Published in AJSE, Vol: 25, Issue:1
Received on 5th June 2025
Revised on 24th August 2026
Published on 31st August 2026

AIUB JOURNAL OF SCIENCE AND ENGINEERING
ISSN: 1608 - 3679 (print) 2520 - 4890 (Online)
https://doi.org/10.53799/stn2d409

Performance Evaluation of an IoT-Enabled Mobile Robotic System for Smart Homes

Nafiz Ahmed Chisty*, Mohammad Alif Arman, SM Walid, Mohammad Shorif Uddin

Abstract*— This paper introduces a cost-effective mobile robotic platform for automatic patrolling of multi-room apartments, which includes on-board vision-based occupancy detection using a model implemented on the Mobile Single Shot MultiBox Detector (MobileNet-SSD) on a Raspberry Pi 4B and load control using a relay module on ESP32. There is an independent kitchen gas-safety subsystem operating independently of the patrol cycle, characterized by 50 trials (mean response time 18.4 s, SD 4.3 s) and a false-alarm rate of 0.18 events/hour of cooking. Total hardware expense is 23,484 BDT (≈ USD 187, at 122 BDT/USD, 2025–2026 average). The detector attains an accuracy of 82% (95% Wilson score confidence interval (CI): 69%–90%) on a test set of 50 images, a precision of 95.7%, and a recall of 73.3%. For this two-room prototype, the latency for Google Firebase Realtime Database is lower than 500 milliseconds (ms) with a mean update time of 271 ms when no humans are detected and 338 ms when they are detected. A fan and lighting load is modeled to result in a net energy savings of 259 kilowatt-hours (kWh) per year, which translates into annual bill reductions of approximately 2,007 BDT under the Bangladeshi ascending-slab tariff. Simple payback is approximately 11.4 years for fan/lighting as built, reducing to 5.8 years with lower-cost compute hardware. The proposed system improves upon static passive infrared (PIR) sensors by detecting stationary occupants, ensuring privacy through on-device inference, integrating gas safety monitoring, and distributing the cost of a single sensing platform across multiple rooms. This work is a proof-of-concept prototype in support of the United Nations (UN) Sustainable Development Goals (SDGs) 7, 9, 11, 12, and 13.

***Index Terms*—Internet of Things (IoT); MobileNet-SSD; navigation; occupancy-responsive load control; techno-economic analysis.**

I. Introduction

With the growth of electronic appliances, electricity consumption in domestic buildings has also increased rapidly, increasing the economic cost of electricity for domestic consumers and the electricity provider, especially in urban areas, where electricity consumption is higher [1,2].

Nafiz Ahmed Chisty, PhD candidate, Department of Computer Science and Engineering, Jahangirnagar University; Associate Professor, Department of EEE, AIUB, Dhaka, Bangladesh (email: chisty@aiub.edu)
Mohammad Alif Arman, Assistant Professor, Department of EEE, AIUB, Dhaka, Bangladesh (email: dr.alifarman@aiub.edu)
SM Walid, Student, Department of EEE, AIUB, Dhaka, Bangladesh (email: 23-50005-1@student.aiub.edu)
Mohammad Shorif Uddin, Professor, Department of Computer Science and Engineering, Jahangirnagar University, Savar, Bangladesh (email: shorifuddin@juniv.edu)

Much of this consumption is lost due to the tendency to leave lights, fans, and air conditioning in unused areas [3,4]. Manual switching and timer-based controllers are conventional mitigation strategies that require complete occupant cooperation and cannot react to real-time occupancy changes. The issue is particularly acute in Bangladesh where electricity demand is dominated by the residential segment and where dependence on imported fossil fuels for electricity generation is high. In recent months, households have thus been facing intermittent electricity cuts across the country. The national tariff is an ascending-slab tariff, meaning that the more consumption that occurs each month, the higher the price for the marginal unit of consumption. The highest-consuming households, therefore, bear the highest marginal rates and are also likely to have the highest amount of idle load in the high-cost tariff bands, so that each kWh of avoided consumption has a disproportionately large financial impact.

The current smart-home technologies typically offer privacy or cost but not both. Fixed occupancy sensors scale the system cost linearly with the number of rooms [4] and fixed camera-based nodes provide finer-grained detection but are known to raise privacy concerns because they typically upload images to external servers. The autonomous mobile robot (AMR) literature has shown that environmental sensing from a mobile platform can help to determine the occupancy of that space, but most of these environmental sensing systems are concerned with collecting environmental data instead of directly actuating electrical loads [5].

The purpose of this paper is to close these gaps by using a single mobile robot that performs several functions, each of which, in isolation, is normally carried out by a separate static robot. The contributions provided, which are limited to a prototype feasibility study, are as follows:

1. One platform delivers on-device vision-based occupancy detection, relay-based load control, cloud synchronization, and an independent gas-safety subsystem.
2. All image inference is done on the device (Raspberry Pi 4B); no image data will leave the robot.
3. Engineering model that is clearly defined and quantifies the energy saving and payback with explicit parameters such as a patrol-timing penalty, robot self-consumption, etc. in the context of the Bangladeshi ascending-slab tariff.
4. Detection accuracy is presented with a confidence interval and cloud latency, hardware reliability and failure cases are quantified in a real 2-room prototype.
5. This work is a working prototype that was developed to determine technical feasibility. It does not claim to be the economically better solution for load-only occupancy (LOO) control applications compared with fixed PIR sensors. All energy and economic estimates are based on the model, and not on actual measurements taken from a deployed system.

The primary objectives are: (i) to design and implement a low-cost mobile robotic platform for occupancy-responsive load control in multi-room residential settings; (ii) to evaluate its detection performance experimentally against a conventional PIR baseline under identical occupancy scenarios; (iii) to quantify potential energy savings and economic payback through a transparent techno-economic model tailored to the Bangladeshi ascending-slab tariff; and (iv) to identify key limitations and outline future research directions toward real-world deployment.

The problems that the work is inspired by are three pressing problems in Bangladesh: 1) the rising electric tariff is very much affecting the lower-income households, 2) frequent load shedding due to lack of power supply, and 3) the absence of scalable, privacy-preserving smart-home solution which can detect stationary or absent occupants without having to install sensors room-wise.

This paper is structured as follows: Section II presents related work and research gaps. The system architecture and methodology are given in Section III. Implementation is explained in Section IV. The experimental results are reported in Section V which include detection performance, latency, and the results of the techno-economic analysis. The limitations of PIR and its correlation with sustainability are discussed in Section VI and compared with the experimental data. Future work is summarized and concluded in Section VII.

II. RELATED WORK

A. IoT-Based Home Energy Management

Internet of Things (IoT) has changed Home Energy Management Systems (HEMS) from time-based control and home automation to smart systems with occupant awareness, and energy efficiency without sacrificing occupant comfort [6,7]. In the early systems, schedules and threshold rules were preprogrammed, but this approach was insufficient in the presence of varying occupant behavior [6]. To tackle this, Safdarian et al. [8] introduced a reinforcement learning-based dynamic appliance scheduling approach that considers energy pricing and user preferences. Similarly, comfort-based optimization-based scheduling was developed by Rastegar et al. [9] to minimize electricity costs while providing comfort. Machine learning has also been used for energy consumption forecasting to help with proactive management instead of reactive management [10].

