RailRelax : Enhancing train travel comfort

Submitted in partial fulfillment of the requirements of the degree

BACHELOR OF ENGINEERING IN **COMPUTER ENGINEERING**

By

Anisha Shankar D12B / 06

Himaja Pannati D12B/40

Wafiya Shaikh D12B/48

Anjali Thakrani D12B/57

Name of the Mentor : **Mrs. Lifna C S**

Vivekanand Education Society's Institute of Technology,

An Autonomous Institute affiliated to University of Mumbai

HAMC, Collector's Colony, Chembur,

Mumbai-400074

University of Mumbai (AY 2024-25)

CERTIFICATE

This is to certify that the Mini Project entitled **" RailRelax "** is a bonafide work of **Anisha Shanker(6),Himaja Pannati (40), Wafiya Shaikh (48), Anjali Thakrani(57)** submitted to the University of Mumbai in partial fulfillment of the requirement for the award of the degree of **"Bachelor of Engineering"** in **"Computer Engineering" .**

(Mrs. Lifna C S)
Mentor

(Prof.__________________)
Head of Department

(Prof.__________________)
Principal

Mini Project Approval

This Mini Project entitled "<u>RailRelax-Enhancing train travel comfort</u>" by **Anisha Shanker(D12B/6), Himaja Pannati(D12B/40), Wafiya Shaikh(D12B/48), Anjali Thakrani (D12B/57)** is approved for the degree of **Bachelor of Engineering** in **Computer Engineering.**

Examiners

1..
(Internal Examiner Name & Sign)

2..
(External Examiner name & Sign)

Date: 22nd October, 2024

Place: Chembur

Contents

Abstract

In metropolitan cities like Mumbai, overcrowding in local trains is a significant challenge, leading to discomfort, safety concerns, and delays for passengers. This project proposes a smart solution by utilizing cameras to monitor and count the number of people in each train compartment in real-time. Through advanced image processing and machine learning techniques, the system can accurately detect the crowd density in each compartment. The gathered data on the occupancy levels of different compartments will be relayed to passengers before the train arrives at the station, either via mobile apps, station displays, or digital signage. This enables passengers to make informed decisions about where to board, opting for less crowded compartments for a more comfortable and safer journey. By distributing passengers more evenly across compartments, the system not only helps reduce congestion but also facilitates smoother boarding and deboarding processes. This could significantly decrease the chaotic rush during peak hours, leading to a more organized travel experience. Additionally, providing passengers with real-time crowd information can alleviate frustration and anxiety associated with overcrowded trains. From a broader perspective, the project also aims to enhance overall passenger safety. By avoiding overly packed compartments, the likelihood of accidents and injuries caused by overcrowding can be minimized. For railway authorities, this system offers valuable insights into compartment-wise passenger loads, which can help in optimizing train schedules, improving capacity management, and planning additional services during peak times. Overall, the project seeks to make train travel more efficient, comfortable, and safer for the millions of daily commuters in busy cities like Mumbai.

Acknowledgement

The opportunity to work on the Rail Relax project has been an invaluable experience, contributing significantly to both our learning and professional development. We would like to extend our heartfelt gratitude to everyone who supported and guided us throughout the development of this innovative crowd analysis application. The successful completion of this report would not have been possible without the collaborative efforts, dedication, and encouragement of many individuals and organizations.

Firstly, we would like to express our deepest appreciation to our project supervisor, Mrs. Lifna C S, for her invaluable insights and technical expertise, which played a crucial role in shaping the design, development, and documentation of the Rail Relax project. Her unwavering guidance fostered an environment of creativity and innovation, enhancing the quality of our work at every stage. We also extend our thanks to our peers and all the participants involved in surveys and testing, whose feedback was instrumental in refining our project.

This project and report reflect the dedication, hard work, and teamwork that are fundamental to successful academic initiatives. We hope the insights and findings presented here will be beneficial to both our institution and the wider academic community. We would also like to acknowledge our fellow students for their input, brainstorming sessions, discussions, and constant moral support throughout the project journey.

Once again, thank you all for your contributions, guidance, and unwavering commitment to the success of this project and report.

Sincerely,

Anisha Shankar D12B / 06

Himaja Pannati D12B / 40

Wafiya Shaikh D12B / 48

Anjali Thakrani D12B / 57

Chapter -1 : Introduction

This chapter gives a brief introduction to the project highlighting the problem scenario, motivation, key objectives and overall outline of the upcoming chapters in this report

1.1 Introduction

Urban rail systems, particularly in densely populated regions, face significant challenges in managing passenger flow, especially during peak hours. Overcrowded trains result in passenger discomfort, longer boarding times, and suboptimal use of available space. Despite technological advancements in transport, many train systems still rely on manual or outdated methods to monitor passenger occupancy, leading to inefficiencies in seat allocation and overall travel experience.

This project addresses these challenges by developing an intelligent, real-time train occupancy monitoring system using state-of-the-art image processing and deep learning techniques. The system captures video input from multiple strategically placed cameras to detect and count passengers, both seated and standing, inside train compartments. By providing real-time data on the availability of seats and crowd density, the system aims to optimize passenger distribution and improve the overall efficiency of train travel.

The main goal is to ensure that passengers are informed about compartment occupancy before they board the train, thus enabling more efficient movement, reducing overcrowding, and enhancing the comfort of the journey. With cities growing at rapid rates, such smart solutions are vital to ensuring that public transport systems can keep up with increasing demand and improve passenger satisfaction.

1.2 Motivation

One of the most pressing issues in metropolitan train systems is the discomfort caused by overcrowding. Commuters often face uncertainty regarding seat availability, leading to rushed decisions, overcrowded compartments, and suboptimal passenger distribution. This issue is especially critical during peak hours when trains are at their fullest capacity. Standing for extended periods in crowded conditions not only makes the journey uncomfortable but can also lead to fatigue, health issues, and stress. The motivation behind this project stems from the need to alleviate these issues and enhance the commuting experience by leveraging modern technology. By monitoring real-time occupancy and providing passengers with up-to-date information about compartment crowding, the system empowers them to make informed choices, reducing congestion and ensuring a more balanced distribution of passengers across compartments.Additionally, transport operators can use this data to better manage train operations, ensuring that resources are allocated efficiently. This technological intervention not only helps reduce stress for passengers but also optimizes the train system's operational capacity, contributing to an overall improvement in public transportation services.

1.3 Problem Statement & Objectives

Current train systems suffer from overcrowded compartments, particularly during peak travel hours, causing discomfort for passengers and inefficient use of available space. Passengers often do not have access to real-time information about seat availability, leading to confusion and overcrowding in certain compartments while others remain underutilized. Train operators also lack precise data to optimize passenger distribution and train capacity utilization.

