Project Description – Edge Computing & Analytics

Smart Crowd Monitoring at Bus Interchange

Design & Justification

Team Number: T5

Tutorial Group: P1

Student Names: Wu Jiakai, Leshane Lee, Jullisha Sasikumar, Lee Ke Yi, Lim Si Hui

Contents

Introduction

Efficient public transportation is vital for the smooth functioning of urban environments. However, commuters often face challenges such as uncertainty surrounding wait times and crowdedness at bus interchanges. To address this issue, our project proposes a **Smart Crowd Monitoring System** utilising **edge computing technologies** at bus interchanges. Leveraging a combination of **computer vision, sound analysis**, and **bluetooth tracking,** we aim to provide real-time information to commuters, enhancing their experience and making public transportation more user-friendly.

Problem Statement

Bus interchanges frequently experience issues of **overcrowding** and **uncertain wait times**, leading to **inconvenience** and **frustration** for commuters. The **lack of accurate and real-time information** about queue lengths, estimated waiting times, and bus arrivals contributes to **inefficient public transportation utilization** and **wasted passenger time**. This project aims to address these challenges by developing a Smart Crowd Monitoring System that leverages on edge computing to provide **precise** and **timely information** to commuters, **improving their overall experience** and **optimizing bus operations**.

Design

Our Smart Crowd Monitoring System comprises several components, designed to operate efficiently within the **resource-constrained environment** of edge computing:

<u>Queue Monitoring</u>
Implementing a **computer vision system** to identify and monitor queues at each bus lane berth using **bounding boxes**. Crowd levels will be assessed by comparing them with the length of railings and the seating capacity of a single-decker bus. **Efficient algorithms** are utilized to minimize computational power requirements.

<u>Real-time Information</u>
Providing real-time information through the use of edge computing, which will pass through a **Telegram bot**. Information will include the number of people in the queue, estimated waiting times, and crowd range indicating the likelihood of securing a seat in the upcoming bus.

<u>Bus Arrival Detection</u>
Utilising **sound analysis** to detect the arrival of buses by analyzing **frequency and amplitude patterns**. Users will be notified when the bus arrives, and estimated waiting times for the next bus will be provided.

<u>Bluetooth Tracking</u>
Deploys **Bluetooth Low Energy (BLE)** devices with low power consumption to track **crowd movement patterns** around the bus berth, providing additional data for crowd level estimation.

User Alerts and Notifications

Enables subscribed users to receive alerts and notifications through the **Telegram bot**. Users can customize alerts based on their preferences, ensuring they stay informed about the status of their bus and the overall crowd level at the interchange.

Integration with Existing Apps

Cross-referencing information with existing bus arrival apps to **validate** and **enhance** the **accuracy** of the provided information, serving as a **reliable backup** and **improving user trust**.

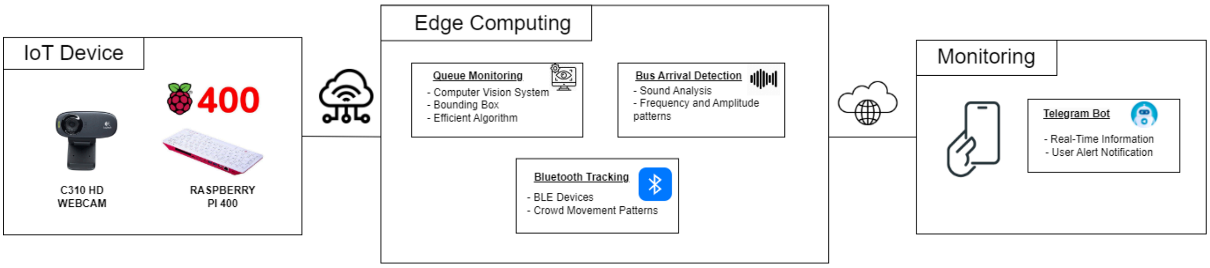

Fig 1. Architecture Diagram

The chosen architecture for edge computing scenarios, featuring IoT devices such as a webcam and Raspberry Pi alongside edge computing functionalities like sound analytics, computer systems, and Bluetooth tracking, exhibits a highly suitable design. By integrating these components, the architecture enables **efficient processing of data** at the edge of the network, reducing the need for constant communication with centralized servers.