One paradigm that has come to the fore is occupancy-aware energy management. The low-cost sensors such as early PIR, ultrasonic and magnetic cannot detect an occupant when it is stationary and require multiple nodes per room [11]. However, the use of these sensors has been greatly improved by machine learning algorithms [12] and, as demonstrated by Razavi et al [4] occupancy can even be predicted utilizing only data from the wall-mounted smart-meters that are already installed in homes. Recently, Natarajan et al. [3] demonstrated the robustness of machine learning methods over a range of occupancy conditions compared to the robustness of rule-based algorithms.

Deep learning has further expanded HEMS capabilities. Occupant-centric control has been identified as the current major paradigm for residential energy management systems [13]. Recently, conventional optimization approaches have increasingly been complemented by data-driven methods, including convolutional neural networks (CNNs), long short-term memory (LSTM) networks, reinforcement learning (RL), and deep reinforcement learning (DRL), which are being investigated for adaptive energy management. For example, Mocanu et al. [14] used deep reinforcement learning to enable autonomous agents to learn optimal energy control policies without explicit rules. For residential load forecasting Kong et al. [15] have used LSTM networks which have proven to be better than the traditional statistical models. Zhang and Kong [16] used deep learning and multi-objective optimization to achieve a balance between energy savings and occupant comfort. There are also studies that demonstrate better accuracy using sensor fusion and learning techniques compared with single-sensor techniques in complex indoor environments [17]

Cloud Computing was superseded by Fog and Edge Computing, which mitigates latency, bandwidth, and privacy issues, by making decisions locally [18,19]. Most HEMS research however assumes fixed sensor or camera per room, with system cost and complexity increasing with the size of the building. The idea of a shared mobile sensing platform for monitoring multiple rooms is not studied extensively, and this work tackles this challenge by incorporating mobile occupancy sensing, edge intelligence, relay-level appliance control and cloud connectivity into a single low-cost robotic platform.

B. Autonomous Indoor Robot Navigation

From basic reactive navigation to intelligent systems based on perception, indoor robot navigation has come a long way. The first approach of line-following was based on PID control and infrared sensors on fixed tracks [20] and the wall-following was by using a rotating ultrasonic scanner for avoiding obstacles without external positioning [21] These methods are inexpensive and only applicable to the structured setting and pre-planned courses of action.

With growing computational power on board, studies focused on simultaneous localization and mapping (SLAM), visual odometry and navigation using LiDAR data, allowing robots to localize and map an unknown environment. ORB-SLAM2 [22] is a widely used feature-based visual SLAM technique that works for monocular, stereo and RGB-D cameras. To improve the robustness of LiDAR-based SLAM in the indoor environment, Shan et al. [23] proposed the combination of LiDAR and inertial sensor data. But SLAM-based methods are still expensive and require computation, which is suitable for only low-cost residential systems. For smart homes in repetitive and predictable layouts the lightweight navigation methods are still viable.

A few studies have examined the utilization of mobile robots in the home. Dario et al. [24] created the MOVAID robotic assistant for assistive tasks and navigation in a domestic environment. Vanus et al. [5] proposed an occupancy monitoring mobile robot using IoT in home-care applications. But these systems did not incorporate energy management, relay level appliance control or safety monitoring. Energy management without human control and intelligent home control for obtaining occupancy information are underexplored aspects of autonomous navigation.

C. Vision-Based Occupancy Detection on the Edge

From motion detection (PIR) to intelligent vision detection systems. Although PIR sensors are simple, low cost and power efficient, they are unable to detect stationary occupants [25]. Motion-based detection can be directly replaced by computer vision.

Illumination variations and background clutter were the challenges faced by the early vision-based systems based on hand crafted features such as Haar cascades and Histogram of Oriented Gradients (HOG) [26]. Since then, deep convolutional neural networks (CNNs) have been able to perform better in terms of accuracy by learning the visual features of all scales. Guo et al. [27] applied edge computing and model predictive control (MPC) for building occupancy detection using YOLOv5 in real time.

MobileNet-SSD is appealing for low-power, embedded applications because of its compactness and efficiency. MobileNet-SSD was demonstrated to be much better than the traditional OpenCV-based detection by Honmote et al. [28] and Kamath et. al. [29] deployed it on a Raspberry Pi for real-time human following. Surveys have shown that lightweight CNN models are suitable for the edge to provide optimal detection performance, low latency, memory consumption, and energy consumption [30,31,32]. To reduce further the number of computations, Sandler et al. [33] introduced the MobileNetV2, which uses inverted residual blocks. Liu et al. [34] proposed SSD in real-time detection system with enhanced speed/accuracy trade-off. Redmon et al. [35] used a real-time regression approach for object detection. MobileNet-SSD has been shown to be viable on Raspberry Pi by Howard [36] and Lane et al. [37] studied efficient inference methods for mobile and embedded devices.

D.Kitchen Safety and Gas Detection

The first generation of gas monitoring systems introduced semiconductor MQ-series sensors (typically MQ-2, MQ-5, or MQ-6) for detecting combustible gases including liquefied petroleum gas (LPG), methane, and propane, with threshold-based alarms triggered when analog voltage readings exceeded empirically determined setpoints [38]. While these systems are simple to construct and inexpensive, they are susceptible to environmental drift (temperature and humidity sensitivity), require periodic recalibration, and typically cannot distinguish between hazardous leaks and transient fluctuations from cooking activities. Recent studies have addressed these limitations through IoT connectivity, cloud logging, and machine-learning-based classification to improve reliability and reduce false alarms. Khakim et al. [38] evaluated MQ-2, MQ-4, MQ-5, and MQ-6 sensors for LPG detection across 15–31 °C, reporting MQ-2 as the best performer (R^2 = 0.96, RMSE 7–12 ppm); all four sensors showed usable sensitivity in the 200–10,000 ppm range. We use the MQ-6 for its local availability and pin-compatibility with the MQ series. Jena et al. [39] demonstrated an MQ-6-based LPG detection system with automatic alarm generation, GSM notification, and exhaust fan activation. Kumar et al. [40] enhanced gas detection reliability by integrating IoT communication with machine learning (support vector machine classification of sensor time-series features) for automated emergency responses including ventilation activation, audible/visual alarms, and remote cloud monitoring with push notifications to occupants' mobile devices.

Multi-sensor fusion techniques, which integrate the output of gas, temperature, smoke and flame sensors have further enhanced detection accuracy and minimized false alarms. But current kitchen safety systems are standalone IoT applications, not connected to the smart-home energy management system. They are not often used in conjunction with occupancy detection, mobile robotic monitoring, or intelligent appliance control. The proposed solution does this by combining an independent gas-detection IoT module to the energy-management platform on the mobile device, which is occupancy-aware.

E. Research Gap

Significant advancements have been made in four areas: intelligent HEMS [1],[2], autonomous navigation [5], vision-based occupancy detection [28], and IoT-enabled kitchen safety [38]. HEMS have shifted from rule-based scheduling to AI-driven optimization with edge/fog computing enabling real-time decision-making [8],[9],[10]. Mobile robotics has progressed from line-following [20],[21] to SLAM-based navigation [22],[23]. PIR sensors have been replaced by lightweight deep learning models for occupancy detection [25],[26],[27], and kitchen safety now incorporates IoT gas detection and emergency response [39],[40].