There is a need for a system that can accurately monitor the number of seated and standing passengers in real-time and report this data to passengers and operators alike. Such a system would help in reducing overcrowding, optimizing seat utilization, and improving the overall travel experience.

The key objectives of the project are :

1. Develop a real-time occupancy monitoring system: The system will capture and process video footage from multiple cameras installed in train compartments to detect and count seated and standing passengers.
2. Enhance passenger experience: By providing real-time data on seat availability and compartment occupancy, passengers can make informed decisions on where to board, reducing overcrowding in certain areas.
3. Improve operational efficiency: Train operators will receive real-time data on passenger distribution, allowing them to make informed decisions on managing train services and adjusting operations based on passenger flow.
4. Utilize deep learning and image processing techniques: The system will leverage modern deep learning frameworks (such as YOLO or Faster R-CNN) for object detection, ensuring high accuracy in passenger counting and tracking.
5. Optimize train capacity: Through the monitoring system, the project aims to achieve more balanced passenger distribution, reducing congestion and improving seat occupancy.

1.4 Organization of the Report

This report is structured to provide a clear overview of the Rail Relax system, designed to address overcrowding on Mumbai local trains by offering real-time compartment occupancy data. Chapter 2, surveys existing systems and their limitations in managing train crowding, identifying the gap that Rail Relax fills with its real-time automated solution. Chapter 3, Details the architecture and design of Rail Relax, including the use of sensors and algorithms to count passengers in each compartment, and how the data is relayed to passengers. Hardware and software specifications, along with validation results, are included. Chapter 4, summarizes the project outcomes and outlines future enhancements, including potential scalability and integration with rail networks. Chapter 5, lists any published papers or articles related to the project.

Chapter - 2 : Literature Survey

This chapter gives a brief introduction to the Survey performed on the existing system, limitations of the existing system and the contributions of individual members in the project.

2.1 Survey of Existing System

In recent years, public transportation authorities across the world have sought to improve operational efficiency and passenger experience by implementing various occupancy monitoring systems. The primary goal of these systems is to provide real-time information about passenger load within trains, buses, or other modes of public transport, especially during peak hours. The following is a brief survey of existing systems currently in use:

1. Passenger Counting Using Infrared Sensors

Several public transport systems use infrared (IR) sensors installed above train doors to detect passengers boarding and alighting from the train. These sensors count the number of people entering and leaving the compartment, providing an estimate of the compartment's occupancy. While this method is cost-effective and relatively easy to implement, it has limitations in accuracy, particularly during periods of heavy boarding when passengers crowd the entry points.

2. Pressure-Sensitive Floors

Another system that has been explored involves the use of pressure-sensitive flooring. These floors detect the weight distribution inside the train compartment, which can then be used to estimate the number of standing and seated passengers. While pressure-sensitive systems provide continuous data, they are sensitive to factors such as passengers' varying weights, luggage, and uneven distribution, which can reduce accuracy. Additionally, the installation and maintenance costs of such systems can be prohibitively high.

3. Smart Card and Ticketing Systems

Many cities use smart card or ticketing systems (such as the Oyster card in London or MetroCard in New York City) to track the number of passengers entering and exiting the train system. These systems provide an indirect method of estimating train occupancy by comparing the number of swipes at various stations. However, they are unable to provide real-time compartment-level data and do not account for passengers moving between compartments.

4. Camera-Based Systems

More advanced camera-based systems, such as those used in cities like Tokyo and Singapore, utilize Closed Circuit Television (CCTV) networks to monitor passenger flow. These systems use video feeds to estimate crowd density and provide real-time updates to passengers and transport operators. Although effective, these systems often require extensive manual monitoring, and traditional computer vision techniques used in these systems may struggle with accuracy in crowded scenarios.

5.. Mobile Applications and Crowdsourcing

Several transportation authorities have begun experimenting with mobile applications that allow passengers to report the occupancy levels of their compartments in real-time. While this method provides crowd-sourced data, its reliability is dependent on passenger participation, and it lacks the precision needed for detailed, real-time monitoring.

6. Deep Learning-Based Object Detection Systems

Some recent advancements have explored the use of deep learning models, such as Convolutional Neural Networks (CNNs), for real-time passenger detection and counting. These systems rely on image-processing algorithms that can detect individual passengers in crowded environments. Though this approach is promising, the implementation of such systems at scale remains a challenge due to the computational resources required and the complexity of deploying such models in real-time across multiple train compartments.

2.2 Limitation Existing system or Research gap

The lacuna of the existing system are listed as follows:

1. **Lack of Real-Time Seat Availability Information**: The current system does not provide passengers with real-time data on available seats, leading to congestion as passengers rush to board without knowing where space is available. This could be mitigated by a system offering accurate, up-to-date information on seat availability in each compartment.
2. **Inefficient Passenger Distribution**: With no system to evenly distribute passengers across compartments, certain sections of the train become more crowded than others. This results in inefficiency and an increased likelihood of discomfort or safety concerns during travel.
3. **Inconsistent Boarding Experience**: Boarding can be chaotic, especially during rush hours when trains are heavily crowded. The absence of a structured process or technological aid to facilitate smooth boarding contributes to the inconsistencies passengers face daily.

4. **Safety Concerns**: Overcrowding exacerbates safety risks. Passengers hanging out of doors or standing in cramped conditions pose a threat to themselves and others. A more organized system that controls boarding and seating can significantly improve safety.
5. **Limited Technological Integration**: The current system does not leverage modern technology like real-time data analytics, video processing, or smart sensors, which could greatly enhance the efficiency of passenger management.

2.3 Mini Project Contribution:

Each team member made significant contributions to the development and success of the Rail Relax project:

- As the project leader, **Wafiya** took charge of the overall direction of the project. She worked on the development of the project's model, including the training and testing phases to ensure accurate crowd counting.
- ***Anisha*** contributed by designing the user interface using Flutter and Figma, ensuring the system's usability. She also designed the flowchart to represent the system's functionality and data flow.
- ***Anjali*** contributed the design efforts in Figma and Flutter and authored the research paper, documenting the technical aspects and findings of the project.
- ***Himaja*** played a key role in working on the model, focusing on its development and improvement alongside Wafiya to ensure the system met its technical goals.

Chapter - 3 : Proposed System

This chapter gives a brief introduction to the Proposed System, architectural framework, algorithm and process design, methodology, hardware and software specifications followed by result analysis and discussions.

