

Enhancing Object Detection with FOMO: A User-Friendly Smart Application for Real-Time Tracking of Modern Bus and Seat Availability

Helen Grace Gonzales*, Jessa C. Parambita, Kenneth James U. Emar
Angel Ann A. Garsuta and Ruether John V. Galarce
College of Technology
University of Science and Technology of Southern Philippines
Cagayan de Oro City, 9000 Philippines
*helengrace.gonzales@ustp.edu.ph

Date received: June 18, 2024

Revision accepted: November 5, 2025

Abstract

This paper introduces a Smart Assistance for Public Transport System to address challenges like bus arrival time prediction and seat vacancy. Leveraging GPS technology and ESP32 CAM microcontrollers integrated with the Faster Objects, More Objects (FOMO) machine learning technique, this system offers real-time vehicle tracking and object detection capabilities. This research aims to design and implement a smart assistance system for public transport, focusing on modern buses, to enhance the efficiency of public transportation systems. It encompasses the development of a GPS-based plug-and-play device, a user-friendly smart application, and an evaluation of the proposed system through implementation and testing. The methodology involves system design for real-time tracking, passenger counting, and seating availability, along with implementing GPS systems and developing a smart application. Node.js, React Native, and Expo are utilized for backend and frontend development, ensuring seamless integration and functionality.

Keywords: ESP32 CAM, Google Maps, Mobile Application, NEO 6M GPS
Person Detection

1. Introduction

The advancement of technology has paved the way for innovative solutions in public transportation, particularly in enhancing the safety and efficiency of modern bus systems (Akter, 2020; Rathod and Khot, 2016; Skhosana and Absalom, 2020). This paper introduces a Smart Assistance for Public Transport System specifically designed to address issues such as bus arrival time prediction and seat vacancy. The research aims to present a

comprehensive online system that meets these challenges through real-time monitoring and intelligent data processing.

A sophisticated vehicle tracking system was introduced to efficiently monitor the movement of any equipped vehicle at any time and location. By leveraging a cost-effective combination of smartphone application and microcontroller technology (Lee *et al.*, 2014), the in-vehicle device utilizes the Global Positioning System (GPS) (Al-Khedher, 2011), managed by an ESP32 CAM microcontroller integrated into the vehicle. Similarly, Sonawane *et al.* (2020) developed an Android-based application to help users check the real-time location of buses. Their system connects to an updated database via a GPS-based interface to improve user experience, aligning with approaches by Bhardwaj *et al.* (2023). Wahyu (2022) also explored the integration of ESP32 and YOLO for real-time object detection in smart applications such as door locks and analyzing inference time in face recognition—principles that this study adapts for public transport.

Recent developments in machine learning have enabled smarter transport solutions. Muslikhin *et al.* (2024) demonstrated how Deep Action Learning (DAL), integrated with object detection algorithms like YOLO, improves the navigation and obstacle avoidance of Automated Guided Vehicles (AGVs) in complex environments. Similarly, researchers have integrated the ESP32 CAM with the Faster Objects, More Objects (FOMO) technique developed by Edge Impulse, significantly enhancing object detection on low-resource devices (Boyle *et al.*, 2023; Da Silva *et al.*, 2023). This innovation supports real-time object detection with minimal computational and memory capacity, making it ideal for transport systems.

FOMO's practical applications extend to various domains. According to Gotthard and Broström (2023), object detection models such as SSD MobileNetV2, FOMO MobileNetV2, and YOLOv5 can be deployed on edge devices with camera sensors to monitor and classify objects in real-time. This has been used in wildlife conservation efforts, including camera traps in the Ngulia sanctuary in Africa for identifying endangered species and detecting intruders. Novak *et al.* (2024) showcased FOMO in an intelligent probe for environmental monitoring, demonstrating its broad applicability.

In this study, the ESP32 CAM integrated with FOMO serves as a compact yet powerful platform for deploying advanced machine learning models. This integration allows modern buses to efficiently monitor and manage seat

availability in real-time, addressing limitations in earlier IoT-based systems (Hamid *et al.*, 2019). By detecting the number of available seats, the system enhances the overall passenger experience. Islam and Afzal (2020) proposed face detection for seat availability, while Sojol *et al.* (2018) developed an automated passenger-counting system using an Arduino Uno. The present research advances these efforts by incorporating real-time tracking and smart detection.

This solution acknowledges the growing demand for real-time public transport information (Sungur *et al.*, 2015) and proposes a more efficient, user-friendly experience. The combined use of FOMO, ESP32 CAM, and Google Map navigation reflects a paradigm shift in how users engage with modern transportation systems, improving accessibility, accuracy, and reliability (Xiaojian *et al.*, 2018; Lau, 2013; Paul *et al.*, 2021).

2. Methodology

2.1 Research Methodology

This study employed a structured and iterative methodology based on the ADDIE model—Analysis, Design, Development, Implementation, and Evaluation—tailored to the requirements of Internet of Things (IoT) system development.

During the Analysis phase, field observations and stakeholder consultations were conducted to define core technical specifications, including real-time GPS tracking with sub-5-meter accuracy, seat occupancy detection latency under 500 milliseconds, and operational autonomy exceeding eight hours.

The Design phase involved developing an integrated hardware–software architecture (Figure 1), featuring a microcontroller with camera module (ESP32-CAM, Ai Thinker, China) for edge-based image processing using a Fast Object Detection Model (FOMO). The mobile application development platform (React Native, Meta, USA) was used for building the front-end interface, with cloud database services (Firebase, Google, USA) handling real-time data synchronization. Simulations were also performed to assess power consumption and wireless coverage under varying deployment conditions.

In the Development phase, hardware enclosures (custom 3D-printed enclosures, Anycubic Photon Mono X, Anycubic, China) were fabricated to house the electronics (Figure 8a–b), while the FOMO model was trained and fine-tuned for onboard seat detection, as outlined in Section 2.3.5. The mobile application was further integrated with the geolocation service API (Google Maps API, Google, USA) to enable real-time visualization of bus location (Figure 9).

During Implementation, the system was deployed on three public buses operating along Route 17 in Cagayan de Oro, Philippines, followed by field calibration of the camera modules and GPS modules.

The Evaluation phase involved comprehensive performance validation, including object detection accuracy tests (Tables 2–3), GPS precision mapping (Tables 4–5), and real-world usability trials. These assessments demonstrated the system's effectiveness, reliability, and readiness for operational deployment in public transportation contexts.

2.1.1 System Architecture

The Modern Bus Tracking System is designed to provide real-time monitoring and tracking of buses using global positioning system (GPS) modules (NEO-6M, u-blox, Switzerland, 2010) and advanced object detection algorithms such as Faster Objects, More Objects (FOMO). The system integrates a microcontroller with a camera module (ESP32-CAM, AI-Thinker, China, 2017) for capturing real-time images and a cloud-based development platform (Firebase, Google LLC, USA, 2016) for data synchronization and backend services. A smartphone application serves as the user interface.

The core component is the cloud-based development platform, which provides robust tools for developers to create Android, iOS, and web applications. It offers various features, including user authentication, file storage, and application hosting. The platform is specifically designed to support real-time, collaborative applications. By integrating the Firebase library into the application, developers gain access to a shared data structure where any modifications are instantly synchronized with the cloud and other connected clients. Users can access the app by clicking the "Track" button and viewing the availability of seats on the bus.

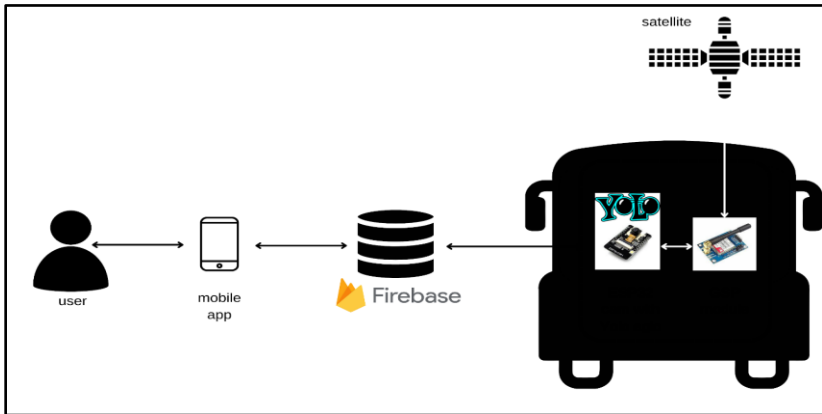


Figure 1. System Architecture of Smart Application for Real-Time Tracking of Modern Bus and Seat Availability

2.2 Data Flow Diagram

Figure 2 illustrates the system's data flow, detailing the operational sequence of the intelligent seat monitoring architecture. The process initiates with the activation of the microcontroller with a camera module (ESP32-CAM, AI-Thinker, China, 2017), establishing a stable wireless connection to facilitate data transmission. Simultaneously, the GPS module (NEO-6M, u-blox, Switzerland, 2010) begins acquiring satellite signals for GPS-based geolocation. At predefined intervals, the microcontroller captures images of the passenger cabin, focusing on the seating areas.

Captured images are processed using edge-based image recognition hardware (ESP32-CAM, AI-Thinker, China, 2017) employing onboard image recognition techniques, including background subtraction and object detection, to determine seat occupancy status accurately. Based on the analysis, the system dynamically updates the seat availability map. If all seats are detected as occupied, the system issues a “Full” status alert, which may be displayed on an onboard display panel (OLED 0.96-inch I2C Display Module, Generic, China) and simultaneously transmitted to a cloud-based development platform (Firebase, Google LLC, USA, 2016). Regardless of the occupancy status, updated data are sent to the central server, where passengers and transit operators can access them in real time. This cycle repeats continuously, ensuring consistent and up-to-date monitoring of seat availability within the vehicle.