However, these technologies have developed in isolation. HEMS rely on fixed sensing infrastructure, scaling cost with room count [4]. Mobile robots focus on navigation or monitoring rather than integrated energy management [5],[24]. Vision-based systems assume stationary cameras [26],[27], and gas monitoring remains an independent subsystem [38],[39].

To the best of our knowledge, no previous study has integrated (i) mobile indoor navigation, (ii) on-device vision-based occupancy detection, (iii) relay-level appliance control, (iv) cloud synchronization, and (v) an independent kitchen gas-safety subsystem within a unified mobile edge-computing framework. The proposed system addresses this research gap by integrating these capabilities into a single mobile platform for intelligent multi-room smart-home energy management.

In Table I the main features of the current home energy management systems, autonomous robotic navigation, vision-based occupancy detection, and kitchen safety systems are summarized. The comparison illuminates the development of single technologies and illustrates that the majority of the current research on the technologies is directed toward a single functionality or application area. Specifically, the previous systems are mainly based on static sensing nodes and are seldom integrated with the integration of mobile multi-room sensing, edge-based vision inference, relay-type appliance control, cloud synchronization, and kitchen gas safety. This comparison clearly demonstrates the research gap that the proposed system addresses.

TABLE I
COMPARISON OF REPRESENTATIVE STUDIES RELATED TO THE PROPOSED SYSTEM

Related works	HEMS	Occupancy Detection	Edge AI	Mobile Sensing	Relay Control	Gas Detection	Integrated Platform
Proposed System	✓	✓	✓	✓	✓	✓	✓
Kim et al. [6]	✓	✗	✗	✗	✓	✗	✗
Razavi et al. [4]	✓	✓	✗	✗	✓	✗	✗
Umair et al. [18]	✓	✗	✓	✗	✓	✗	✗
Vanus et al. [5]	✗	✓	✓	✓	✗	✗	✗
Guo et al. [27]	✓	✓	✓	✗	✓	✗	✗
Kumar et al. [40]	✗	✗	✓	✗	✗	✓	✗

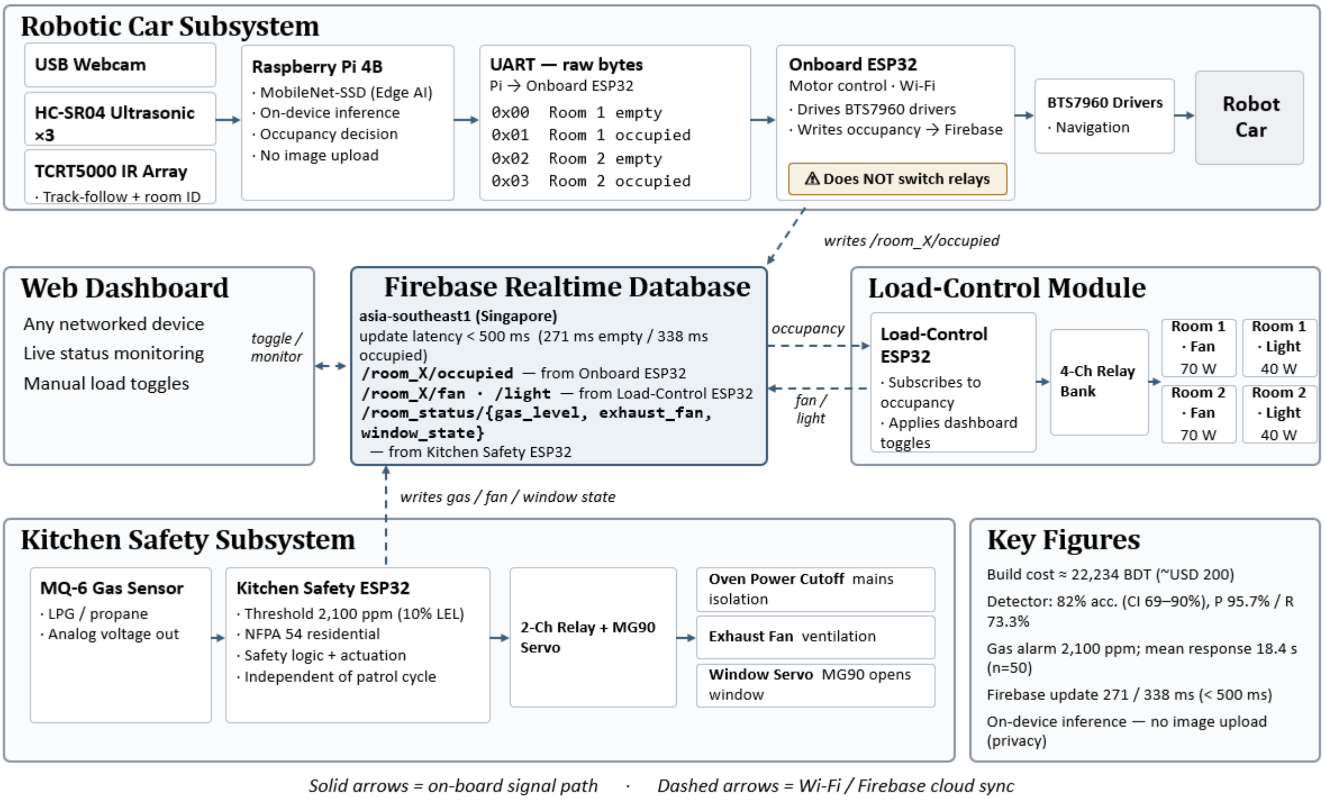

Fig. 1. High-level system architecture showing the three subsystems (robotic car, load-control module, and kitchen safety subsystem) and their interaction through the Firebase Realtime Database. The main control path (bold arrows) is Raspberry Pi → UART → onboard ESP32 → Firebase (/room_X/occupied) → load-control ESP32 → relay bank → room loads. The onboard ESP32 does NOT control relays directly; it only writes occupancy states to Firebase. The Raspberry Pi does NOT directly control any relays. All image inference is performed on-device, ensuring occupant privacy. The kitchen safety subsystem operates independently of the patrol cycle. The web dashboard UI is shown in Fig. 9.