3.1 Introduction

As urban populations grow and public transportation networks become more congested, providing passengers with real-time information about train occupancy levels has become crucial. Overcrowded compartments not only reduce passenger comfort but also pose safety risks, particularly during peak travel hours. While various systems have been implemented to monitor train occupancy, most either lack real-time precision, are limited in scalability, or fail to provide actionable insights to passengers in advance.

The proposed system addresses these challenges by offering a comprehensive, AI-driven train occupancy monitoring solution that uses real-time video feeds from compartment-installed cameras, processed through deep learning models. The system accurately detects and counts passengers, providing compartment-level occupancy data with a high degree of precision, even in crowded conditions. This data is not only valuable for transit authorities but can also be shared with passengers in real time via digital displays at stations or mobile applications, allowing them to make informed decisions about which compartment to board.

By leveraging state-of-the-art computer vision techniques and cloud-based infrastructure, this system aims to overcome the limitations of traditional occupancy monitoring solutions, offering a scalable, cost-effective, and privacy-conscious alternative. Additionally, the integration of this system with train management tools can further enhance operational efficiency by allowing for dynamic scheduling and resource allocation based on real-time occupancy patterns. The proposed system is designed to transform the passenger experience by providing transparency, improving safety, and enhancing the overall efficiency of urban rail networks.

3.2 Architectural Framework / Conceptual Design

The architectural framework for the real-time train occupancy monitoring system consists of several key components, each playing a crucial role in achieving accurate passenger counting and efficient data reporting. Below is a point-wise description of the system's design:

1. **Video Capture:**
    - Cameras installed at multiple positions (overhead, angled, and platform-level) continuously capture footage of passengers entering and exiting the train compartments.
    - These cameras are equipped with high resolution and night vision capabilities for accurate detection in varying lighting conditions.
2. **Preprocessing:**
    - Frame Extraction: Each video stream is broken down into individual frames for easier processing.
    - Normalization & Resizing: The frames are resized and normalized to reduce computational complexity without compromising accuracy.
    - Background Subtraction: Techniques like Gaussian Mixture Models (GMM) or deep learning-based methods are used to separate passengers from the background, detecting moving objects.
3. **Object Detection (Passenger Detection):**
    - A deep learning model, such as a Convolutional Neural Network (CNN), is used to detect passengers in each frame. Models like YOLO (You Only Look Once), SSD (Single Shot MultiBox Detector), or Faster R-CNN are commonly used.
    - The system identifies standing and seated passengers with high accuracy, ensuring reliable detection even in crowded scenarios.
4. **Object Tracking:**
    - After detection, passengers are tracked across consecutive frames using algorithms like Simple Online and Realtime Tracking (SORT) or DeepSORT.
    - Tracking ensures that each passenger is followed through their movements, avoiding duplicate counting or missed detections.
5. **Passenger Counting:**
    - Virtual entry and exit lines are drawn at the compartment doors to count the passengers entering or leaving.
    - As people cross these lines, the system increases or decreases the passenger count accordingly.
    - Counting is updated in real-time, ensuring an accurate occupancy report for each train compartment.

6. **Data Aggregation:**
    - The passenger count is aggregated over time (e.g., per minute, per hour) to generate meaningful occupancy statistics.
    - The data is collected from all cameras and compiled to provide a compartment-level view of occupancy.
7. **Real-Time Reporting:**
    - Occupancy data is relayed to train operators and passengers through digital displays or mobile applications.
    - Passengers can view the availability of seats and standing room before boarding the train, allowing better decision-making during peak hours.

Hardware Components:

1. Cameras: High-resolution cameras are placed at various angles (overhead, angled, and platform-level) for comprehensive coverage of the train compartments.
2. Processing Units: GPUs like NVIDIA GeForce RTX handle real-time video processing and deep learning inference.
3. Storage: High-capacity storage units store both raw footage and processed data.

Software Components:

1. Deep Learning Frameworks: TensorFlow or PyTorch are used for training and deploying object detection and tracking models.
2. Computer Vision Libraries: OpenCV is used for preprocessing and background subtraction.
3. Database Management: SQL or NoSQL databases store passenger count data and statistics for later analysis.
4. Web Servers: Nginx or Apache host the real-time reporting dashboard, accessible by operators and passengers.

System Scalability:

- The system is designed to scale across multiple train compartments and routes.
- Cloud infrastructure (AWS, Azure) allows for processing and data storage to be scaled as more trains and cameras are added to the system.

Monitoring and Logging:

- Tools like Prometheus and Grafana provide real-time system monitoring, ensuring that the camera feeds and data processing units function without disruption.
- Logging is handled by the ELK (Elasticsearch, Logstash, Kibana) stack for tracking performance and detecting issues.

This architectural framework ensures a reliable, scalable, and efficient system for real-time monitoring of train occupancy, offering both immediate benefits for passengers and long-term improvements in public transportation management.