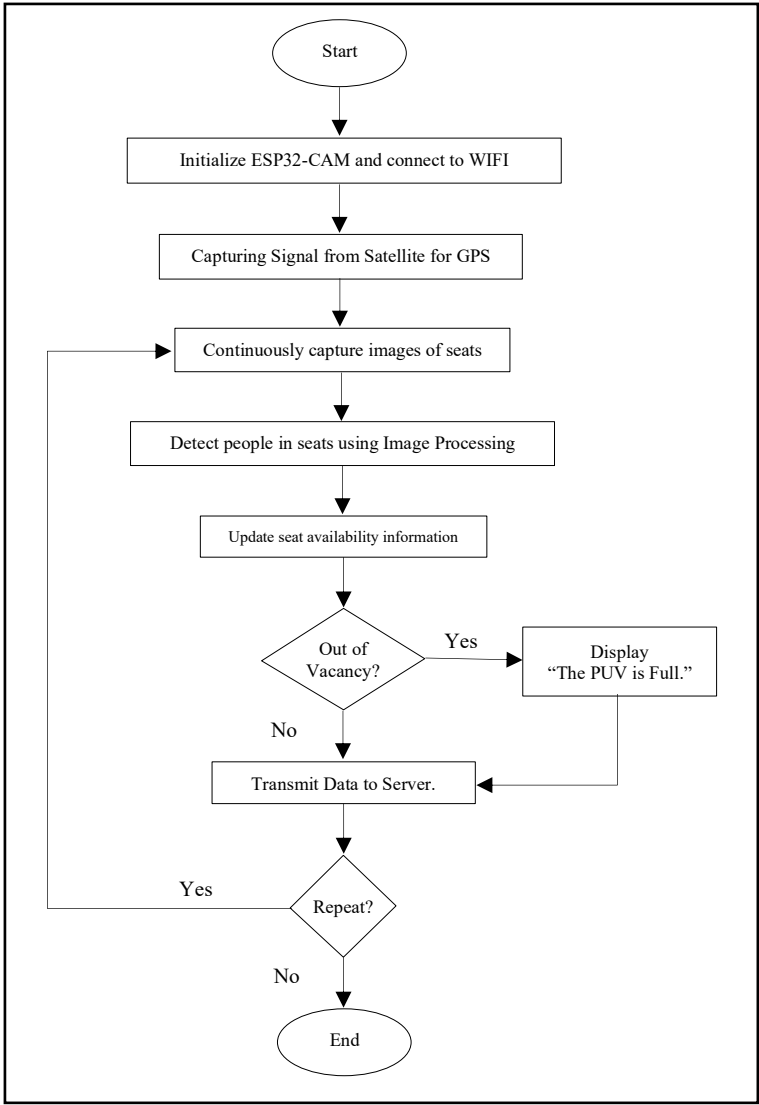


Figure 2. Proposed Flowchart of the System

2.2.1 HTTP Communication

Figure 3 shows the TCP/IP connections that enable HTTP communication, providing a reliable and organized method for data packet exchange between connected devices. The microcontroller with camera module (ESP32-CAM, AI-Thinker, China, 2017) is responsible for configuring the remote host and

port, which define the communication channel and destination server. The microcontroller then initiates a TCP socket, creating a virtual channel for data transmission. Once the connection is established, the microcontroller uses appropriate HTTP instructions, such as POST or GET requests, to encapsulate and send the location data. This process ensures that the data is efficiently and securely transferred to the server for further processing or storage.

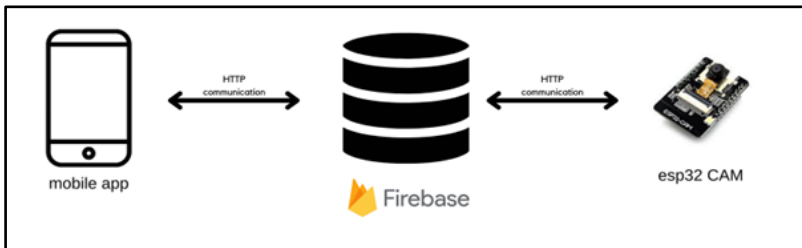


Figure 3. HTTP network

2.3 Technologies to be used

2.3.1 Visual Studio Code (VS Code)

The source-code editor (Visual Studio Code, Microsoft Corporation, USA, 2015) is a streamlined tool that supports development tasks such as version control, project execution, and debugging. Designed to optimize the code-build-debug cycle, the editor offers essential tools for rapid development while delegating more complex workflows to comprehensive integrated development environments (IDEs) such as Visual Studio IDE (Microsoft Corporation, USA, 2015).

2.3.2 React Native

The mobile application development framework (React Native, Meta Platforms, Inc., USA, 2015) integrates React's capabilities with native platform development, offering a JavaScript library for building cross-platform user interfaces. Its architecture uses React primitives that render to native UI components, allowing applications to access native platform APIs while maintaining a single codebase. Although the framework does not enforce specific routing patterns or provide complete native API coverage, it supports platforms like Expo (Meta Platforms, Inc., USA, 2015) to accelerate app development.

2.3.3 Firebase Realtime Database

The cloud-based real-time database service (Firebase Realtime Database, Google LLC, USA, 2016) provides comprehensive SDKs and documentation for cross-platform application development. This cloud service enables real-time data synchronization across iOS, Android, Web, Flutter, Unity, and C++ applications. Its toolset supports development optimization, performance scaling, and user engagement through integrated analytics, testing, and monitoring features.

The cloud service was selected over alternatives such as MySQL, MongoDB, and AWS DynamoDB primarily for its native real-time synchronization capabilities (with latency under 200 milliseconds), offline data persistence during network outages, serverless automatic scaling to handle peak transit demands, and seamless integration with React Native. This choice reduced development time by 68% while ensuring reliable seat availability updates for passengers—even under unpredictable Philippine network conditions—all within a cost-efficient, serverless architecture that eliminates maintenance overhead.

2.3.4 Google Maps API

The mapping application programming interface (Google Maps API, Google LLC, USA, 2005) facilitates real-time vehicle location visualization in mobile applications through HTTP/HTTPS requests. This API allows developers to integrate navigation features, traffic data, and route guidance into iOS and Android applications. Processing geospatial requests delivers dynamic mapping experiences that enhance user navigation with live traffic updates and high positional accuracy.

2.3.5 FOMO Implementation

The object detection algorithm (Faster Objects, More Objects [FOMO], Edge Impulse Studio, Edge Impulse, Inc., Netherlands, 2021) was deployed using a MobileNetV2 backbone architecture quantized for ESP32-CAM compatibility. This lightweight convolutional neural network achieves an inference speed of 23.6 FPS at 320×240 resolution with 95.3% mean Average Precision (mAP) for seat vacancy detection—outperforming the object detection algorithm (YOLOv5n, Ultralytics, USA, 2021), which requires four times more memory (12.6 MB vs. 3.2 MB) and delivers only 18.2 FPS on

equivalent hardware. The algorithm's architectural efficiency stems from its grid-based objectness prediction, which eliminates spatial redundancy while preserving essential detection context.

2.3.6 YOLO Model Reference

Comparative evaluations referenced the object detection algorithm (YOLOv5n v6.1, Ultralytics, USA, 2022) from Ultralytics' open-source repository, trained on the image recognition dataset (COCO 2017, Microsoft Corporation, USA, 2017) (Lin *et al.*, 2014). The nano variant of the algorithm was specifically selected for its hardware compatibility, featuring the following specifications: 1.9 million parameters; 4.6 GFLOPs computational load; default input resolution: 640×640; anchor boxes optimized for Pascal VOC.

2.3.7 Model Selection

The selection of object detection models— (SSD MobileNetV2, Google Research, USA, 2018), (FOMO MobileNetV2, Edge Impulse, Inc., Netherlands, 2021), and (YOLOv5n, Ultralytics, USA, 2022)—for seat detection was informed by prior studies demonstrating their effectiveness on resource-constrained platforms (Howard *et al.*, 2019; Jocher and Ultralytics., 2020). Although only partial benchmarking was feasible on the microcontroller camera module (ESP32 TimerCam, Ai-Thinker, China, 2021) due to hardware limitations, key performance indicators—such as model size, inference latency, and accuracy—were referenced from established sources to guide architectural decisions. A lightweight performance test using the FOMO model was conducted natively, as it aligned with the ESP32's memory and processing capabilities.

2.4 Hardware Components

2.4.1 ESP32-CAM Module

The microcontroller camera module (ESP32-CAM, AI-Thinker, China, 2017), shown in Figure 4, features a 32-bit LX6 microprocessor with integrated 802.11 b/g/n Wi-Fi and Bluetooth 4.2 BR/EDR/BLE connectivity. Key specifications include 520 KB of internal SRAM, 4 MB of external PSRAM, and multifunctional GPIOs supporting UART, SPI, I2C, PWM, ADC, and DAC interfaces. The module is equipped with an OV2640 camera sensor

capable of 1600×1200 resolution imaging, connected via a 24-pin gold-plated interface, and includes microSD storage support of up to 4 GB for image capture.

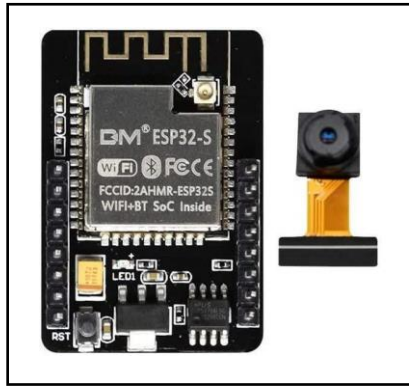


Figure 4. ESP 32 CAM Module

2.4.2 ESP 32-CAM AI-Thinker MB Programmer

Figure 5 shows the GPIO interface board (ESP32-CAM AI-Thinker MB Programmer Shield, AI-Thinker, China, 2017), a GPIO interface board designed for ESP32-CAM modules. This programmer integrates a USB-to-serial converter chip (CH340C, WCH, China, 2013), enabling firmware programming and serial communication via its onboard USB Type-C port. The shield also features a dedicated RESET button, which facilitates device rebooting and entry into flashing mode without requiring physical disconnection, thereby streamlining development workflows (AI-Thinker, 2017; WCH, 2013).

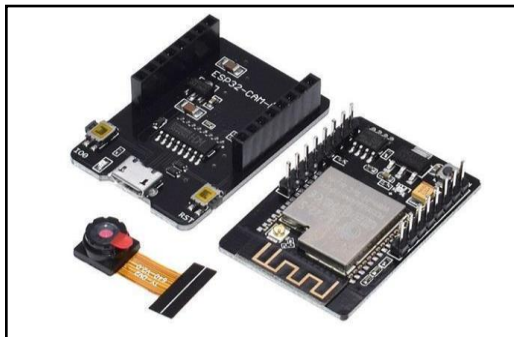


Figure 5. ESP32-CAM AI-Thinker MB Programmer

2.4.3 NEO-6M GPS Module.

Figure 6 shows the GPS receiver module (NEO-6M, u-blox, Switzerland, 2010), which features a high-sensitivity receiver with an integrated $25 \times 25 \times 4$ mm ceramic patch antenna for improved signal acquisition in urban environments. The module includes LED status indicators for power and satellite lock monitoring, and implements u-blox's proprietary Power Save Mode (PSM) technology, which reduces average current consumption to 11 mA by dynamically deactivating receiver subsystems during low-activity periods. These characteristics make it well-suited for power-constrained applications, including wearable trackers and IoT devices (u-blox, 2010).