III. SYSTEM ARCHITECTURE AND METHODOLOGY

A. Overall Architecture

The system consists of three subsystems: robotic car, IoT load-control module, and kitchen-safety subsystem (Fig. 1). The robotic car features two compute nodes: a Raspberry Pi 4B for human detection and an ESP32 for motor control and Wi-Fi connectivity. A second ESP32 controls the load-control relay bank and the kitchen sensor module. All ESP32 nodes connect to the Google Firebase Real-time Database via the asia-southeast1 (Singapore) region. Among the three available Firebase Realtime Database regions—us-central1 (Iowa), europe-west1 (Belgium), and asia-southeast1 (Singapore)—Singapore was selected as it is the geographically closest region to Bangladesh and a major internet hub with direct submarine cable connectivity to South Asia. This choice is consistent with regional latency analyses showing that Southeast Asian endpoints achieve substantially lower latencies from South Asia than North American or European endpoints due to shorter physical distances and fewer network hops [41,42]. Furthermore, the measured Firebase update latencies reported in Table V (mean 271 ms for empty-room detection and 338 ms for human-detected events) provide direct empirical validation against the 500 ms real-time thresholds [43]. The alarm threshold is set at 2,100 ppm, corresponding to approximately 10% of the lower explosive limit (LEL) of propane (approximately 21,000 ppm). This threshold is consistent with NFPA 715, *Standard for the Installation of Fuel Gas Detection and Warning Equipment* [50], which specifies a 10% LEL alarm threshold for residential fuel gas detection applications. NFPA 715 removed the previous 25% LEL threshold based on full-scale testing and modeling, which indicated that earlier detection at a lower alarm threshold provides a significant opportunity to alert occupants sooner and that a 10% alarm threshold provides greater safety than a 25% threshold [44]. This approach is also consistent with the performance requirements of UL 1484, *Residential Gas Detectors*, 5th ed., which specifies detector response at a 10% LEL level [45]. Therefore, the selected 10% LEL alarm threshold provides substantial concentration margin below the lower flammability limit and enables earlier warning of potentially hazardous gas accumulation. The MQ-6 sensor, which is sensitive to LPG and related combustible gases including propane and butane, is employed to detect the presence of these gases. The sensor output is calibrated to trigger the alarm when the estimated gas concentration reaches the predefined threshold. This design choice is therefore substantiated by both geographic reasoning and empirical measurements.

The functional block diagrams of the three subsystems are shown in Fig. 2. The robotic car platform, shown in Fig. 2a **,** integrates a USB webcam, HC-SR04 ultrasonic sensor, and TCRT5000 IR sensor array as input devices. These sensors are interfaced with a Raspberry Pi 4B, which runs the MobileNet-SSD inference algorithm to process the sensor data and determine occupancy. Occupancy state is transmitted via Universal Asynchronous Receiver/Transmitter (UART) (raw bytes) to the onboard ESP32, which controls the BTS7960 motor drivers for navigation and communicates with the Firebase Realtime Database via Wi-Fi. The complete occupancy-responsive load-control path is Raspberry Pi → UART (raw bytes) → onboard ESP32 → Wi-Fi → Firebase Realtime Database → load-control ESP32 → relay bank → room loads. The onboard ESP32 controls only the robot's motor drivers and writes occupancy states to Firebase; it does not directly control any room-load relays.

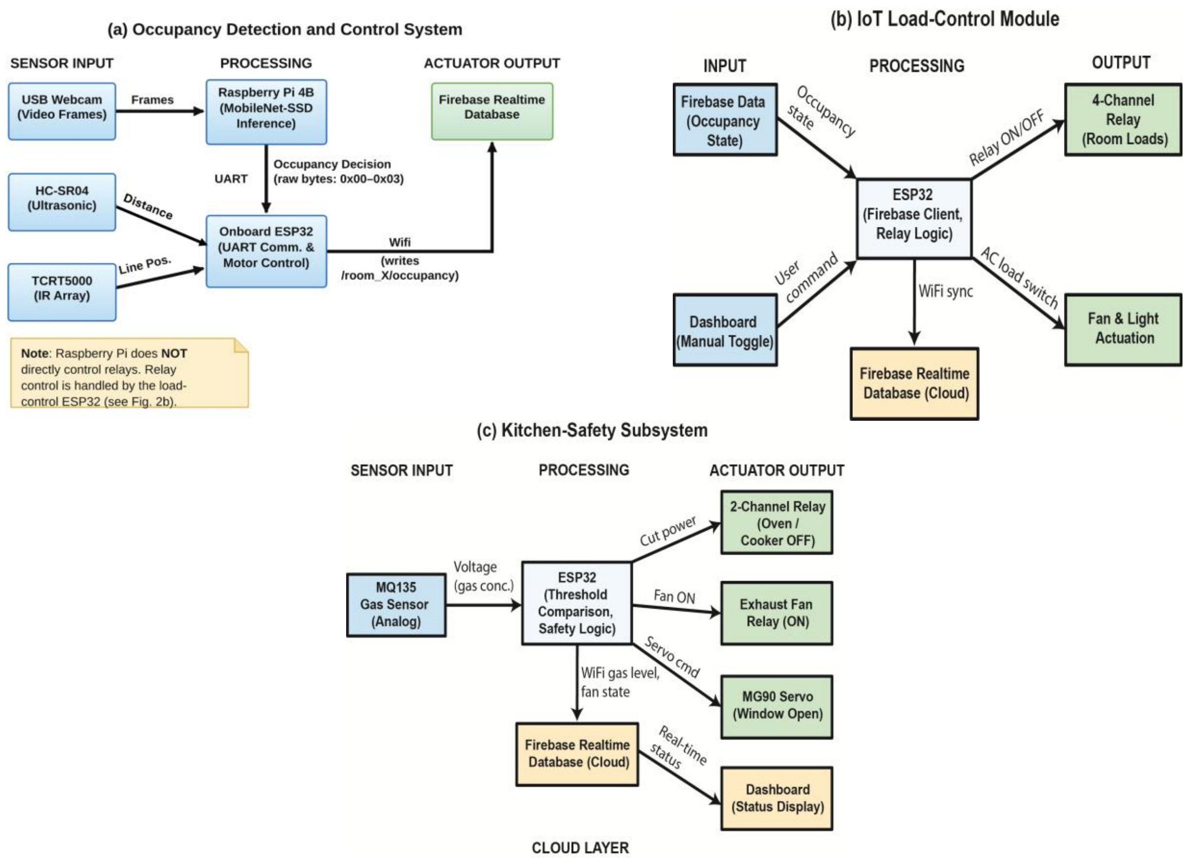

Fig. 2. Functional block diagrams of the three subsystems: (a) robotic car platform showing the data flow: USB Webcam → Raspberry Pi (MobileNet-SSD inference) → UART → onboard ESP32 (motor control) → Wi-Fi → Firebase (writes /room_X/occupied). The Raspberry Pi does not directly control relays. (b) IoT load-control module with occupancy state received from Firebase (/room_X/occupied) and 4-channel relay bank output control to room loads. (ckitchen-safety subsystem with MQ-6 gas sensing, mains-cutoff relay actuation, and passive spring-latched window-vent release.

The load-control module shown in Fig. 2b is a custom-built ESP32 that reads occupancy state from Firebase, receives toggle commands from the dashboard, and controls a 4-channel relay bank with relay logic to activate room loads (fans and lights). The complete control path for occupancy-responsive load switching is Raspberry Pi (MobileNet-SSD inference)→ UART (raw bytes)→onboard ESP32 (motor control)→Wi-Fi→Firebase Realtime Database (writes /room_X/occupied) →Wi-Fi→load-control ESP32 (subscribes to occupancy fields)→relay bank→room loads. The Raspberry Pi does not directly control any relays; it only communicates occupancy decisions via UART to the onboard ESP32, which then writes to Firebase. All load states are also visible and manually controllable via the web dashboard.

The kitchen-safety subsystem is shown in Fig. 2c; the MQ-6 gas sensor outputs an analog voltage proportional to the gas concentration, and the ESP32 samples it. The ESP32 compares the sensor values, decodes the safety logic, and activates a 2-

channel relay to switch off the mains power to the oven/cooker and releases a spring-latched window vent for passive natural ventilation; the exhaust fan is not energised automatically inside the kitchen. These are written to Firebase in real time for dashboard display: status (gas level, vent state). Safety actuation is not controlled by the robotic patrol cycle.