3.3 Algorithm and Process Design

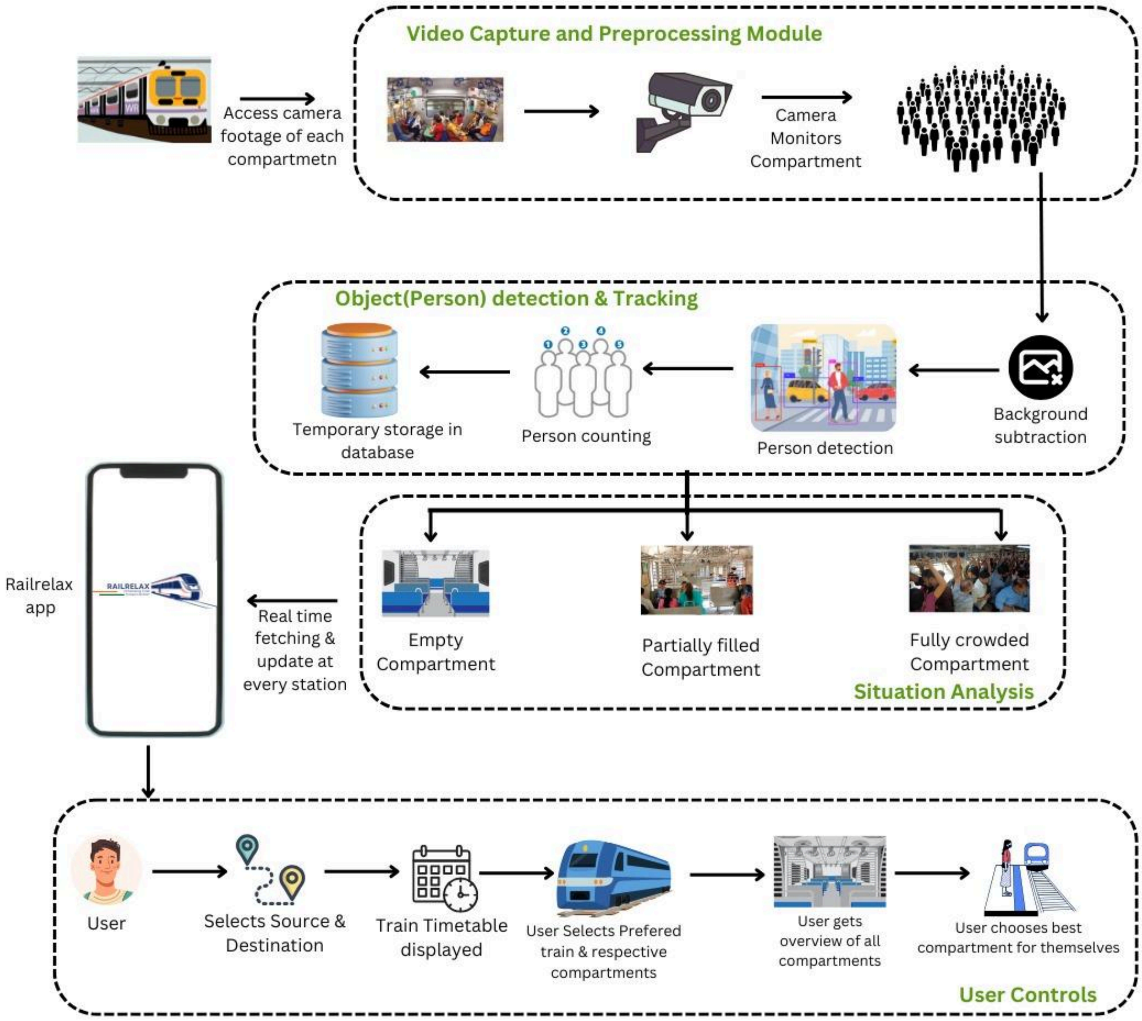

Fig. 1 : Block Diagram

Algorithm:

1. **Capture Live Video:**Cameras inside train compartments capture live video of passengers.
2. **People Detection Using YOLOv3:**The YOLOv3 model processes each video frame, detecting people and drawing bounding boxes around them.
3. **Count People:**The system counts the number of detected people in each frame.
4. **Classify Crowd Density:**Based on the count:
    a. High Density: Many people.

b. Medium Density: Moderate number of people.
c. Low Density: Few or no people.

5. **Display Results in App:**The crowd density is updated in real time and displayed on the app along with other features like train maps, chat, and alerts.

3.4 Methodology Applied

The entire methodology used in the project is explained in a stepwise manner as follows:

1. Problem Definition : The goal of this project is to create a real-time system that detects and classifies the crowd density inside train compartments before the train reaches the station. This will help passengers choose less crowded compartments for a more comfortable journey. The system categorizes each compartment's crowd level as High, Medium, or Low, based on the number of passengers detected within the compartment.

2. Data Preprocessing: To work effectively with video data, the video streams from train compartments were first converted into individual frames. Each frame represents a snapshot of the compartment at a specific moment in time. For each frame, annotation files were created, containing bounding boxes around every person detected. These bounding boxes serve as the ground truth data for training object detection models.

The frames were resized and normalized to ensure uniform input for the model, and the dataset was split into training, validation, and test sets to evaluate model performance accurately.

3. Model Selection and Comparison : A range of object detection models was compared to find the most suitable one for real-time people detection in crowded environments. The models considered include:

- YOLOv11: The latest iteration of YOLO with enhanced performance, particularly on modern hardware, offering faster detection speeds and improved accuracy.
- YOLOv3: Known for its balance between speed and accuracy, making it ideal for real-time detection even in crowded environments.
- YOLOv8: A newer version of YOLO that incorporates architectural improvements for better precision and faster processing compared to earlier versions.
- YOLO-NAS: A cutting-edge model designed to optimize neural architecture

search, yielding high accuracy while maintaining competitive inference speeds, especially suited for real-time object detection in complex scenarios.

The models were evaluated based on:

- **Precision**: The model's ability to avoid false positives, ensuring that each detected object is truly a person.
- **Recall:** The model's ability to detect all people present in the frame, minimizing missed detections.

After extensive testing, YOLOv3 emerged as the best-performing model for this project due to its capability to handle dense crowds while maintaining fast processing times. Its architecture enables high accuracy, even in scenarios where people are closely packed together or partially occluded.

4. YOLOv3 for Real-Time People Detection : YOLOv3 was chosen because of its ability to detect objects with high precision and recall, even in crowded environments. YOLOv3 processes an entire image in one forward pass, making it highly efficient for real-time applications. The model works by dividing the input image into a grid and predicting bounding boxes for each cell, along with confidence scores for object presence. The model was trained using a combination of frames extracted from video data, where each person was annotated with bounding boxes. The training process involved adjusting hyperparameters like the learning rate, batch size, and the number of epochs to maximize performance.

5. Crowd Density Classification : Once people are detected in each frame, the number of individuals is counted to determine the crowd density of the compartment. The crowd level is then classified into one of three categories:

- High Density: A large number of people, indicating a very crowded compartment.
- Medium Density: A moderate number of people, indicating some available space.
- Low Density: Few or no people, indicating ample available space.

This classification is updated in real-time as the train approaches the station, enabling passengers to choose less crowded compartments for boarding. The system ensures fast, accurate updates using the YOLOv3 model's real-time detection capabilities.

6. Real-Time Deployment : The YOLOv3 model is deployed to process live video feeds from train compartments. The model processes each frame in real time, detecting people and classifying the compartment's crowd density. The system runs on edge devices placed inside the train, allowing for real-time decision-making. The inference results are displayed to passengers through mobile apps or station screens, allowing them to view crowd levels and choose the most appropriate compartment.

7. App Development : In addition to crowd estimation, the app we're developing will offer several other features to enhance the commuter experience. These features include:

- **Live Train Maps:** Real-time tracking of trains, allowing users to check the location and status of trains on their route.
- **In-App Chat:** A communication platform for passengers to share updates or ask questions, improving connectivity during their journey.
- **Notifications & Alerts**: Updates on train schedules, delays, and crowd density changes.
- **Other Common Features:** Similar to apps like M-Indicator, the app will provide essential information such as train timings, fare details, and station information.