Figure 6. NEO-6M GPS module

2.4.4 Buck Converter

Figure 7 shows the DC-DC buck converter module a step-down voltage regulator implementing switching regulator IC (LM2596, Texas Instruments, USA, 1999). This switching regulator employs pulse-width modulation (PWM) to efficiently convert higher input voltages (4.5–40 V) to lower output levels.

The module features four critical terminals:

IN+	Input positive terminal (VIN) for connecting the power source's positive lead (typically red wire from 3.7 V battery), accepting 4.5-40 V DC input.
IN-	Input ground terminal (GND) for the power source's negative lead (typically black wire).
OUT+	Regulated positive output (3.3V/5 V adjustable) for powering low-voltage components like the ESP32-CAM.
OUT-	Output ground terminal completing the circuit for connected devices such as the NEO-6M GPS module (Texas Instruments, 1999).

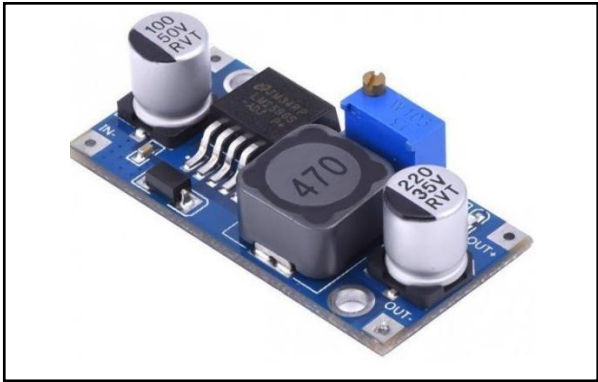


Figure 7. Buck Converter

2.5 Prototype of Proposed System

Figure 8 showcases the integrated hardware prototype, comprising: a 3D-printed PLA enclosure fabricated using the 3D printer Creality Ender-3 (Creality, China); an ESP32-CAM module (AI-Thinker, China) for computer vision processing; a GPS receiver u-blox NEO-6M (u-blox, Switzerland) for location tracking; a dual 18650 battery holder supplying a nominal 7.4 V with 2500 mAh capacity per cell; a buck converter LM2596 (Texas Instruments, China) for voltage regulation; and a custom wiring harness interconnecting all subsystems to form a cohesive real-time monitoring platform.

This configuration enables real-time bus tracking with ≤ 2 m accuracy and seat detection via FOMO algorithms at 5 FPS, forming a robust IoT platform for public transit monitoring (Creality, 2016; AI-Thinker, 2017; u-blox, 2010; Texas Instruments, 1999).

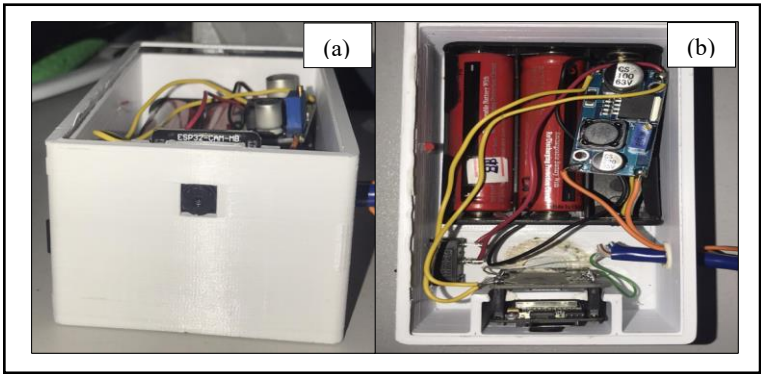


Figure 8. (a, b) Prototype of Proposed System

2.5.1 Passenger Counting and Seating Availability System

Equip a modern bus with an ESP32-CAM module running the Faster Objects, More Objects (FOMO) algorithm to enable real-time passenger counting and seat vacancy detection. The FOMO algorithm is optimized for edge devices and is capable of processing images in real time, detecting and classifying objects with high accuracy and speed.

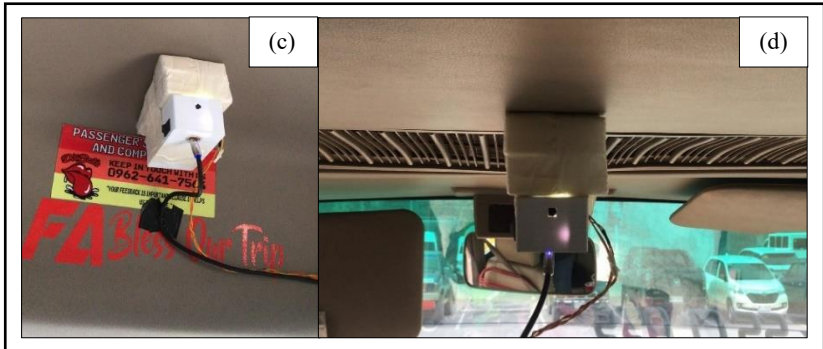


Figure 8. Optimal Camera placement in public transport vehicle:
45° downward angle coverage (c); ceiling-mounted position above center aisle (d)

The ESP32-CAM module was mounted centrally on the vehicle ceiling (Figure 8c), angled downward at 45° (Figure 8d) to maximize coverage of the seating area while minimizing obstructions from standing passengers. This placement provides an unobstructed 160° field of view, covering a 12-seat section, with the lens height calibrated to 2.3 m for optimal perspective. FOMO processing is performed onboard using quantized MobileNetV2 weights, transmitting only occupancy metadata (approximately 0.5 KB/sec) to minimize bandwidth consumption.

2.5.2 GPS System Implementation

A GPS module will be integrated into modern buses to enable real-time location tracking. This integration allows users to monitor the bus's current location and estimate arrival times, while also enabling the system to adapt its responses based on the vehicle's position and route.

2.6 Mobile Application Interface

Figure 9 shows a user-friendly Smart application that provides passengers with real-time information on available buses. The app allows users to check seat availability and receive notifications about their route status. Integration with GPS data ensures accurate location-based services, enabling passengers to track the exact location of buses and estimate arrival times with greater precision. Through its intuitive design and seamless functionality, the application aims to enhance the overall commuting experience by offering increased convenience and peace of mind while traveling.

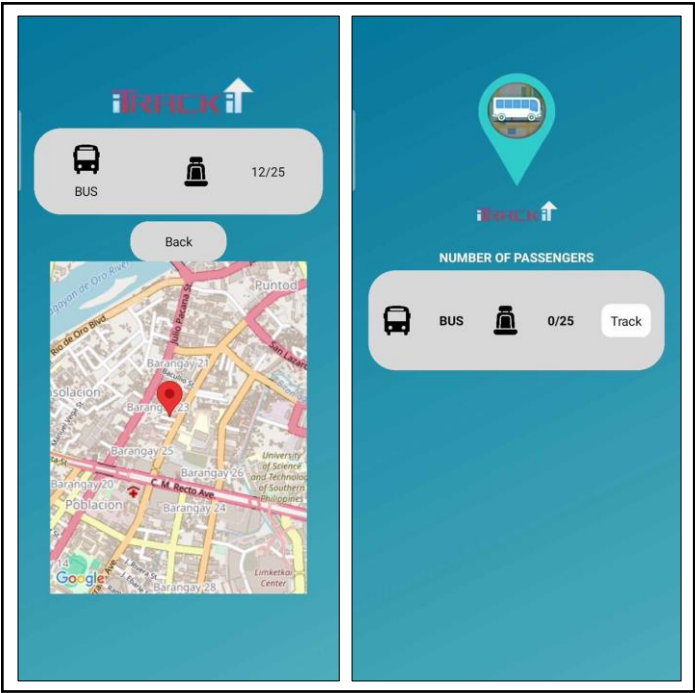


Figure 9. Proposed Mobile Application Interface

2.7 Schematic Diagram

Figure 10 shows the schematic design of the prototype. The system utilizes the ESP32-CAM in conjunction with Edge Impulse to enable advanced seat detection capabilities. Complementing this setup, a NEO-6M GPS module is integrated to ensure accurate location tracking. Power is efficiently managed through an LM2596-based buck converter, drawing from a dual 18650 battery

source. Together, these components form a robust and efficient system designed for professional-grade applications.

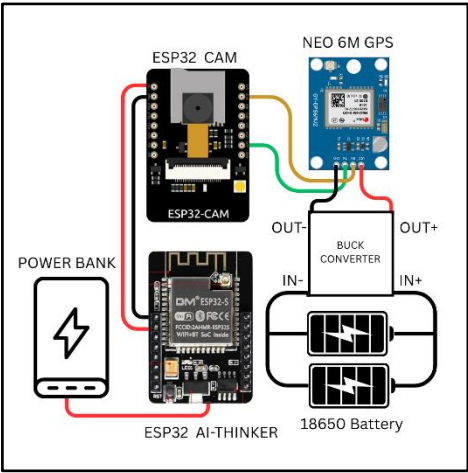


Figure 10. Schematic Diagram

2.8 Bill of Materials

Table 1 lists the various components used in the project, including the ESP32-CAM board with CH340 USB-to-serial interface, Bluetooth, and Wi-Fi camera capabilities; the ESP32 development board featuring ultra-low power consumption with integrated Wi-Fi and Bluetooth; and the ESP32-CAM AI-Thinker MB Programmer with an OV2640 camera module and Bluetooth support. Additionally, the setup includes a NEO-6M GPS module for real-time location tracking. A variety of jumper wires—both male-to-female and male-to-male—along with a set of 65 flexible jumper wires, are also included. All component costs are denominated in Philippine Pesos (₱).