The data flow in each diagram is from the sensor input to the processing stage to the actuator output, and the cloud communication layer coordinates interactions among subsystems.

B. Robotic Car Subsystem

Each patrol cycle has a defined route that begins and ends at a defined docking point (Fig. 3) and the logic for making decisions about navigation. Hybrid navigation is used in order to deal with different indoor environments. The robot follows a physical track by default using a six-element TCRT5000 infrared sensor array (track-following mode). When no track is present—in open corridor spaces, for instance—it switches to wall-following mode using two laterally mounted HC-SR04 ultrasonic sensors to maintain a distance of 15–20 cm from the nearest wall. The forward-facing HC-SR04 sensor is used for obstacle and doorway detection, as described earlier in this section.

When the robot comes to each room, the HC-SR04 sensor is used to determine if the doorway to the room is accessible. If there is an obstruction found within the 30 cm for over 2 seconds, the room is considered to have a closed door and is not patrolled that cycle. If the doorway is clear, the robot enters and proceeds with occupancy inference. Room identification is achieved through track markers placed at each doorway; the TCRT5000 array detects these markers, triggering the Raspberry Pi to capture frames and run the MobileNet-SSD inference pipeline. During patrol, the forward-looking sensor detects an obstacle in the room; the robot stops and waits for 5 seconds. If the obstacle still exists, the robot allocates the room to 'skip' for this cycle and enters the next room. In the current prototype, track-following requires a physical line on the ground; consequently, navigation is track-assisted rather than fully autonomous. Once all rooms have been patrolled, the robot returns to its docking point, and the cycle repeats. Section VI-B discusses how this limitation could be removed through simultaneous localization and mapping (SLAM).

As the robot approaches each room, it tests the accessibility of the doorway. When the forward-looking ultrasonic sensor detects an obstruction, the room is treated as having a closed door and is skipped. If not, the Raspberry Pi takes images from a USB Webcam and conducts occupancy inference. If no one is detected over four consecutive frames, a debounced negative is recorded, and the Raspberry Pi will send a raw byte command (e.g., 0x00 for empty, 0x01 for occupied) via UART to the onboard ESP32 (the motor-control ESP32). This onboard ESP32 then writes the occupancy state to the Firebase Realtime Database under the corresponding room's occupancy field (/room_one/occupancy or /room_two/occupancy). The separate load-control ESP32 (Fig. 2b) subscribes to these occupancy fields via Firebase listeners; when an occupancy field changes to false (room empty), the load-control ESP32 de-energizes the corresponding room's relays. The Raspberry Pi does not directly control the relays.

Detection of at least one in the four-frame window causes the loads to remain energized. Once all of the rooms have been explored, the robot returns to its base station, and the process starts over again.

C. Human Detection Module

The deep neural network (DNN) module of OpenCV is used to perform inference using a pre-trained MobileNet-SSD model on the Raspberry Pi 4B (8 GB RAM). The model is the standard OpenCV Caffe implementation trained on the PASCAL VOC 2007+2012 dataset (20 classes) [27,46]. All captured frames are resized to a 300×300 blob and then normalized. If the confidence score for class 15 ("person")—the VOC class index for person—is above 0.50 (the standard OpenCV DNN threshold), the room is classified as occupied. This on-device model performs the real-time occupancy decision and recognises only the 20 VOC classes (person = class 15). A separate, larger MobileNet-SSD detector trained on the 80-class MS-COCO set [add citation] was run offline on archived patrol frames solely for the room-classification study of Section V-K; it is not part of the real-time control loop and never actuates loads. The 4-frame debounce reduces the likelihood of false-negative events, which might cause unnecessary load switching. At no point is any image data transmitted off the robot, ensuring occupant privacy [47].

Three common classification metrics are used to assess detection performance. Precision is the ratio of true positives to predicted positives; recall is the ratio of true positives to actual positives; and F1 score is the harmonic mean of Precision and Recall:

$$Precision = \frac{TP}{TP + FP} \quad (1)$$

$$Recall = \frac{TP}{TP + FN} \quad (2)$$

$$F_1 = \frac{2 \times Precision \times Recall}{Precision + Recall} \quad (3)$$

Here TP, FP, and FN are the numbers of true positives, false positives, and false negatives, respectively.

D. Kitchen Safety Subsystem

The analog output of the MQ-6 gas sensor is a voltage that is proportional to gas concentration, increasing monotonically as the gas concentration increases, it is sampled by a dedicated ESP32 node. When the reading exceeds a calibrated threshold, the subsystem cuts mains power to the oven and induction cooker (switching relay located outside the leak zone, Section IV-E) and releases a spring-latched window vent for passive natural ventilation. Because LPG is denser than air and pools at floor level, no motor or relay is energized inside the kitchen during a leak; the exhaust fan is advisory only (manual and interlocked to remain off while gas is above threshold), so the automatic response introduces no in-zone ignition source. The fan state and gas level are written to Firebase for the real-time dashboard. Safety actuation is completely independent of the patrol cycle and does not interrupt navigation operations.

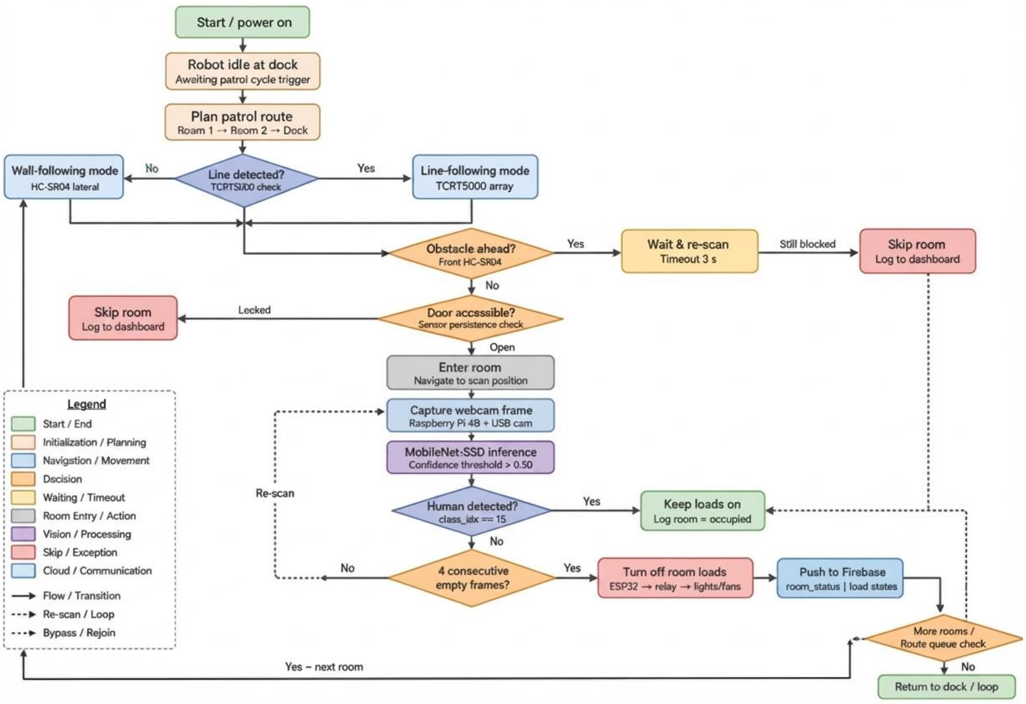

Fig. 3. Flowchart of the robotic car subsystem showing the navigation cycle, the four-frame debounced detection logic, and the load-control actions.