3.5 Hardware & Software Specifications

The hardware and software specifications of the project is enlisted below:

Hardware Requirements:

- Cameras: High-resolution video capture devices.
- Computing Hardware:CPU: Multi-core processors for video processing and object detection.
- GPU: High-performance GPUs for accelerated image processing and machine learning tasks.
- Storage: Sufficient storage for video data and processed results.

Software Requirements:

- Operating System: Compatible with video processing and machine learning tools (e.g., Windows, Linux).
- Video Processing Libraries:
- OpenCV (for video capture and preprocessing).
- NumPy (for handling numerical operations).

Machine Learning Frameworks:

- TensorFlow or PyTorch (for object detection and tracking).
- Scikit-learn (for additional machine learning tasks if needed).

Programming Languages:

- Python (for its extensive libraries and ease of use in ML and video processing)
- Flutter & Dart for UI creation
- Django for API development

3.4 Experiment and Results for Validation and Verification

3.4.1 Dataset

For training and testing the YOLOv4-tiny model, I collected annotated images through the Roboflow platform. A total of 1057 images were gathered for the training process. These images were sourced from a combination of pre-existing datasets on Roboflow. Since the annotation labels varied across datasets (e.g., some used "person," while others used "people"), I consolidated these labels into a single class labeled "person" to maintain consistency.

Pre-processing : The following pre-processing steps were performed on the dataset:

- **Auto Orient:** This step ensured that all images were uniformly oriented.
- **Resizing:** The images were resized to dimensions of 900 x 450 pixels to standardize their size for training.
- **Class Modification:** The different labels representing individuals (such as "person" and "people") were remapped into a single "person" class to ensure uniformity across the dataset.

The dataset was split into training and testing sets with a 10% test split,

3.4.2 Model Selection

The following table depicts the experimental results with respect to Precision and Recall while using the variations of YOLO models while training.

Model	mAP@0.50	Precision	Recall
YOLOv4-tiny	0.622	0.84	0.51
YOLOv8	0.782	0.838	0.681
YOLOv11	0.776	0.817	0.683
YOLO-NAS	0.754	0.261	0.844

Table - 1 : Comparison of various models tested using various Evaluation Measures

While YOLOv8, YOLOv11, and YOLO-NAS provide impressive metrics, they come with increased computational demands and, in some cases, trade-offs in accuracy. YOLOv4-tiny, with its balance of efficiency, adequate performance, and real-time capabilities, emerged as the best choice for our railway crowd analysis project. Its ability to run efficiently on limited hardware while providing reliable detection makes it an optimal solution for ensuring safety and effective crowd management in transit environments.

YOLOv4-Tiny : The training was conducted using the YOLOv4-tiny model, a streamlined version of the YOLOv4 model optimized for faster inference and lower computational resource usage. YOLOv4-tiny is ideal for edge devices and real-time applications due to its smaller size and quicker processing time compared to the full YOLOv4 model.

Architecture : YOLOv4-tiny consists of the following layers:

- **Convolutional Layers:** The model includes fewer convolutional layers compared to the full YOLOv4, reducing complexity. These layers apply filters to extract features from the input image.
- **Batch Normalization:** This layer normalizes the input to each layer, stabilizing the training process and improving convergence.
- **Leaky ReLU Activation:** This non-linear activation function introduces non-linearity into the model, allowing it to learn complex patterns in the input data.
- **Max Pooling:** Pooling layers are used to down-sample the feature maps, which reduces spatial dimensions and focuses on the most important features.
- **YOLO Head:** The final layer of the model predicts the bounding boxes and the corresponding object classes.

YOLOv4-tiny is approximately 90% smaller than the full YOLOv4 model. This reduction is primarily due to the reduced number of layers and parameters, which allows for faster inference speeds but slightly lower accuracy. However, the trade-off is beneficial when working with edge devices or when real-time performance is crucial.

3.4.3 Training Process

The training of the YOLOv4-tiny model took approximately 2 hours, during which the loss values, including class loss and bounding box loss, were calculated and gradually decreased. The training process followed a typical schedule where the model adjusted its weights to minimize these loss values.

3.4.4 Deployed Model

This section outlines the process of implementing the YOLOv4-tiny model for object detection in an Android application using TensorFlow Lite (TFLite). The model is first converted from Darknet to TensorFlow format, which facilitates integration into a broader ecosystem and allows the use of TensorFlow's extensive functionalities. Following this, the model is converted to TFLite, a lightweight version optimized for mobile and edge devices, which reduces model size and inference time, making it suitable for real-time applications. The resulting TFLite model enables real-time object detection on Android devices, providing fast inference and low resource consumption. This lightweight model is particularly effective for devices with limited capabilities, allowing for immediate object recognition. Such features are essential for applications like crowd analysis in railway compartments, enhancing user experience through efficient real-time data processing and interaction.

3.4.5 App Development

Flutter has been chosen for its ability to create natively compiled applications for mobile, web, and desktop from a single codebase. This approach not only accelerates the development process but also facilitates a seamless user experience across multiple platforms. The app's user-friendly interface and efficient performance make it well-suited for real-time crowd analysis.

The Flutter app will implement the logic necessary to categorize the crowd situation as low, medium, or high based on the data received from the TensorFlow Lite model. This categorization is determined by predefined thresholds that define what constitutes low, medium, and high crowd density.

For instance, a low density may be represented by a count of zero to five individuals, medium density by six to fifteen individuals, and high density by counts exceeding fifteen. By continuously monitoring the crowd data and updating in real time, the app provides users with timely insights into the occupancy levels of each compartment, which is crucial for passenger safety and comfort.

3.5 Result Analysis and Discussion

The dataset preparation involved collecting 1057 annotated images from various sources on the Roboflow platform and standardizing labels to a single class, "person." Images were resized to 900 x 450 pixels and split into training and testing sets, with 10% allocated for testing, ensuring effective model evaluation as depicted in Fig. 1.