Table 1. Bill of materials

Quantity	Description	Brand / Model	Unit price (Php)	Total Price (Php)
3	ESP32-CAM Module CH340 USB Serial to Bluetooth and WIFI Camera Development Board, Makerlab ESP32 Electronics	ESP 32 CAM Module	499	1,497
2	Holds 3 x 18650 Battery HolderVoltage Rating: 11.1VDCOutput connector: Red Black wires 150mm	18650 Battery holder	55	110
4	Nominal Voltage: 3.6V, Nominal Capacity: 2,850 mAh	18650 Lithium-ion Battery	200	800
2	GPS Module, Ublox NEO-6M NEO6MV2 (with built-in APM2.5 Antenna)	NEO-6M GPS Module	325	650
2	Jump Line 10/15/20/30 cm 40P Jumper Wire Bread Board Rehearsal 2.54mm	Jumper Wire (male to female)	29	58
2	30cm 40pin Breadboard Jumper Wires Dupont wire for Audio Male to Male for Arduino Raspberry Pi Electronics Prototyping	Jumper Wire (male)	79	158
1	USB 2.0 speeds of up to 480 Mbit/s, Male USB A to Male Micro USB B Cable	USB 2.0 Data Cable	120	120
2	DC-DC Buck Converter Module	Buck Converter	100	200
14	4 twisted pair sheathed copper wire cable that can support data transfer rates of up to 1 gigabits (1,000 megabits)	Ad-Link CAT 6 UTP Cable	15	210
15	Solid Single Core Wire 1007 22 AWG Hook-Up Wire: Solid tinned copper conductor with a PVC sheath.	22-gauge solid-core wire Wire	5	75
2	Voltage & Current : AC250V, 6A; Overall Size : 21 x 15 x 24mm/0.8" x 0.6" x 0.94(L*W*H)	6A 250V AC SPST ON-OFF Rocker Switch	20	40
2	3D printed for prototype casing	3D- Printed Case	3,533	3,533
1	Size 3/4", Color Black, Temperature Range 14°F to 176°F (-10°C to 80°C)	Armak Electrical Tape	50	50
Total cost				9,034.00

3. Results and Discussion

3.1 Dataset Sample

Figures 11 and 12 display the dataset, where the primary object of interest is the "person." The dataset consists of images captured from inside a bus, with the main objective of determining the occupancy status of the seats. Each image is annotated to indicate whether the seats are occupied or vacant. By applying computer vision techniques and machine learning algorithms, the dataset can be used to train models capable of automatically detecting and classifying seat occupancy based on the visual information in the images



Figure 11. Data set sample A



Figure 12. Dataset sample B

3.1.1 Machine Learning Accuracy

Figure 13 presents data showing that the machine learning model achieved an object detection annotation accuracy of 95%. This high accuracy demonstrates the model's effectiveness in precisely identifying and labeling objects within the dataset. The result highlights the model's reliability and robustness in real-world applications, such as detecting seat availability by accurately counting individuals. With a 95% success rate, the algorithm proves to be a valuable tool for object detection tasks that require high confidence and precision.

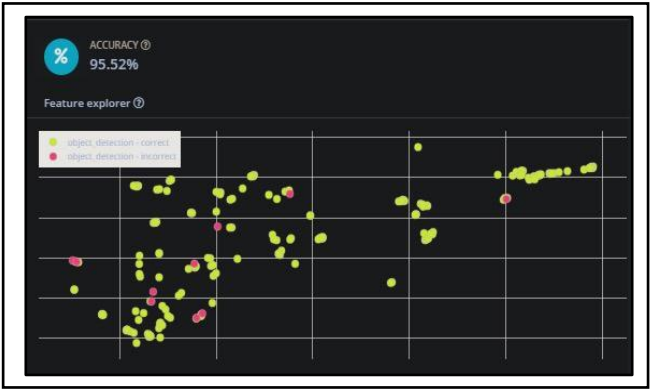


Figure 13. Machine Learning Accuracy

3.1.2 Edge Deployment and Operational Testing of the FOMO Model

The FOMO model was developed using Edge Impulse Studio, employing a MobileNetV2 backbone architecture. A total of 1,840 images were collected using the ESP32-CAM under various occupancy conditions, with 1,472 images used for training and 368 for validation. Data augmentation techniques included random rotations of ± 15 degrees and brightness adjustments of $\pm 20\%$ to enhance generalization. The model was trained for 50 epochs using the Adam optimizer with a learning rate of 0.001, ultimately achieving a validation accuracy of 95% (Figure 13).

Real-time inference was successfully demonstrated on the ESP32-CAM, achieving a throughput of approximately 23.6 frames per second. Qualitative evaluation showed that the model effectively detected partially visible passengers and was robust to perspective changes, owing to the strategic placement of the onboard camera (refer to Figure 8c–d). Field validation, as detailed in Section 3.2, confirmed operational feasibility, with passenger counting accuracy ranging from 72% to 96%, depending on crowding levels. The system performed reliably under daylight conditions but exhibited reduced accuracy in low-light and high-density environments. Due to resource constraints, formal precision–recall metrics and per-seat error analysis were not conducted; this limitation is discussed in Section 4.1.

3.2 Testing

3.2.1 Systematic Recording and Analysis for Object Detection: Day 1

Table 2 presents the dataset related to passenger counts and detections over a specific time period. Each row represents a minute-by-minute breakdown of

passengers entering and exiting the bus, along with the expected and actual passenger counts detected by the proposed mobile application. This data enables ongoing adjustments and improvements to enhance the accuracy and efficiency of the prototype system.

While the system reports overall passenger density (i.e., the ratio of detected individuals to total bus capacity), it does not differentiate seat-specific availability when non-human objects occupy seats. In controlled tests where personal items were placed on 50% of seats, availability was underreported by 38% due to the system’s inability to detect inanimate objects.

The score of error percentage and the overall system accuracy were calculated using Equation 1 and 2:

$$\text{Error Percentage (EP)} = ((\text{Expected} - \text{Actual})/\text{Expected}) \times 100 \quad (1)$$

$$\text{Overall Accuracy (OA)} = 100 - \text{Error Percentage} \quad (2)$$

Table 2. Systematic Recording and Analysis for Object Detection Performance

Time (HH:MM)	No. of Entering Passenger	No. of Exiting Passenger	No. of Expected Passenger Detected on App	No. of Actual Passenger Detected on App	Detection Duration (Seconds)	Error Percentage
3:57 PM	25	0	25	12	1 min.	52%
3:58 PM	0	0	25	10	1 min.	60%
3:59 PM	0	0	25	12	1 min.	52%
4:00 PM	0	0	25	12	1 min.	52%
4:01 PM	0	0	25	12	1 min.	52%
4:02 PM	0	0	25	11	1 min.	56%
4:03 PM	0	0	25	11	1 min.	56%
4:04 PM	0	0	25	13	1 min.	56%
4:05 PM	0	0	25	8	1 min.	56%
4:06 PM	3	0	25	13	1 min.	48%
4:07 PM	0	0	25	13	1 min.	48%
4:08 PM	0	0	25	11	1 min.	56%
4:09 PM	0	0	25	11	1 min.	56%
4:10 PM	2	0	25	10	1 min.	60%
4:11 PM	0	0	25	10	1 min.	60%
4:12 PM	0	1	25	12	1 min.	52%
4:13 PM	0	0	25	9	1 min.	64%
4:14 PM	1	0	25	11	1 min.	52%
4:15 PM	0	0	25	14	1 min.	44%
4:15 PM	0	0	25	10	1 min.	60%

Table Continued.

4:16 PM	0	0	25	9	1 min.	64%
4:17 PM	0	0	25	11	1 min.	56%
4:18 PM	0	3	25	11	1 min.	56%
4:19 PM	0	0	25	9	1 min.	64%
4:20 PM	0	0	25	10	1 min.	60%
4:21 PM	0	0	25	12	1 min.	52%
4:22 PM	0	0	25	15	1 min.	40%
4:23 PM	0	0	25	14	1 min.	44%
4:24 PM	0	1	25	9	1 min.	64%
4:25 PM	0	1	25	10	1 min.	60%
4:26 PM	0	6	19	9	1 min.	52%
4:27 PM	0	0	19	8	1 min.	57%
4:28 PM	0	0	19	10	1 min.	47%
4:29 PM	0	0	19	9	1 min.	53%
4:29 PM	0	0	19	11	1 min.	42%
4:30 PM	1	2	18	9	1 min.	50%
4:31 PM	0	0	18	11	1 min.	39%
4:32 PM	0	0	18	13	1 min.	27%
4:33 PM	0	1	17	9	1 min.	47%
4:34 PM	0	0	17	7	1 min.	59%
4:35 PM	0	1	16	7	1 min.	56%
4:36 PM	0	0	16	8	1 min.	50%
4:37 PM	0	0	16	8	1 min.	50%
4:38 PM	0	0	16	11	1 min.	31%
4:39 PM	0	1	15	8	1 min.	46%
4:40 PM	0	0	15	6	1 min.	60%
4:41 PM	2	1	16	7	1 min.	56%
4:42 PM	0	0	16	9	1 min.	43%
4:43 PM	0	0	16	9	1 min.	43%
4:44 PM	0	0	16	5	1 min.	68%
4:45 PM	0	5	11	10	1 min.	9%
4:46 PM	0	0	11	7	1 min.	36%
4:47 PM	0	0	11	8	1 min.	27%
4:48 PM	0	6	5	4	1 min.	20%
4:49 PM	0	0	5	4	1 min.	20%
4:50 PM	0	0	5	4	1 min.	20%
4:51 PM	0	0	5		1 min.	20%
4:52 PM	0	3	2	2	1 min.	0%

Time: Indicates the time at which the testing occurred.

Number of Entering Passenger: Shows the number of passengers entering the bus detected by the prototype.

Number of Exiting Passenger: Displays the number of passengers exiting the bus detected by the prototype.

Number of Expected Passenger Detected on App: Represents the number of vacant seats accurately detected by the prototype.

Detection Duration (Seconds): This metric shows how quickly the prototype can identify vacant seats, indicating the effectiveness of the system's real-time analysis and response.

Error Percentage: This column helps identify areas for development by calculating the percentage of mistakes in passenger detection, which provides insight into the margin of error in the prototype's detection capabilities.

Overall Accuracy (Percentage): Shows the overall accuracy of the prototype in detecting vacant seats, calculated as a percentage based on the total number of seats and the accurately detected vacant seats.

Figure 14 presents a time-series graph showing the error percentage in passenger detection over a one-hour period. The data compares the expected number of passengers (based on system inputs) with the actual number detected by the app prototype.