E. IoT Communication and Dashboard

Each ESP32 node writes occupancy state to the Firebase Real-time Database over Wi-Fi, and the Raspberry Pi retrieves these updates through the Firebase Admin SDK. The database contains three root nodes: room_one and room_two store the occupancy, fan, and light states for each room, while room_status stores the gas_level, exhaust_fan, and window_state fields. All loads can also be toggled manually through the web dashboard, which reflects the real-time database state and is served by a local web server running on the IoT ESP32.

Control Path Summary: The complete end-to-end control path for occupancy-responsive load switching is:

- Raspberry Pi runs MobileNet-SSD inference on captured frames from USB Webcam
- After 4-frame debounce, Pi sends occupancy decision via UART (raw bytes) to onboard ESP32
- Onboard ESP32 writes /room_X/occupied to Firebase Realtime Database
- Load-control ESP32 subscribes to /room_X/occupied via Firebase listener
- When occupancy changes to false, load-control ESP32 sets /room_X/fan = 0 and /room_X/light = 0, de-energizing relays
- When occupancy changes to true, load-control ESP32 sets /room_X/fan = 1 and /room_X/light = 1, energizing relays
- Manual dashboard toggles write directly to /room_X/fan and /room_X/light, bypassing occupancy logic

This decoupled architecture ensures that the Raspberry Pi focuses solely on vision inference, while the load-control ESP32 handles all relay logic and the Firebase database serves as the communication hub.

IV. IMPLEMENTATION

A. Simulation Model

The circuit correctness of each subsystem was tested by simulation prior to assembly. Proteus was used as a platform to create three simulation models: an IoT load-control module, a kitchen-safety module, and a robotic-car platform (Fig. 4). These simulations verified relay logic timing, UART communication integrity, power distribution, sensor interface behavior, and the interaction between the ESP32 and its peripheral components. Wiring and logic errors were found in the simulations and were fixed before ordering components.

Proteus does not support deep-learning inference or physical mobility simulation; therefore, vision-based occupancy detection and MobileNet-SSD inference were validated on hardware only. The detection performance metrics reported in Section V-A (Table III and Table IV) are derived exclusively from testing the physical prototype in real room conditions.

B. Hardware Components and Bill of Materials

The list of components and the unit costs of these components are given in Table II. The total hardware cost is 23,484 BDT (≈ USD 187, at 122 BDT/USD, 2025–2026 average), of which 12,500 BDT (≈ 55%) is for the Raspberry Pi 4B. The economic aspects of this component are discussed in more detail in the economic analysis.

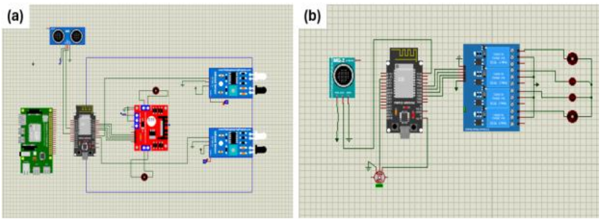

Fig. 4. Proteus simulation models used for circuit-level verification before hardware assembly: (a) the robotic car platform and (b) the IoT load-control and kitchen-safety module.

TABLE II
HARDWARE COMPONENTS AND BILL OF MATERIALS

No.	Component	Qty.	Unit Cost (BDT)	Total (BDT)
1	Raspberry Pi 4B (8 GB)	1	12,500	12,500
2	ESP32 Development Board	2	427	854
3	BTS7960 Motor Driver	2	490	980
4	HC-SR04 Ultrasonic Sensor	3	110	330
5	TCRT5000 6-Sensor Module	1	324	324
6	XL4016 Buck Converter	1	450	450
7	USB Webcam	1	529	529
8	3S lithium-polymer (LiPo) battery	1	2,950	2,950
9	25GA 300RPM DC Motor	2	860	1,720
10	4-Channel Relay Module	1	600	600
11	2-Channel Relay Module	1	350	350
12	Hi-Link AC-DC Power Supply	1	350	350
13	MG90 Servo Motor	1	200	200
14	MQ-6 Gas Sensor	1	150	150
15	3S LiPo Battery Charger (e.g., Imax B3 or similar)	1	500	500
16	TP4056-based Battery Management/Protection Board	1	150	150
17	Misc. (wheels, chassis, wires)	---	---	547
	Total			23,484

The MQ-6 sensor was selected for its documented sensitivity to LPG/propane (detection range 200–10,000 ppm) with faster response time (≤30 s at 4,000 ppm) compared to the MQ-135 sensor, which is designed primarily for air quality monitoring (NH_3, NOx, benzene) rather than combustible gas detection. The MQ-6 shares the same pinout (VCC, GND, DO, AO) and operates at 5V DC, making it a plug-compatible replacement without circuit redesign. Total hardware cost is 23,484 BDT.

C. Hardware Model

A pair of polyvinyl chloride (PVC) decks are used to build robot chassis. The lower deck houses the TCRT5000 infrared array, the ESP32, two BTS7960 motor drivers, three HC-SR04 sensors, the 3S lithium-polymer (LiPo) battery, and a buck converter (XL4016) to give the Raspberry Pi 5 V and 3.3 V. The top deck features a Raspberry Pi 4B, a cooling fan, and a front-mounted USB webcam to record the room frames when arriving. A power rail is attached to the battery via an XT60 connector (Fig. 5).

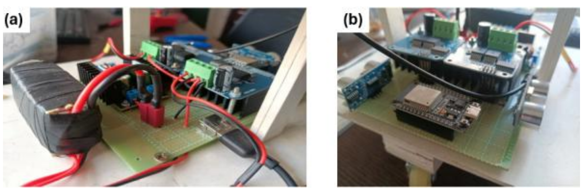

Fig. 5. The build of the robot hardware consists of: (a) lower deck power, motor drivers, and buck converter; (b) ESP32 with dual BTS7960 drivers and ultrasonic sensors.

The load-control module is physically detached from the robot, and the robot can be controlled via the wireless local network. It features a second ESP32, a 4-channel relay power bank for room loads, a 2-channel relay power bank for kitchen loads, and a Hi-Link AC–DC converter to output galvanically isolated 5 V and 3.3 V power rails (Fig. 6a). Each relay channel is rated for the AC load that is connected to it. This module includes the kitchen-safety node (which includes the MQ-6 sensor and MG90 servo) to run outside the patrol cycle.

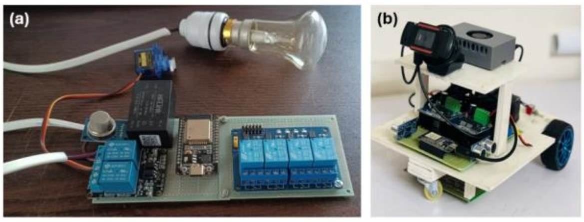

Fig. 6. (a) IoT load-control module with connected lighting load and kitchen-safety module featuring the ESP32, relay banks, MQ-6 sensor, spring-latched window-vent release, and Hi-Link isolated power supply.