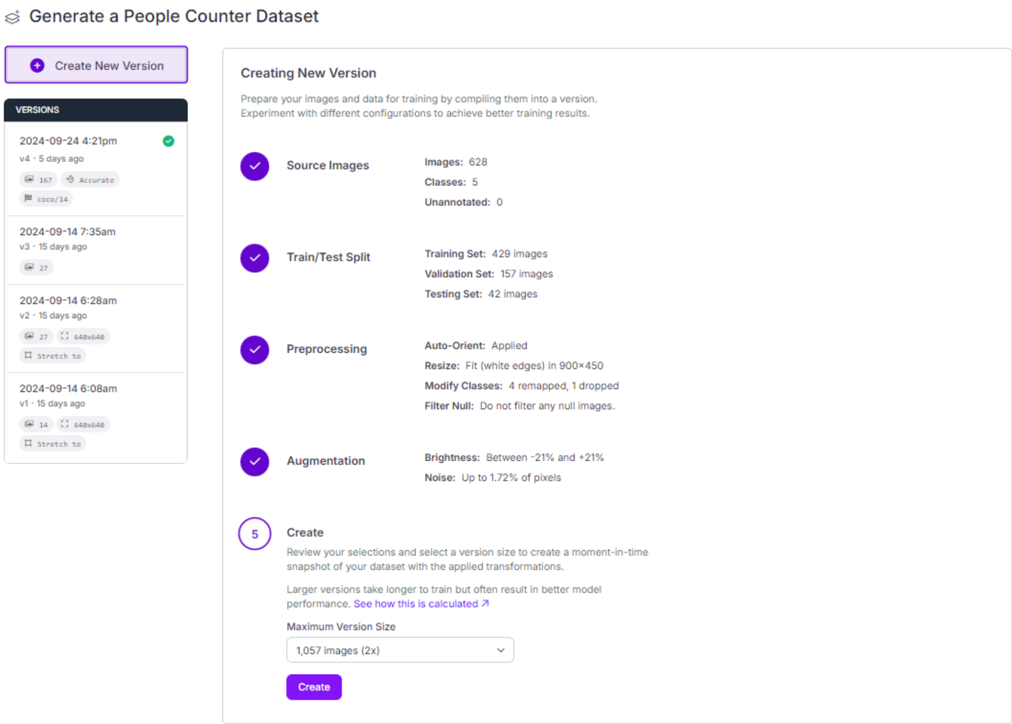

Fig 1. Dataset Preparation

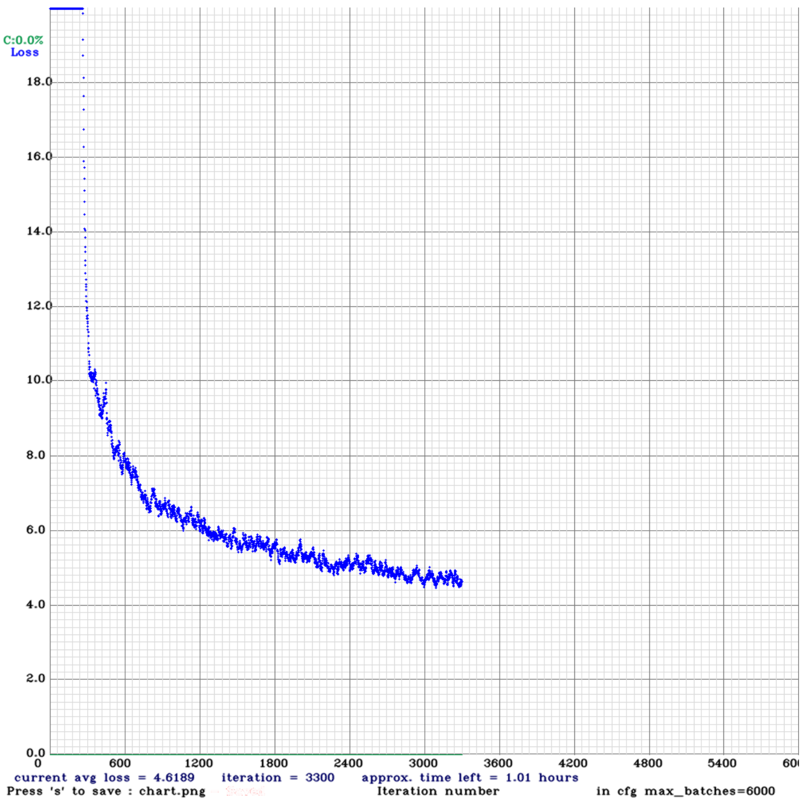

Fig 2. Model Training Results : Loss vs Batch Size Graph

Fig. 2, illustrates the relationship between batch size and loss values during training, showing an initial high loss that decreases steadily as training progresses. Smaller batch sizes resulted in more fluctuations in loss, while larger batches provided smoother curves and faster convergence. This highlights the impact of batch size on training stability and model performance.

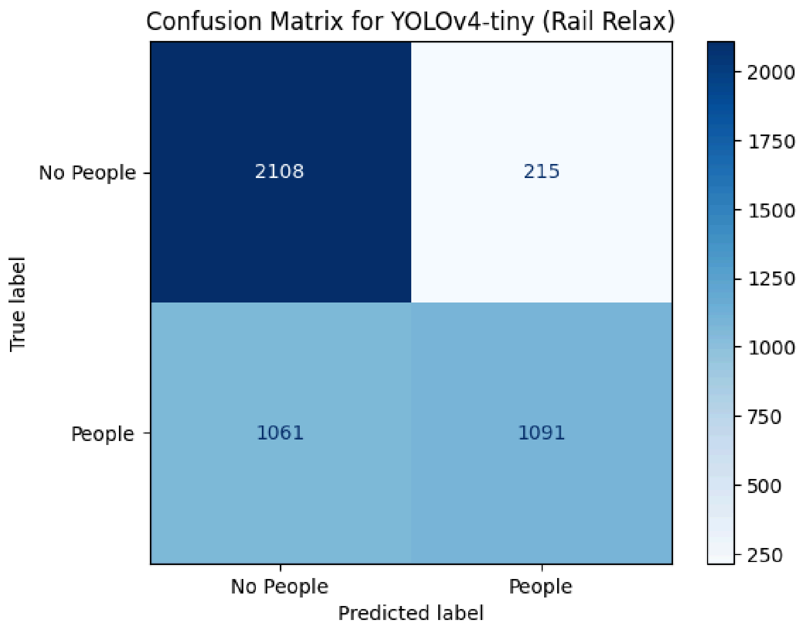

Fig 3. Confusion Matrix of Yolo-v4 tiny Model

The confusion matrix as shown in Fig. 3, indicates that the YOLOv4-tiny model performs well with 1091 true positives in detecting individuals, but it struggles with 1061 false negatives, often missing people in images. Additionally, 215 false positives reveal some misclassifications, while 2108 true negatives show a reasonable ability to identify scenes without people.

Fig. 4, demonstrates, in the test image, the model detected multiple people in a crowded scene, drawing bounding boxes around each individual. The boxes are labeled "person" along with a confidence score, indicating how certain the model is about each detection.