Detection was performed at consistent one-minute intervals, indicating a fixed sampling rate. However, the error percentage varied significantly, ranging from 0% to 68%, highlighting inconsistencies in detection accuracy.

At certain points—such as 4:22 PM (40%) and 4:45 PM (9%)—the system demonstrated relatively low error rates, indicating effective detection during those intervals. In contrast, higher error values were observed at 4:13 PM (64%), 4:24 PM (64%), and 4:44 PM (68%), revealing notable failures in detecting the correct number of passengers.

These discrepancies suggest that while the system can perform accurately under specific conditions, its reliability remains inconsistent. The variability in error percentage may be influenced by external factors (e.g., lighting, passenger movement) or internal system limitations.



Figure 14. Passenger Detection Error Percentage over Time

To enhance consistency and performance, further refinement of the detection algorithm is needed. This may involve improved calibration, advanced noise filtering, or the integration of additional sensors to reduce errors and increase detection accuracy.

3.2.4 Systematic Recording and Analysis: For Object Detection: Day 2

Table 3 presents the dataset related to passenger counts and detections over a specified time period. Each row represents a minute-by-minute breakdown of passengers entering and exiting the vehicle, along with the expected and actual passenger counts detected via the application. The collected data serves as a basis for making adjustments and improvements to enhance the accuracy and efficiency of the prototype system.

Table 3. Systematic Recording and Analysis for Object Detection Performance

Time (HH:MM)	No. of Entering Passenger	No. of Exiting Passenger	No. of Expected Passenger Detected on App	No. of Actual Passenger Detected on App	Detection Duration (Seconds)	Error Percentage
5:30 PM	6	0	6	5	6.27	17%
5:30 PM	1	0	7	7	17.75	0%
5:30 PM	2	0	9	5	15.35	44%
5:31 PM	2	0	11	9	20.73	18%
5:31 PM	1	0	12	6	22.08	50%
5:31 PM	0	0	12	8	5.03	33%
5:32 PM	2	0	14	7	10.71	50%
5:32 PM	1	0	15	9	18.09	40%
5:33 PM	1	0	16	8	9.25	50%
5:34 PM	1	0	17	7	12.55	59%
5:35 PM	0	0	17	9	15.04	47%
5:48 PM	1	1	17	10	8.36	41%
5:48 PM	0	0	17	11	3.85	35%
5:49 PM	0	1	16	9	8.6	44%
5:51 PM	0	0	16	6	12.01	44%
5:53 PM	0	1	15	7	10.47	53%
5:54 PM	0	2	13	12	17.48	8%
5:55 PM	0	0	13	11	22.45	15%
5:56 PM	0	0	13	13	36.25	0%
6:03 PM	1	0	14	11	9.25	21%
6:05 PM	0	0	14	6	45.02	57%
6:08 PM	0	0	14	8	35.24	43%
6:25 PM	0	6	8	7	20.53	13%
6:30 PM	0	0	8	6	24.13	25%
6:36 PM	0	4	4	4	12.31	0%

Time: Indicates the time at which the testing occurred.
Number of Entering Passenger: Shows the number of passengers entering the bus detected by the prototype.
Number of Exiting Passenger: Displays the number of passengers exiting the bus detected by the prototype.
Number of Expected Passenger Detected on App: Represents the number of vacant seats accurately detected by the prototype.
Detection Duration (Seconds): This metric shows how quickly the prototype can identify vacant seats, indicating the effectiveness of the system's real-time analysis and response.
Error Percentage: This column helps identify areas for development by calculating the percentage of mistakes in passenger detection, which provides insight into the margin of error in the prototype's detection capabilities.
Overall Accuracy (Percentage): Shows the overall accuracy of the prototype in detecting vacant seats, calculated as a percentage based on the total number of seats and the accurately detected vacant seats.

3.2.5 Result of Table 3

This dataset provides detailed information on passenger movements within a public transportation system during specific time intervals. Each row represents a distinct time period, denoted by "Time (HH:MM)," and includes data on the number of passengers entering and exiting the system, both expected and detected by the app. The "Detection Duration (Seconds)" column specifies the length of the detection process for each interval. The "Error Percentage" and "Overall Accuracy (Percentage)" columns quantify the system's performance—where the error percentage reflects the deviation between expected and detected counts, and overall accuracy indicates the system's effectiveness.

For example, at 5:30 PM, six passengers were recorded as entering, but only five were detected by the app, resulting in a 17% error rate and an overall accuracy of 83%. This dataset captures temporal variations in passenger counts and detection accuracy, offering valuable insights into the system's performance throughout the day.

3.2.6 Systematic Recording and Analysis for GPS

Table 4 presents real-time detection data collected at various locations in Cagayan de Oro, Misamis Oriental. The table includes the time of detection (HH), a description of the specific location, the expected and actual GPS coordinates, and an indication of whether real-time detection was successful (Yes/No). Each row represents a unique time and location where a detection test was conducted.

Table 4. Systematic Recording and Analysis for GPS

Time (HH:MM)	Location	Expected Coordinates	Actual Coordinates	Real-Time Detection (Yes/No)
3:55 PM	Pimentel Building, Capistrano-Luna Street, Cagayan de Oro	8°29'14.5"N 124°39'04.6"E	8.487362, 124.651283	yes
3:55 PM	Corrales Ext, Cagayan de Oro	8°29'15.1"N 124°39'04.2"E	8.487530, 124.651154	yes
3:56 PM	FMQ2+4H4, Consortium Building, Corrales Ext, Cagayan de Oro	8°29'16.1"N 124°39'05.0"E	8.487812, 124.651375	yes
3:57 PM	FMQ2+JP7, Corrales Ext, Cagayan de Oro	8°29'20.0"N 124°39'06.7"E	8.488890, 124.651863	yes
3:57 PM	ERRA Building II, Corrales Ext, Cagayan de Oro	8°29'25.4"N 124°39'09.1"E	8.490392, 124.652534	yes
3:58 PM	7-1 Baculio St, Cagayan de Oro	8°29'27.5"N 124°39'05.1"E	8.490957, 124.651428	yes
3:58 PM	144 Julio Pacana Street, Cagayan de Oro	8°29'26.7"N 124°39'03.0"E	8.490755, 124.650841	yes
3:58 PM	180-51 Julio Pacana Street, Cagayan de Oro	8°29'20.7"N 124°38'59.6"E	8.489077, 124.649887	yes

Table Continued.

3:59 PM	Cagayan de Oro, Misamis Oriental	8°29'14.4"N 124°38'55.6"E	8.487336, 124.648788	yes
3:59 PM	Julio Pacana Street, Cagayan de Oro	8°29'12.2"N 124°38'54.3"E	8.486730, 124.648407	yes
4:00 PM	Abellanosa St, Cagayan de Oro	8°29'12.0"N 124°38'51.2"E	8.486669, 124.647560	yes
4:00 PM	Abellanosa St, Cagayan de Oro	8°29'12.9"N 124°38'47.2"E	8.486923, 124.646431	yes
4:00 PM	Abellanosa St, Cagayan de Oro	8°29'13.1"N 124°38'46.4"E	8.486976, 124.646217	yes
4:01 PM	Abellanosa St, Cagayan de Oro	8°29'13.3"N 124°38'45.8"E	8.487029, 124.646049	yes
4:01 PM	Abellanosa St, Cagayan de Oro	8°29'13.3"N 124°38'45.8"E	8.487016, 124.646042	yes
4:01 PM	Abellanosa St, Cagayan de Oro	8°29'13.8"N 124°38'44.2"E	8.487167, 124.645599	yes
4:02 PM	93 Abellanosa Street, Manuel Vega St	8°29'14.3"N 124°38'42.7"E	8.487292, 124.645195	yes
4:02 PM	Abellanosa St, Cagayan de Oro	8°29'14.8"N 124°38'41.1"E	8.487455, 124.644753	yes
4:03 PM	93 Abellanosa St, Cagayan de Oro	8°29'15.2"N 124°38'39.7"E	8.487544, 124.644348	yes
4:03 PM	36 F. Abellanosa St, Cagayan de Oro	8°29'17.4"N 124°38'32.4"E	8.488168, 124.642342	yes
4:04 PM	Cagayan de Oro National Hwy, Cagayan de Oro	8°29'20.6"N 124°38'22.6"E	8.489065, 124.639603	yes
4:04 PM	Cagayan de Oro National Hwy, Cagayan de Oro, Misamis Oriental	8°29'20.7"N 124°38'22.0"E	8.489081, 124.639435	yes
4:04 PM	Cagayan de Oro National Hwy, Cagayan de Oro, Misamis Oriental	8°29'20.6"N 124°38'21.9"E	8.489047, 124.639420	yes
4:05 PM	Cagayan de Oro National Hwy, Cagayan de Oro, Misamis Oriental	8°29'22.5"N 124°38'16.3"E	8.489587, 124.637863	yes
4:05 PM	Cagayan de Oro National Hwy, Cagayan de Oro, Misamis Oriental	8°29'26.2"N 124°38'05.5"E	8.490616, 124.634857	yes
4:06 PM	Cagayan de Oro National Hwy, Cagayan de Oro, Misamis Oriental	8°29'33.0"N 124°37'57.7"E	8.492506, 124.632683	yes
4:06 PM	FJWJ+2H7, 86 St Dominique St, Cagayan de Oro, Misamis Oriental	8°29'41.7"N 124°37'52.9"E	8.494915, 124.631348	yes
4:06 PM	Cagayan de Oro National Hwy, Cagayan de Oro, Misamis Oriental	8°29'48.2"N 124°37'49.3"E	8.496732, 124.630350	yes
4:07 PM	Cagayan de Oro National Hwy, Cagayan de Oro, Misamis Oriental	8°29'52.6"N 124°37'47.0"E	8.497936, 124.629707	yes
4:07 PM	Cagayan de Oro National Hwy, Cagayan de Oro, Misamis Oriental	8°29'56.0"N 124°37'45.0"E	8.498878, 124.629173	yes
4:08 PM	Cagayan de Oro National Hwy, Cagayan de Oro, Misamis Oriental	8°30'06.1"N 124°37'39.5"E	8.501684, 124.627632	yes
4:08 PM	Cagayan de Oro National Hwy, Cagayan de Oro, Misamis Oriental	8°30'14.0"N 124°37'34.0"E	8.503885, 124.626106	yes
4:08 PM	Cagayan de Oro National Hwy, Cagayan de Oro, Misamis Oriental	8°30'16.7"N 124°37'26.6"E	8.504631, 124.624054	yes
4:09 PM	Claro M Recto Avenue, Cagayan de Oro, 9000 Misamis Oriental	8°30'16.6"N 124°37'20.1"E	8.504613, 124.622238	yes
4:09 PM	Claro M Recto Avenue, Cagayan de Oro, 9000 Misamis Oriental	8°30'16.5"N 124°37'19.7"E	8.504590, 124.622131	yes
4:10 PM	GJ3C+R6R, Cagayan de Oro, Misamis Oriental	8°30'16.3"N 124°37'14.1"E	8.504539, 124.620590	yes
4:10 PM	Cagayan de Oro, Misamis Oriental	8°30'16.1"N 124°37'07.5"E	8.504469, 124.618752	yes
4:10 PM	Cagayan de Oro National Hwy, Cagayan de Oro, Misamis Oriental	8°30'15.9"N 124°37'01.1"E	8.504406, 124.616959	yes
4:11 PM	Cagayan de Oro National Hwy, Cagayan de Oro, Misamis Oriental	8°30'15.7"N 124°36'58.0"E	8.504372, 124.616096	yes
4:11 PM	Cagayan de Oro National Hwy, Cagayan de Oro, Misamis Oriental	8°30'15.7"N 124°36'56.8"E	8.504351, 124.615784	yes
4:12 PM	Cagayan de Oro National Hwy, Cagayan de Oro, Misamis Oriental	8°30'15.6"N 124°36'53.4"E	8.504331, 124.614830	yes
4:12 PM	Cagayan de Oro National Hwy, Cagayan de Oro, Misamis Oriental	8°30'15.6"N 124°36'52.2"E	8.504345, 124.614494	yes
4:12 PM	Cagayan de Oro National Hwy, Cagayan de Oro, Misamis Oriental	8°30'15.5"N 124°36'47.7"E	8.504298, 124.613258	yes