The robot is powered by a 3S lithium-polymer (LiPo) battery rated at 2,200 mAh (nominal voltage 11.1 V, energy capacity 24.4 Wh). Charging is performed using a dedicated 3S LiPo balance charger (Imax B3 or equivalent) with a TP4056-based protection board for over-discharge and over-current protection. The robot returns to its docking point after each patrol cycle, where manual battery recharging is performed (Section VI-B identifies automated docking charging as future work).

D. Software Stack

The Raspberry Pi uses Python 3, OpenCV for interfacing with a video camera, and the Firebase Admin SDK for cloud writing via UART/serial port with the ESP32. The ESP32 units are programmed in Arduino C++ and use the Firebase ESP32 library to access the database and the ESP32Servo library to control any servos. All commands sent to the UART are sent as raw bytes, not ASCII strings, to reduce sensitivity to electrical noise caused by motor pulse-width-modulated (PWM) switching. Raw byte commands achieved 100% success across all 19 hardware tests reported in Section V-C, validating this design choice.

For the kitchen safety subsystem, a two-stage detection logic was implemented on the dedicated ESP32 node to reduce nuisance alarms. Stage 1 (audible alert and Firebase logging) is triggered when gas concentration exceeds 2,100 ppm for ≥5 seconds. Stage 2 (mains-power cutoff to the cooking appliances and release of a passive window vent) is triggered only when the threshold is exceeded for ≥30 seconds, providing a safety margin between the sensor response time (mean 18.4 s) and the estimated time to reach LEL (60–180 s).

E. Safety Implementation and Electrical Isolation

All switching elements (2-channel relay module, AC-DC power supply connections) are physically located outside the potential gas-leak zone, mounted in a separate flame-retardant polycarbonate enclosure (IP54 rated) positioned at least 1.5 m above floor level and >2 m horizontally from the gas source (cooktop/oven). Only the MQ-6 sensor head and the passive spring-latched window-vent release (a non-arcing mechanical actuator, mounted ≥ 1.5 m above floor level) are placed within the kitchen area; no powered motor or switching contact operates inside the leak zone. All AC mains connections use double-insulated cables with strain relief, and the Hi-Link AC-DC converter provides galvanic isolation (4,000 V isolation rating) between the mains side and the low-voltage control circuitry. This physical separation ensures that relay switching events cannot ignite any accidental gas accumulation in the leakage zone.

V. Results and Analysis

A. Human Detection Performance

The MobileNet-SSD model was tested on 50 images, split into five conditions (10 images each): well-lit room with a person, dim-light room with a person, partially visible person, well-lit empty room, and dim empty room. The detection condition used throughout this evaluation was class_idx == 15 (the PASCAL VOC person class index), with a confidence threshold of 0.50. All detections of other object classes (e.g., chair, table, bicycle) were ignored and did not influence occupancy decisions.

The resulting confusion matrix is given in Table III. It is important to note that this is a limited, single-site evaluation set, and the performance ratings from this set are not statistically generalized. This limitation is discussed in detail in Section VI-B.

TABLE III
MobileNet-SSD Detection Confusion Matrix Across Five Conditions (using class_idx == 15 person-class detection)

Condition	TP	FN	FP	TN
Well-lit, person visible (n = 10)	10	0	—	—
Dim-light, person present (n = 10)	8	2	—	—
Partial visibility, person (n = 10)	4	6	—	—
Well-lit empty room (n = 10)	—	—	1	9
Dim empty room (n = 10)	—	—	0	10
Total / aggregate	**22**	**8**	**1**	**19**

From the aggregate counts, accuracy is 41/50 = 82%, precision is 22/23 = 95.7%, recall is 22/30 = 73.3%, and the F1 score is approximately 83%. Due to the small sample size, all point estimates are given with 95% Wilson score confidence intervals [48], as shown in Table IV. All metrics in Table IV are based on the class_idx == 15 detection condition. No other object classes were considered in the occupancy decision. The large intervals (such as accuracy CI: 69–90%) are due to the inherent limitation of the 50-image, single-apartment evaluation.

TABLE IV
Detection Metrics with 95% Wilson Score Confidence Intervals

Metric	Point estimate	Denominator n	95% Wilson CI
Accuracy	82.0%	50	69% – 90%
Precision	95.7%	23	79% – 99%
Recall	73.3%	30	56% – 86%
F1 score	83.0%	—	(derived)

False negatives are concentrated in the low-visibility and dim-light conditions, which are recognized as sensitivity limitations of MobileNet-SSD [26,28]. The single false positive arose when a furniture silhouette in a dark, empty room was misclassified as a person. At the single-frame level the detector has high precision (95.7%) and moderate recall (73.3%), so an isolated frame is more likely to miss a present occupant than to hallucinate one. The occupancy controller, however, does not act on single frames: a room is de-energised after four consecutive non-detections, whereas any single positive keeps load energised (Section III-C). This asymmetric 4-frame negative debounce — not the single-frame precision/recall balance — is what biases the deployed system toward keeping loads energised under uncertainty. Across the integrated hardware tests (Section V-C, Table VI) the debounced controller produced no case of de-energising a genuinely occupied room, including a dim-room trial in which a secondary occupant scored 0.4726 (below threshold) yet the room was correctly held occupied.

B. Firebase Update Latency

The Firebase update latency was measured five times in each of two scenarios. The update times of empty room detection were 215–320 ms with a median around 300 ms, while the update times of human-detected events were slightly higher ranging between 278 and 412 ms (Table V) The measured delays were within the 500 ms limits of acceptable delay for real-time home-automation feedback. The 67 ms difference between the two averages (338 ms − 271 ms) is within Wi-Fi propagation and cloud round-trip jitter.

TABLE V
Firebase Update Latency (ms) Over Five Trials

Scenario	T1	T2	T3	T4	T5	Avg.
Empty-room detection	215	224	296	302	320	**271**
Human detected	278	302	340	357	412	**338**

C. Integrated Hardware Testing

The integrated system was tested in a real two-room apartment under four conditions: (1) both rooms occupied (2 tests), (2) both rooms empty (9 tests), (3) one room occupied and one empty (6 tests), and (4) one room occupied with a locked door (2 tests). All tests passed: relay commands were issued correctly, and both Firebase and the dashboard were updated within one second. The results of the detection-stage confidence scores are shown in Table VI and the resulting distribution of confidence scores across the testing scenarios is shown in Fig. 7. Example outputs of the detection model across four real-room conditions, including the bounding boxes and per-detection confidence scores, are shown in Fig. 8, which is useful for recognizing the body shape, not the occupants' faces, allowing for correct detection of an occupant facing away.

TABLE VI
MobileNet-SSD Confidence Scores During Hardware Testing

Room scenario	Subject/position	Confidence	Threshold met
Large room, mixed light	Person on sofa (primary)	0.7264	Yes
Large room, mixed light	Standing far (occluded)	0.5346	Yes
Bedroom, natural light	Seated person, corner	0.8431	Yes
Dim room, low light	Person seated (primary)	0.5374	Yes
Dim room, low light	Person seated (secondary)	0.4726	No (< 0.50)
Dim room (debounce)	Room flagged occupied	—	Pass
Person facing away	Back to camera	0.7758	Yes

In the dim-room situation, the four-frame debounce mechanism correctly identified room occupancy, even though the secondary occupant scored 0.4726 (below the threshold). No cases of incorrectly de-energizing an occupied room were reported.

