occupancy at night or in low-visibility conditions.

**(B) IoT Sensor-Based Detection**

- **LiDAR Sensors**: Used for accurate person tracking in large areas.
- **PIR Sensors**: Detect human presence in specific zones.
- **RFID/NFC Sensors**: Used with ID cards to track registered users.
- **BLE Beacons**: Track people using mobile apps for proximity detection.

**(C) Data Preprocessing at the Edge**

- **Edge AI Devices** (NVIDIA Jetson, Google Coral TPU, Intel Movidius) run lightweight AI models to:
  - Detect & count people using YOLOv8/DeepSORT.
  - Perform background subtraction to improve detection.
  - Send filtered data (bounding boxes, counts) to the cloud, reducing bandwidth.

---

## 3.2 Edge Processing Layer

This layer runs AI/ML models on edge devices for real-time inference.

**(A) Model Processing at Edge**

- **Object Detection Models**: YOLOv8, SSD, Faster R-CNN (for person detection)
- **Tracking Algorithms**: DeepSORT, ByteTrack, SORT (for person re-identification)
- **Pose Estimation (Optional)**: OpenPose for detecting group formations.

**(B) Data Aggregation & Message Broker**

- **Lightweight Processing Frameworks**: TensorFlow Lite, OpenVINO, PyTorch Mobile.
- **Message Broker**: MQTT/Kafka to stream processed events to the cloud.

---

## 3.3 Cloud Processing & Analytics Layer

This layer processes, stores, and analyzes data at scale.

**(A) Data Ingestion**

- **Streaming Pipeline**: Apache Kafka (Confluent Cloud) / AWS Kinesis / Google Pub/Sub.
- **Streaming Processing**: Apache Flink / Spark Structured Streaming for real-time transformations.

**(B) Data Storage**

- **Time-Series Database**: InfluxDB / TimescaleDB (for historical occupancy trends).
- **NoSQL Database**: MongoDB / DynamoDB (for storing detected frames, metadata).
- **Relational Database**: PostgreSQL / MySQL (for structured analytics).

**(C) AI Model for Behavioral Analysis**

- **Crowd Detection Model**: CNN-based deep learning for anomaly detection.
- **Occupancy Prediction Model**: Time-series forecasting (LSTM, Prophet).
- **Event-based Detection**: Detect unusual behaviors like loitering.

### 3.4 Visualization & Reporting Layer

- **BI Tools**: Tableau / Power BI / Kibana for analytics dashboards.
- **Custom Web Dashboard** (React.js + Flask/FastAPI backend).
- **Mobile App**: Notifications and reports for facility managers.

### 3.5 Alerting & Notification System

- **Event-Driven Alerts** (Kafka + WebSockets for live updates).
- **Notification Services**: Twilio (SMS), Firebase (push notifications).
- **Integration with Security Teams**: Automated alerts for intrusions.

## 4. System Workflow

1. **Data Capture**: Cameras & IoT sensors capture live feed.
2. **Edge Processing**: AI models run at the edge to detect and count people.
3. **Message Streaming**: Data is sent via Kafka to the cloud.
4. **Cloud Processing**: AI models analyze and store real-time occupancy data.
5. **Dashboarding**: Live analytics are displayed on dashboards.
6. **Alerts & Notifications**: Event-based alerts are triggered.

## 5. Tech Stack

### Edge Layer

- Hardware: NVIDIA Jetson, Google Coral TPU, Intel Movidius.
- AI Models: YOLOv8, DeepSORT, OpenPose.

- Edge Processing: TensorFlow Lite, OpenVINO.
- Message Broker: MQTT/Kafka.

### Cloud Layer

- Data Streaming: Apache Kafka, Apache Flink, Spark Streaming.
- Storage: MongoDB, InfluxDB, PostgreSQL.
- AI/ML Models: TensorFlow, PyTorch, LSTM (for predictions).

### Visualization & Alerts

- BI Tools: Tableau, Power BI, Kibana.
- Web Dashboard: React.js, Flask/FastAPI.
- Notification Services: Twilio, Firebase.

---

## 6. Advanced Features

- **Face-Blurring for Privacy Compliance**.
- **Mask Detection for Safety Enforcement**.
- **Heatmaps for Space Utilization Insights**.
- **Anomaly Detection for Intrusion Alerts**.