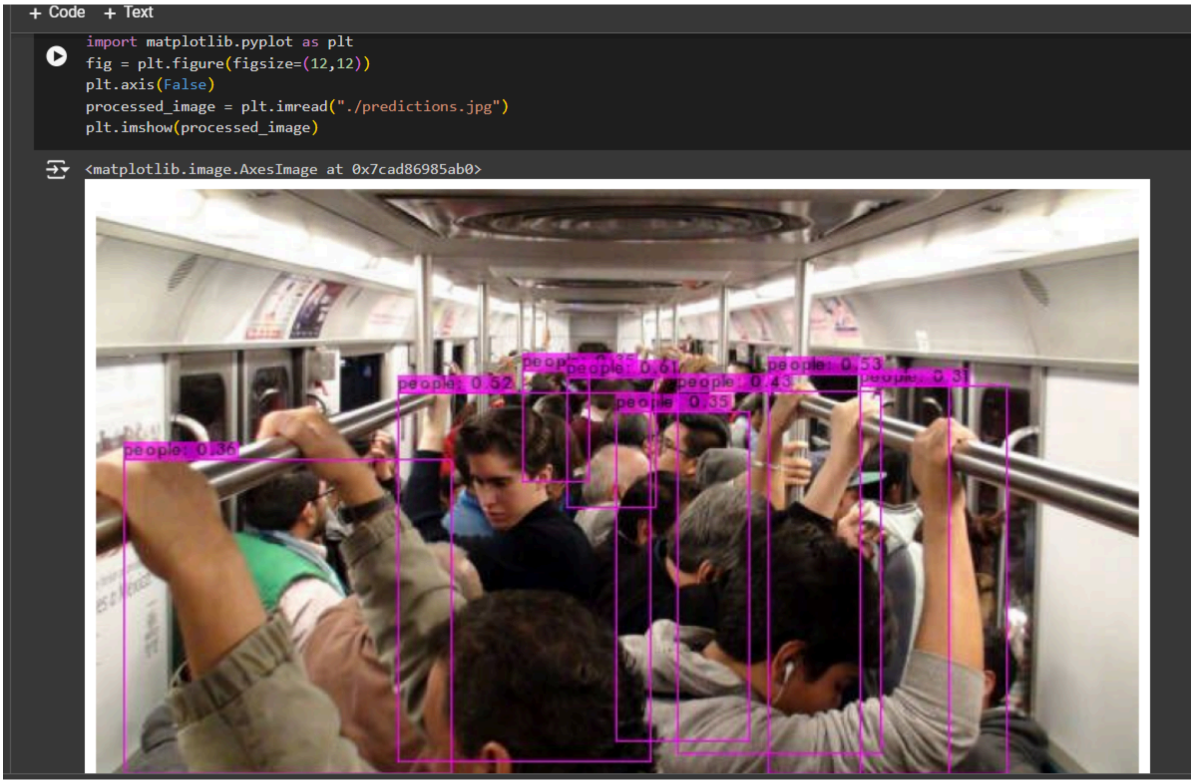

Fig 4. Model Testing on Image

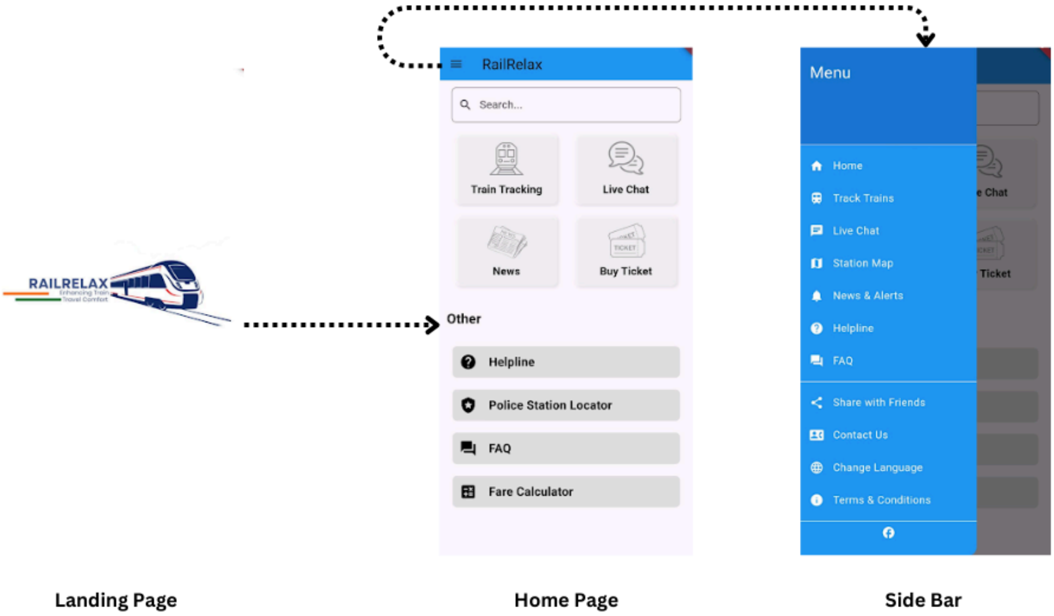

Fig 5. Initial Pages of Rail Relax App

These are the screens of the app , starting with the logo of Rail Relax on the Landing Page, and Apart from crowd compartment analysis ,listing various features like Helpline, FAQ , Live Chat(to be implemented later on) as shown in Fig. 5 and Fig. 6.

Fig 6. Side Bar Page : Features Page of Rail Relax App

This is one of the feature , Helpline Page which shows the Emergency Contacts in case of emergency as shown in Fig.6

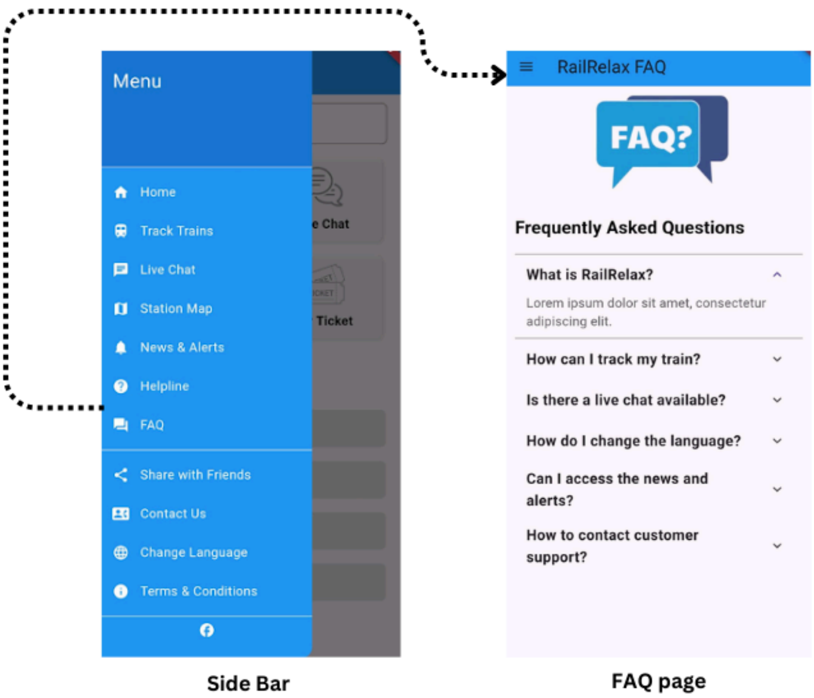

Fig 7. FAQ page : Feature Page of Rail Relax App

Fig 7, shows the FAQ page, which will answer queries regarding the navigation of the app, and feature accessing.