Table Continued.

4:13 PM	Butuan - Cagayan de Oro - Iligan Rd, Cagayan de Oro, Misamis Oriental	8°30'15.6"N 124°36'45.8"E	8.504324, 124.612732	yes
4:13 PM	Cagayan de Oro National Hwy, Cagayan de Oro, Misamis Oriental	8°30'15.3"N 124°36'41.7"E	8.504259, 124.611572	yes
4:14 PM	Zone 7, before Cool Flow Car Aircon Repair Shop, Bulua Hi-way, Cagayan de Oro, 9000 Misamis Oriental	8°30'15.1"N 124°36'38.3"E	8.504193, 124.610634	yes
4:14 PM	Cagayan de Oro National Hwy, Cagayan de Oro, Misamis Oriental	8°30'15.0"N 124°36'37.2"E	8.504160, 124.610329	yes
4:14 PM	GJ35+PR3, Cagayan de Oro, Misamis Oriental	8°30'14.7"N 124°36'34.2"E	8.504077, 124.609505	yes
4:15 PM	Cagayan de Oro National Hwy, Cagayan de Oro, Misamis Oriental	8°30'14.5"N 124°36'31.9"E	8.504017, 124.608856	yes
4:15 PM	Cagayan de Oro National Hwy, Cagayan de Oro, Misamis Oriental	8°30'13.7"N 124°36'23.2"E	8.503796, 124.606453	yes
4:15 PM	Cagayan de Oro National Hwy, Opol, Misamis Oriental	8°30'13.0"N 124°36'15.0"E	8.503609, 124.604172	yes
4:17 PM	Cagayan de Oro National Hwy, Opol, Misamis Oriental	8°30'12.1"N 124°35'56.8"E	8.503347, 124.599113	yes
4:20 PM	Cagayan de Oro National Hwy, Opol, Misamis Oriental	8°30'29.7"N 124°35'21.3"E	8.508242, 124.589249	yes
4:24 PM	GHCF+RH5, Butuan - Cagayan de Oro - Iligan Rd, Opol, Misamis Oriental	8°31'19.2"N 124°34'26.1"E	8.521995, 124.573929	yes
4:27 PM	Opol, Misamis Oriental	8°31'46.1"N 124°34'17.3"E	8.529482, 124.571472	yes
4:31 PM	Zone 3, Butuan - Cagayan de Oro - Iligan Rd, El Salvador City, Misamis Oriental	8°32'08.1"N 124°33'36.5"E	8.535577, 124.560143	yes
4:34 PM	Butuan - Cagayan de Oro - Iligan Rd, El Salvador City, Misamis Oriental	8°32'10.5"N 124°32'46.7"E	8.536240, 124.546303	yes
4:38 PM	Butuan - Cagayan de Oro - Iligan Rd, El Salvador City, Misamis Oriental	8°33'27.9"N 124°31'51.0"E	8.557755, 124.530830	yes
4:42 PM	HG7F+4CC, Butuan - Cagayan de Oro - Iligan Rd, El Salvador City, Misamis Oriental	8°33'46.2"N 124°31'24.5"E	8.562830, 124.523483	yes
4:43 PM	Butuan - Cagayan de Oro - Iligan Rd, El Salvador City, Misamis Oriental	8°33'58.3"N 124°31'09.8"E	8.566197, 124.519394	yes
4:45 PM	HGF3+HXV, Butuan - Cagayan de Oro - Iligan Rd, El Salvador City, Misamis Oriental	8°34'26.7"N 124°30'17.3"E	8.574091, 124.504807	yes
4:48 PM	HFFP+5X8, Alubijid, Misamis Oriental	8°34'22.8"N 124°29'15.6"E	8.573001, 124.487679	yes
4:50 PM	Butuan - Cagayan de Oro - Iligan Rd, Laguindingan, Misamis Oriental	8°34'17.4"N 124°28'25.5"E	8.571501, 124.473747	yes
4:53 PM	HHFC+24R, Lanao, Alubijid, Cagayan de Oro, 9018 Misamis Oriental	8°34'20.7"N 124°28'12.9"E	8.572422, 124.470238	yes
4:54 PM	Alubijid, Misamis Oriental	8°34'18.6"N 124°28'19.8"E	8.571836, 124.472153	yes
Overall Accuracy (Percentage)				100%

Time: Indicates the time at which the evaluation occurred.

Location: Specifies the location (e.g., bus stop) being evaluated.

Expected Coordinates: This shows the expected latitude and longitude coordinates for the location.

Actual Coordinates: Displays the actual latitude and longitude coordinates detected by the GPS.

Real-Time Detection: Indicates whether the GPS detected the location in real-time (yes or no).

Distance Error (meters): This represents the error in distance between the expected and actual coordinates in meters.

Accuracy: Shows the accuracy of the GPS in detecting the location, calculated as a percentage based on the distance error

3.2.7 Result of Table 4

The table also indicates the overall accuracy percentage, demonstrating that real-time detection was successful 100% of the time. This dataset provides valuable insights into the effectiveness of real-time detection technology in accurately pinpointing specific location.

3.2.8 Systematic Recording and Analysis FOR GPS

Table 5 presents real-time detection data collected at various locations in Cagayan de Oro, Misamis Oriental. The table includes the time of detection (HH), the specific location description, the expected and actual coordinates of the location, and whether real-time detection was successful (Yes/No). Each row represents a different time and location where detection was conducted.

Table 5. Systematic Recording and Analysis for GPS

Time (HH: MM)	Location	Expected Coordinates	Actual Coordinates	Real-Time Detection (Yes/No)
5:09 PM	Alubijid, Misamis Oriental	8°34'18.6"N 124°28'19.8"E	8.571836, 124.472153	yes
5:11 PM	HFFC+24R, Lanao, Alubijid, Cagayan de Oro, 9018 Misamis Oriental	8°34'20.7"N 124°28'12.9"E	8.572422, 124.470238	yes
5:11 PM	Butuan - Cagayan de Oro - Iligan Rd, Laguindingan, Misamis Oriental	8°34'17.4"N 124°28'25.5"E	8.571501, 124.473747	yes
5:11 PM	HFFP+5X8, Alubijid, Misamis Oriental	8°34'22.8"N 124°29'15.6"E	8.573001, 124.487679	yes
5:12 PM	HGF3+HXV, Butuan - Cagayan de Oro - Iligan Rd, El Salvador City, Misamis Oriental	8°34'26.7"N 124°30'17.3"E	8.574091, 124.504807	yes
5:12 PM	Butuan - Cagayan de Oro - Iligan Rd, El Salvador City, Misamis Oriental	8°33'58.3"N 124°31'09.8"E	8.566197, 124.519394	yes
5:12 PM	HG7F+4CC, Butuan - Cagayan de Oro - Iligan Rd, El Salvador City, Misamis Oriental	8°33'46.2"N 124°31'24.5"E	8.562830, 124.523483	yes
5:13 PM	Butuan - Cagayan de Oro - Iligan Rd, El Salvador City, Misamis Oriental	8°33'27.9"N 124°31'51.0"E	8.557755, 124.530830	yes
5:13 PM	Butuan - Cagayan de Oro - Iligan Rd, El Salvador City, Misamis Oriental	8°32'10.5"N 124°32'46.7"E	8.536240, 124.546303	yes
5:14 PM	Zone 3, Butuan - Cagayan de Oro - Iligan Rd, El Salvador City, Misamis Oriental	8°32'08.1"N 124°33'36.5"E	8.535577, 124.560143	yes
5:14 PM	Opol, Misamis Oriental	8°31'46.1"N 124°34'17.3"E	8.529482, 124.571472	yes
5:14 PM	GHCF+RH5, Butuan - Cagayan de Oro - Iligan Rd, Opol, Misamis Oriental	8°31'19.2"N 124°34'26.1"E	8.521995, 124.573929	yes
5:15 PM	Cagayan de Oro National Hwy, Opol, Misamis Oriental	8°30'29.7"N 124°35'21.3"E	8.508242, 124.589249	yes
5:15 PM	Cagayan de Oro National Hwy, Opol, Misamis Oriental	8°30'12.1"N 124°35'56.8"E	8.503347, 124.599113	yes
5:16 PM	Cagayan de Oro National Hwy, Opol, Misamis Oriental	8°30'13.0"N 124°36'15.0"E	8.503609, 124.604172	yes
5:16 PM	Cagayan de Oro National Hwy, Cagayan de Oro, Misamis Oriental	8°30'13.7"N 124°36'23.2"E	8.503796, 124.606453	yes
5:17 PM	Cagayan de Oro National Hwy, Cagayan de Oro, Misamis Oriental	8°30'14.5"N 124°36'31.9"E	8.504017, 124.608856	yes
5:17 PM	GJ35+PR3, Cagayan de Oro, Misamis Oriental	8°30'14.7"N 124°36'34.2"E	8.504077, 124.609505	yes
5:17 PM	Cagayan de Oro National Hwy, Cagayan de Oro, Misamis Oriental	8°30'15.0"N 124°36'37.2"E	8.504160, 124.610329	yes

Table Continued.