Visual and auditory data collected by the webcam and sound analytics are analyzed locally, **minimising latency and bandwidth usage**. Moreover, the inclusion of Bluetooth tracking enhances the system's data acquisition capabilities, contributing to more comprehensive analytics.

Crucially, the architecture incorporates mechanisms for remote monitoring and management through a Telegram bot, **facilitating real-time updates and alerts.** This feature enhances the system's usability and accessibility, ensuring seamless operation even for users without specialized technical knowledge. With its focus on local data processing, storage, and remote monitoring, the architecture effectively harnesses edge computing principles to optimize performance and reliability in diverse IoT scenarios.

Justification

Feasibility and Practicality

The implementation of the proposed system is highly feasible due to advancements in:

- **Computer Vision**: readily available hardware and software for queue detection, with efficient algorithms chosen for real-time processing at the edge.
- **Sound Analysis**: readily available hardware and software for bus arrival detection, with efficient algorithms suitable for edge computing.
- **Bluetooth Tracking**: readily available BLE devices with low power consumption for crowd level monitoring.

Furthermore, edge computing platforms offer the computational power and real-time processing capabilities necessary for analyzing data at the source, making the approach **practical** for **real-world applications** with **limited bandwidth and computational resources**.

Impact and Benefits

a. Enhanced Commuter Experience
Reduce frustration and improve satisfaction by providing real-time information about queue lengths, waiting times, and bus arrival status.

b. Improved Efficiency
Optimize bus schedules and resource allocation based on real-time crowd data, leading to smoother operations, reduced congestion, and potentially shorter waiting times.

c. Accessibility
Multiple channels (Telegram bot and existing apps) cater to diverse user preferences and ensure information accessibility.

d. Cost-Effectiveness
Edge computing enables cost-effective real-time data processing at the source, minimizing latency and reducing infrastructure needs compared to traditional cloud-based solutions.

e. Scalability
The modular design allows for adaptation to different bus interchange layouts and requirements by adjusting camera placement, sensor deployment, or adding new functionalities, making it suitable for deployment in various urban settings.

Risk Assessment and Mitigation

Potential risks and challenges associated with the project will be identified and addressed.

a. Data Privacy Concerns
Collecting and processing data through Bluetooth tracking and visual monitoring may raise privacy concerns among commuters.

To mitigate the data privacy concern in terms of visual monitoring, we are **strategically situating the cameras** at elevated positions to minimise the risk of capturing individual faces directly. We have also **mitigated potential privacy issues** associated with bluetooth tracking by focusing solely on collecting the data related to crowd level **without collecting personally identifiable information** such as MAC address. This will reduce the likelihood of capturing identifiable information, thereby enhancing privacy protection measures within our Smart Crowd Monitoring System.

b. Technical Challenges
i. There may be potential variations in computer vision accuracy due to environmental conditions (lighting, occlusions) which may affect queue detection and the crowd estimation levels.

To mitigate this technical challenge we would conduct training on videos that are taken from various real-world conditions to improve robustness. The implementation of filtering and error correction techniques could also be added to enhance data accuracy.

ii. There may be a possibility of limited processing power and bandwidth at the edge.

To mitigate this technical challenge, we could utilise efficient algorithms and data compression techniques to minimise resource requirements. Through optimisation of system communication protocols, we could potentially reduce data transmission burden

c. Additional mitigation strategies

Alternative mitigation strategies include thorough testing and validation of the system before deployment to ensure reliability and identification of potential issues. Continuous monitoring and feedback mechanisms could be implemented to identify and address any emerging risks or challenges throughout the project lifecycle.