3.6 Conclusion and Future work.

The real-time train occupancy monitoring system presents an innovative approach to improving the efficiency and comfort of public transportation. By leveraging deep learning-based object detection and tracking algorithms, the system accurately counts and monitors passengers in real time. This not only provides passengers with up-to-date information on train occupancy but also enables train operators to optimize crowd management and ensure safety, especially during peak travel times. The system's architecture ensures scalability, real-time performance, and high accuracy, making it a valuable tool in modernizing public transit systems.

Through the integration of video-based passenger detection, the system provides a non-invasive solution to track real-time occupancy without requiring any physical interaction from passengers. The benefits of this system include enhanced passenger experience, reduced congestion, better-informed travelers, and improved operational decision-making for transit authorities. While the proposed system demonstrates robust functionality, several areas can be enhanced and expanded in future iterations:

1. **Enhanced Passenger Detection:**
    - Further refine the detection model to better handle edge cases such as detecting passengers in crowded situations, low-light conditions, or partial occlusions.
    - Implement advanced AI models to differentiate between standing and seated passengers and identify luggage or other large objects that may distort passenger counts.
2. **Multimodal Data Integration:**
    - Integrate additional sensor data, such as thermal imaging or RFID, to complement visual-based passenger detection, improving accuracy in extreme weather conditions or during camera malfunctions.
3. **Real-Time Predictive Analytics:**
    - Develop predictive algorithms that forecast train occupancy at upcoming stations, allowing passengers and operators to anticipate crowd levels and make proactive decisions.
4. **Extended Application Scope:**
    - Expand the system's application to monitor station platforms, bus systems, and other public transport modes, creating a unified transportation occupancy management solution.

    - Explore the use of this technology in intercity trains, metro systems, and long-distance buses, where crowd management is equally important.
5. **Edge Computing Deployment:**
    - Explore edge computing to process video streams locally at train compartments, reducing bandwidth usage and enhancing real-time performance by minimizing latency.
6. **Integration with Smart City Ecosystem:**
    - Integrate the system with a broader smart city framework, where real-time occupancy data can inform city planners, optimize traffic flow, and enhance transportation infrastructure.
7. **Passenger Comfort Analytics:**
    - Analyze data on seating availability, overcrowding trends, and passenger movement patterns to develop insights into overall passenger comfort, which can be used to enhance future train designs and service schedules.

References:

1. Khan, H., Yousaf, H., Murtaza, M., & Velastin, F. (2019). Passenger Detection and Counting for Public Transport System.
2. Curiel, G., Guerrero, K., Gomez, D., & Charris, D. (2023). An Improved Architecture for Automatic People Counting in Public Transport using Deep Learning.
3. Wiboonsiriruk, C., Phaisangittisagul, E., Srisurangkul, C., & Kumazawa, I. (2023). Efficient Passenger Counting in Public Transport Based on Machine Learning.
4. Velastin, S., Fernandez, R., & Espinosa Oviedo, J. E. (2020). Detecting, Tracking and Counting People Getting On/Off a Metropolitan Train Using a Standard Video Camera.
5. Pankaj, A. (2022). A Comprehensive Review on Real-Time Object Detection using Deep Learning Models.
6. Khan, H., Yousaf, H., Murtaza, M., & Velastin, F. (2019). Video analytics using deep learning for crowd analysis: A review. Journal of Visual Communication and Image Representation
7. S. Velastin, R. Fernandez, J. Espinosa Oviedo, and A. Bay, "Detecting, Tracking and Counting People Getting On/Off a Metropolitan Train Using a Standard Video Camera," Sensors, vol. 20, no. 21, 2020.
8. Anuj Kumar Suhane, Aryan Vani, Harsh Parihar, Udit Raghuwanshi, Arjun Nimbark,Lucky Saxena,"HUMAN DETECTION AND CROWD COUNTING USING YOLO”.
9. Maryam Hassan, Farhan Hussain, Sultan Daud Khan, Mohib Ullah, Mudassar Yamin, Habib Ullah (2023),”CROWD COUNTING USING DEEP LEARNING BASED HEAD DETECTION”.
10. QUT SAIVT: Speech, audio, image and video technologies research (2012): Crowd counting database . 1. Queensland University of Technology.dataset. 10.4225/09/5858bfb708148 http://researchdatafinder.qut.edu.au/individual/n920
11. From Semi-Supervised to Transfer Counting of Crowds C. C. Loy, S. Gong, and T. Xiang in Proceedings of IEEE International Conference on Computer Vision, pp. 2256-2263, 2013 (ICCV)
12. Cumulative Attribute Space for Age and Crowd Density Estimation K. Chen, S. Gong, T. Xiang, and C. C. Loy in Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, pp. 2467-2474, 2013 (CVPR, Oral)

13. Crowd Counting and Profiling: Methodology and Evaluation C. C. Loy, K. Chen, S. Gong, T. Xiang in S. Ali, K. Nishino, D. Manocha, and M. Shah (Eds.), Modeling, Simulation and Visual Analysis of Crowds, Springer, vol. 11, pp. 347-382, 2013

14. Feature Mining for Localised Crowd Counting K. Chen, C. C. Loy, S. Gong, and T. Xiang British Machine Vision Conference, 2012 (BMVC)

15. Del Pizzo, Luca, et al. "Counting people by RGB or depth overhead cameras." Pattern Recognition Letters 81 (2016): 41-50.

16. Del Pizzo, Luca, et al. "A versatile and effective method for counting people on either RGB or depth overhead cameras." 2015 IEEE International Conference on Multimedia & Expo Workshops (ICMEW). IEEE, 2015.