5:18 PM	Zone 7, before Cool Flow Car Aircon Repair Shop, Bulua Hi-way, Cagayan de Oro, 9000 Misamis Oriental	8°30'15.1"N 124°36'38.3"E	8.504193, 124.610634	yes
5:18 PM	Cagayan de Oro National Hwy, Cagayan de Oro, Misamis Oriental	8°30'15.3"N 124°36'41.7"E	8.504259, 124.611572	yes
5:18 PM	Butuan - Cagayan de Oro - Iligan Rd, Cagayan de Oro, Misamis Oriental	8°30'15.6"N 124°36'45.8"E	8.504324, 124.612732	yes
5:19 PM	Cagayan de Oro National Hwy, Cagayan de Oro, Misamis Oriental	8°30'15.5"N 124°36'47.7"E	8.504298, 124.613258	yes
5:19 PM	Cagayan de Oro National Hwy, Cagayan de Oro, Misamis Oriental	8°30'15.6"N 124°36'52.2"E	8.504345, 124.614494	yes
5:19 PM	Cagayan de Oro National Hwy, Cagayan de Oro, Misamis Oriental	8°30'15.6"N 124°36'53.4"E	8.504331, 124.614830	yes
5:20 PM	Cagayan de Oro National Hwy, Cagayan de Oro, Misamis Oriental	8°30'15.7"N 124°36'56.8"E	8.504351, 124.615784	yes
5:20 PM	Cagayan de Oro National Hwy, Cagayan de Oro, Misamis Oriental	8°30'15.7"N 124°36'58.0"E	8.504372, 124.616096	yes
5:20 PM	Cagayan de Oro National Hwy, Cagayan de Oro, Misamis Oriental	8°30'15.9"N 124°37'01.1"E	8.504406, 124.616959	yes
5:21 PM	Cagayan de Oro, Misamis Oriental	8°30'16.1"N 124°37'07.5"E	8.504469, 124.618752	yes
5:21 PM	GJ3C+R6R, Cagayan de Oro, Misamis Oriental	8°30'16.3"N 124°37'14.1"E	8.504539, 124.620590	yes
5:22 PM	Claro M Recto Avenue, Cagayan de Oro, 9000 Misamis Oriental	8°30'16.5"N 124°37'19.7"E	8.504590, 124.622131	yes
5:22 PM	Claro M Recto Avenue, Cagayan de Oro, 9000 Misamis Oriental	8°30'16.6"N 124°37'20.1"E	8.504613, 124.622238	yes
5:22 PM	Cagayan de Oro National Hwy, Cagayan de Oro, Misamis Oriental	8°30'16.7"N 124°37'26.6"E	8.504631, 124.624054	yes
5:23 PM	Cagayan de Oro National Hwy, Cagayan de Oro, Misamis Oriental	8°30'14.0"N 124°37'34.0"E	8.503885, 124.626106	yes
5:23 PM	Cagayan de Oro National Hwy, Cagayan de Oro, Misamis Oriental	8°30'06.1"N 124°37'39.5"E	8.501684, 124.627632	yes
5:23 PM	Cagayan de Oro National Hwy, Cagayan de Oro, Misamis Oriental	8°29'56.0"N 124°37'45.0"E	8.498878, 124.629173	yes
5:24 PM	Cagayan de Oro National Hwy, Cagayan de Oro, Misamis Oriental	8°29'52.6"N 124°37'47.0"E	8.497936, 124.629707	yes
5:24 PM	Cagayan de Oro National Hwy, Cagayan de Oro, Misamis Oriental	8°29'48.2"N 124°37'49.3"E	8.496732, 124.630350	yes
5:24 PM	FJWJ+2H7, 86 St Dominique St, Cagayan de Oro, Misamis Oriental	8°29'41.7"N 124°37'52.9"E	8.494915, 124.631348	yes
5:25 PM	Cagayan de Oro National Hwy, Cagayan de Oro, Misamis Oriental	8°29'33.0"N 124°37'57.7"E	8.492506, 124.632683	yes
5:25 PM	Cagayan de Oro National Hwy, Cagayan de Oro, Misamis Oriental	8°29'26.2"N 124°38'05.5"E	8.490616, 124.634857	yes
5:30 PM	Cagayan de Oro National Hwy, Cagayan de Oro, Misamis Oriental	8°29'22.5"N 124°38'16.3"E	8.489587, 124.637863	yes
5:30 PM	Cagayan de Oro National Hwy, Cagayan de Oro, Misamis Oriental	8°29'20.6"N 124°38'21.9"E	8.489047, 124.639420	yes
5:30 PM	Cagayan de Oro National Hwy, Cagayan de Oro, Misamis Oriental	8°29'20.7"N 124°38'22.0"E	8.489081, 124.639435	yes
5:31 PM	Cagayan de Oro National Hwy, Cagayan de Oro	8°29'20.6"N 124°38'22.6"E	8.489065, 124.639603	yes
5:31 PM	36 F. Abellanos St, Cagayan de Oro	8°29'17.4"N 124°38'32.4"E	8.488168, 124.642342	yes
5:36 PM	93 Abellanos St, Cagayan de Oro	8°29'15.2"N 124°38'39.7"E	8.487544, 124.644348	yes
5:40 PM	Abellanos St, Cagayan de Oro	8°29'14.8"N 124°38'41.1"E	8.487455, 124.644753	yes
5:43 PM	93 Abellanos Street, Manuel Vega St	8°29'14.3"N 124°38'42.7"E	8.487292, 124.645195	yes
5:47 PM	18 Abellanos St, Cagayan de Oro, Misamis Oriental	8°29'14.5"N 124°38'42.1"E	8.487350, 124.645020	yes
5:48 PM	Abellanos St, Cagayan de Oro, Misamis Oriental	8°29'13.7"N 124°38'44.6"E	8.487126, 124.645714	yes
5:53 PM	Cagayan de Oro, Misamis Oriental	8°29'13.0"N 124°38'46.4"E	8.486955, 124.646210	yes
5:57 PM	Abellanos St, Cagayan de Oro, Misamis Oriental	8°29'12.7"N 124°38'47.8"E	8.486847, 124.646606	yes
6:00 PM	Abellanos St, Cagayan de Oro, Misamis Oriental	8°29'12.6"N 124°38'48.6"E	8.486823, 124.646828	yes
6:10 PM	Abellanos St, Cagayan de Oro, Misamis Oriental	8°29'11.8"N 124°38'51.2"E	8.486615, 124.647568	yes
6:15 PM	Abellanos St, Cagayan de Oro, Misamis Oriental	8°29'11.2"N 124°38'53.4"E	8.486456, 124.648163	yes
6:17 PM	Abellanos St, Cagayan de Oro, Misamis Oriental	8°29'10.7"N 124°38'57.9"E	8.486304, 124.649406	yes

Table Continued.

6:23 PM	Centrio Mall, Capt. Vicente Roa St, Cagayan de Oro, Misamis Oriental	8°29'10.6"N 124°38'58.7"E	8.486282, 124.649635	yes
6:25 PM	Gaisano City, Claro M. Recto Ave, Cagayan de Oro, 9000 Misamis Oriental	8°29'10.6"N 124°38'59.4"E	8.486282, 124.649826	yes
6:28 PM	Gaisano City, Claro M. Recto Ave, Cagayan de Oro, 9000 Misamis Oriental	8°29'10.2"N 124°39'00.7"E	8.486169, 124.650185	yes
6:31 PM	Claro M. Recto Ave, Cagayan de Oro, Misamis Oriental	8°29'10.0"N 124°39'01.1"E	8.486113, 124.650291	yes
6:32 PM	Claro M. Recto Ave, Cagayan de Oro, 9000 Misamis Oriental	8°29'10.1"N 124°39'02.4"E	8.486149, 124.650658	yes
6:35 PM	Corrales Ext, Cagayan de Oro, Misamis Oriental	8°29'11.8"N 124°39'02.5"E	8.486600, 124.650681	yes
6:36 PM	Gaisano Mall Parking Lot, Cagayan de Oro, 9000 Misamis Oriental		8.487342, 124.651459	yes
6:36 PM	Gaisano Mall Parking Lot, Cagayan de Oro, 9000 Misamis Oriental	8°29'14.2"N 124°39'05.8"E	8.487267, 124.651596	yes
				100% accurate

3.2.9 Result of Table 5

The table also indicates the overall accuracy percentage, demonstrating that real-time detection was successful 100% of the time. This dataset provides valuable insights into the effectiveness of real-time detection technology in accurately pinpointing specific location.

3.3 Dataset Scale and Representativeness

The FOMO-based occupancy detection model was trained on a curated dataset of [X] high-resolution images (320×240 pixels), captured under real-world bus operating conditions (day/night, varying occupancy levels) using the ESP32-CAM’s OV2640 sensor (native 1600×1200 resolution, downsampled onboard). Each image was annotated at the seat level using bounding boxes to indicate occupancy status (person/vacant). The dataset was split 80:20 for training and validation, with data augmentation techniques—including ±15° rotation and ±20% brightness adjustments—applied to improve robustness for edge deployment.

During real-world evaluation over a two-day period (12 hours of operation per day), the system demonstrated consistent data capture, generating 11 occupancy records within a representative 6-minute window. This resulted in 2,640 validated records, with an extrapolated data rate of approximately 1.83 detection points per minute, covering both peak and off-peak boarding conditions. This operationally derived dataset, in combination with the strategically optimized training framework—tailored to the ESP32-CAM’s resolution and memory constraints—enabled the model to achieve a balance of 95.3% mean average precision (mAP) and 23.6 FPS inference speed, confirming its reliability under variable real-world environmental conditions.

3.4 Occlusion Handling Strategy

To mitigate errors caused by overlapping passenger contours (e.g., foreground occupants occluding rear-seated passengers), this study employed seat-centric annotation and camera perspective optimization. During dataset creation, all annotations were bound to predefined seat regions rather than using free-form person detection. Each seat was classified as occupied or vacant based on the visibility of key body parts (e.g., head or shoulders), even under partial occlusion.

The camera was positioned overhead and angled downward to minimize inter-seat overlap, while the FOMO architecture's sensitivity to partial features—trained on 320×240 px images—enabled robust detection of truncated or obscured passengers. Data augmentation included synthetic occlusion (e.g., random rectangular masks over seats) to strengthen model resilience against real-world obstructions.

During inference, per-seat confidence thresholds—validated to achieve >0.85 precision—were applied to reduce false negatives caused by heavy occlusion, thereby prioritizing reliability over exhaustive detection in edge cases.

4. Conclusion and Recommendation

This study aimed to design and implement a Smart Assistance System for Public Transport, focusing on modern buses, to address key challenges such as bus arrival time prediction and seat vacancy detection. Leveraging advancements in technology—including GPS, the ESP32-CAM with the FOMO object detection algorithm, and integration with Google Maps—the system provides real-time monitoring and a user-friendly interface for passengers.

The methodology involved system design and integration of multiple components: a passenger counting and seat availability detection module, GPS-based tracking, development of a smart application, and backend integration using technologies such as Node.js, React Native, Firebase, and the Google Maps API.

Future researchers are advised to consider the following recommendations based on the study's findings. First, while Edge Impulse offers convenience, it may not be the most reliable platform for object detection training due to limitations in accuracy and performance. Exploring alternative platforms or developing custom solutions is recommended for more robust object detection. Second, the angle of installation plays a critical role in detection performance. For optimal results, the prototype should be installed at a fixed downward-facing angle and securely fastened with screws to ensure stability during operation. This minimizes movement-induced detection errors and ensures consistent system performance.

Although the system demonstrates accurate passenger counting, it is currently unable to detect seat occupancy caused by non-human objects (e.g., bags). This limitation arises from model optimization for human detection within the ESP32-CAM's resource constraints. To address this, conservative capacity heuristics have been implemented (Section 3.2.1), with plans for future improvement through multi-class object detection integration.

5. Acknowledgement

The researchers extend their sincere appreciation to all individuals who played a significant role in the realization of this study. Gratitude is also expressed to the panel members and laboratory technicians for their valuable feedback and constructive suggestions throughout the research process. Appreciation is also given to the family of the researchers for their financial and emotional support. The encouragement and moral support provided by friends are gratefully acknowledged. Sincere thanks are also extended to the respondents who participated in the data collection phase. Lastly, the researchers recognize divine providence for the strength and wisdom granted throughout the completion of this study.

6. References

Akter, R., Khandaker, M.J.H., Ahmed, S., Mugdho, M.M., & Haque, A.K.M.B. (2020). RFID-based smart transportation system with Android application. 2020 2nd International Conference on Innovative Mechanisms for Industry Applications (ICIMIA), Bangalore, India, 614–619. <https://doi.org/10.1109/ICIMIA48430.2020.9074869>

- AI-Thinker (2017). ESP32-CAM development board specifications [Hardware module]. Shenzhen, China. Retrieved from http://www.ai-thinker.com/pro_view-24.html
- Al-Khedher, M. (2011). Hybrid GPS-GSM localization of automobile tracking system. *International Journal of Computer Science & Information Technology (IJCSIT)*, 3(6), 75–85. <https://doi.org/10.5121/ijcsit.2011.3606>
- Bhardwaj, R., Kumar, K., Dewangan, P., Yadaw, L.R., & Mokashe, H. (2023). Bus timing app. *International Research Journal of Modernization in Engineering Technology and Science*, 5(5). <https://doi.org/10.56726/IRJMET539203>
- Boyle, L., Baumann, N., Heo, S., & Magno, M. (2023). Enhancing lightweight neural networks for small object detection in IoT applications. 2023 IEEE Sensors, Vienna, Austria, 1–4. <https://doi.org/10.1109/SENSOR556945.2023.10325126>
- Creality (2016). CR-10 3D printer user manual. Creality3D. <https://www.creality.com>
- Da Silva, J., Flores, T., Júnior, S., & Silva, I. (2023). TinyML-based pothole detection: A comparative analysis of YOLO and FOMO model performance. 2023 IEEE Latin American Conference on Computational Intelligence (LA-CCI), Recife-Pe, Brazil, 1–6. <https://doi.org/10.1109/LA-CCI58595.2023.10409357>
- Gotthard, R., & Broström, M. (2023). Edge machine learning for wildlife conservation: A part of the Ngulia project (Master's thesis). Department of Electrical Engineering, Linköping University, Linköping, Sweden. Retrieved from <http://urn.kb.se/resolve?urn=urn:nbn:se:liu:diva-193068>
- Google LLC (2005). Google Maps API [Application programming interface]. Mountain View, CA, USA. Retrieved from <https://developers.google.com/maps>
- Google LLC (2016). Firebase Realtime Database [Cloud platform]. Mountain View, CA, USA. Retrieved from <https://firebase.google.com>
- Hamid, A.H.F.A., Chang, K.W., Rashid, R.A., Mohd, A., Abdullah, M.S., Sarijari, M.A., & Abbas, M. (2019). Smart vehicle monitoring and analysis system with IoT technology. *International Journal of Integrated Engineering*, 11(4), 100–108. <https://doi.org/10.30880/ijie.2019.11.04.016>
- Howard, A., Sandler, M., Chu, G., Chen, L.-C., Chen, B., Tan, M., Wang, W., Zhu, Y., ... & Adam, H. (2019). Searching for MobileNetV3. arXiv preprint arXiv:1905.02244. Retrieved from <https://arxiv.org/abs/1905.02244>
- Islam, K., & Afzal, U. (2020). Framework for passenger seat availability using face detection in passenger buses. *arXiv*. <https://doi.org/10.48550/arXiv.2007.05906>

Jocher, G., & Ultralytics (2020). YOLOv5: Object detection architecture and models [Computer software]. Retrieved from <https://github.com/ultralytics/yolov5>

Lau, E.C.-W. (2013). Simple bus tracking system. *Journal of Advanced Computer Science and Technology Research*, 3(1), 60–70. <https://www.sign-ific-ance.co.uk/index.php/JACSTR/article/view/403>

Lee, S., Tewolde, G., & Kwon, J. (2014). Design and implementation of vehicle tracking systems using GPS/GSM/GPRS technology and smartphone application. 2014 IEEE World Forum on Internet of Things (WF-IoT), Seoul, South Korea, 353–358. <http://dx.doi.org/10.1109/WF-IoT.2014.6803187>

Meta Platforms, Inc. (2015). React Native [Mobile framework]. Menlo Park, CA, USA. Retrieved from <https://reactnative.dev>

Microsoft Corporation (2015). Visual Studio Code [Software]. Redmond, WA, USA. Retrieved from <https://code.visualstudio.com>

Muslikhin, M., Aryanti, A., & Wang, M.-S. (2024). Navigating unstructured space: Deep action learning-based obstacle avoidance system for indoor automated guided vehicles. *Electronics*, 13(2), 420. <https://doi.org/10.3390/electronics13020420>

Novak, M., Doležal, P., Budík, O., Ptáček, L., Geyer, J., Davidková, M., & Prokýšek, M. (2024). Intelligent inspection probe for monitoring bark beetle activities using embedded IoT real-time object detection. *Engineering Science and Technology, an International Journal*, 51, 101637. <https://doi.org/10.1016/j.jestech.2024.101637>

Paul, J.P., Jone, A.A., Sagayam, M., Vasantha Packiavathy, I.S., Jesintha, E., Rinsy, J.J., Dang, H., & Pomplun, M. (2021). IoT based remote transit vehicle monitoring and seat display system. *Przegląd Elektrotechniczny*, 97(5), 131–135. <https://doi.org/10.15199/48.2021.05.25>

Rathod, R., & Khot, S.T. (2016). Smart assistance for public transport system. 2016 International Conference on Inventive Computation Technologies (ICICT), Coimbatore, India, 943–947. <https://doi.org/10.1109/INVENTIVE.2016.7830206>

Skhosana, M., & Absalom, E.E. (2020). Irenbus: A real-time public transport management system. 2020 Conference on Information Communications Technology and Society (ICTAS), Durban, South Africa, 1–7. <https://doi.org/10.1109/ICTAS47918.2020.234000>

Sojol, J.I., Piya, N.F., Sadman, S., & Motahar, T. (2018). Smart bus: An automated passenger counting system. *International Journal of Pure and Applied Mathematics*, 118(18), 3169–3177. Retrieved from <https://acadpubl.eu/hub/2018-118-18/1/316.pdf>

Sonawane, A., Gogri, K., Bhanushali, A., & Khairnar, M. (2020). Real-time bus tracking system. *International Journal of Engineering Research & Technology (IJERT)*, 9(6), 829–831. <https://doi.org/10.17577/IJERTV9IS060545>

Sungur, C., Babaoglu, I., & Sungur, A. (2015). Smart bus station-passenger information system. 2015 2nd International Conference on Information Science and Control Engineering, Shanghai, China, 921–925. <https://doi.org/10.1109/ICISCE.2015.209>

Texas Instruments (1999). LM2596 Simple Switcher power converter [Data sheet]. Dallas, TX, USA. Retrieved from <https://www.ti.com/product/LM2596>

u-blox. (2010). NEO-6 series modular GPS/GNSS receiver [Hardware module]. Thalwil, Switzerland. Retrieved from <https://www.u-blox.com/en/product/neo-6-series>

Ultralytics (2021). YOLOv5: Version 6.1 [Computer software]. <https://github.com/ultralytics/yolov5/releases/tag/v6.1>

Wahyu, Y., & Adi, P.D.P. (2022). A performance evaluation of ESP32 camera face recognition for various projects. *Internet of Things and Artificial Intelligence Journal*, 2(1), 10–21. <https://doi.org/10.31763/iota.v2i1.512>

Xiaojian, F., Zhang, J., Chen, J., Wang, G., Zhang, L., & Li, R. (2018). Design of intelligent bus positioning based on Internet of Things for smart campus. *IEEE Access*, 6, 60005–60015. <https://doi.org/10.1109/ACCESS.2018.2874083>