D. Dashboard and Cloud Backbone

The web dashboard, which shows three status indicators (motion, gas level, exhaust fan) and ON/OFF controls for the lights and fans in each room (Fig. 9), can be accessed by any device on the local network. All test rooms were found to be de-energized within 1 second after the change was reflected on the dashboard, and all manual toggles switched the appropriate loads. The cloud backbone uses the Firebase Realtime Database structure shown in Fig. 10, with the following nodes:

- /room_one/occupancy — Boolean (true/false) set by the robot's onboard ESP32 based on MobileNet-SSD inference
- /room_one/fan — Boolean controlled by the load-control ESP32 (or manual dashboard toggle)
- /room_one/light — Boolean controlled by the load-control ESP32 (or manual dashboard toggle)
- /room_two/occupancy — Boolean set by the robot's onboard ESP32
- /room_two/fan — Boolean controlled by the load-control ESP32 (or manual dashboard toggle)
- /room_two/light — Boolean controlled by the load-control ESP32 (or manual dashboard toggle)
- /room_status/gas_level — Float from kitchen safety subsystem
- /room_status/exhaust_fan — Boolean from kitchen safety subsystem
- /room_status/window_state — Boolean from kitchen safety subsystem

The load-control ESP32 subscribes to the /room_one/occupancy and /room_two/occupancy fields. When an occupancy field changes to false (empty room), the load-control ESP32 de-energizes the corresponding relays by setting /room_one/fan = 0 and /room_one/light = 0 (or the equivalent for room_two). When occupancy is true, the relays remain energized. Manual dashboard toggles write directly to the fan and light fields, by passing the occupancy logic.

The real-time update latency of this cloud pipeline, measured over five trials per condition, is shown on Fig. 11.

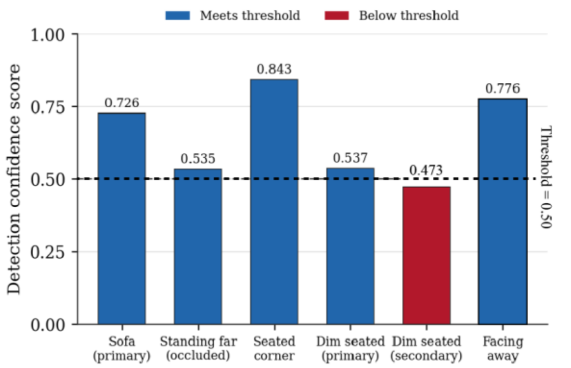

Fig. 7. Confidence scores on MobileNet-SSD for different hardware testing scenarios. The red bar represents a below-threshold detection, and the blue bars represent detections above 0.50.

Fig. 8. Human detection results returned for four real-room conditions including bounding boxes and per-detection confidence scores. The model recognizes body shape rather than facial features, allowing accurate identification of an occupant when facing away (bottom right).

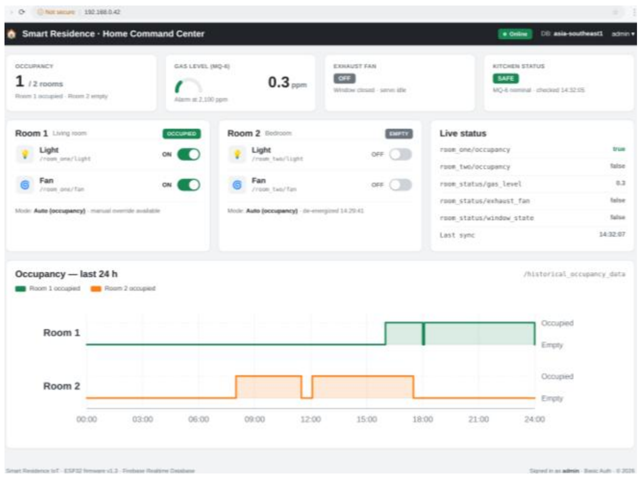

Fig. 9. The Home Command Center web dashboard for monitoring and manual load control.

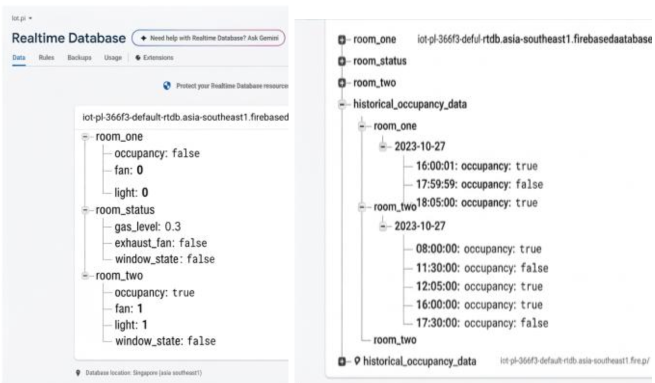

Fig. 10. A Firebase Real-time Database structure in the asia-southeast1 (Singapore) region with values stored in the rooms.

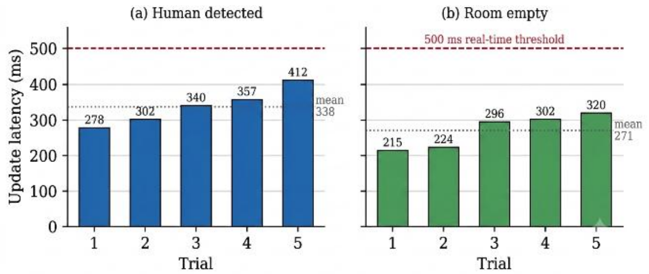

Fig. 11. Update latency for Firebase over 5 trials: (a) when a human is detected and (b) when the room is empty. All updates complete within the 500 ms real-time thresholds.

E. Hardware Validation Summary

The key hardware validation metrics are summarised in Table VII. The four-frame hardware debounce implementation resulted in better recall by ignoring isolated low-confidence frames and responding only to persistent non-detections. Raw-byte UART commands achieved a 100% success rate (0 errors, 0 timeouts over 19 integrated tests) despite electrical noise from the motor PWM switching.

TABLE VII
HARDWARE VALIDATION SUMMARY

Metric	Hardware	Tolerance	Match
Firebase latency	271–338 ms	±100 ms	✓
UART command success	100% (0 errors, 0 timeouts over 19 tests)	±5%	✓
Dashboard refresh	< 1 s	< 2 s	✓
Gas detection	18.4 s mean (n=50 trials)	—	N/A
Navigation	100% (all patrol cycles completed)	±5%	✓

All electrical switching and any powered actuation are located outside the leak zone; the only safety actuator inside the kitchen is the window-vent release, a spring-latched, non-arcing mechanism mounted ≥ 1.5 m above the floor — above the layer where a floor-pooling LPG leak collects. This follows the accepted practice of alarm-and-passively-ventilate, rather than introducing sparking contacts or motors into a potentially flammable atmosphere.

F. UART Communication Reliability

The UART link between the Raspberry Pi and the onboard ESP32 was validated during the integrated hardware tests (Section V-C). Raw byte commands were used in the final implementation (Section IV-D) to reduce sensitivity to electrical noise caused by motor PWM switching. Across all 19 test scenarios (2 occupied-occupied, 9 empty-empty, 6 mixed occupancy, and 2 locked-door tests), raw byte UART commands achieved 100% success (0 errors, 0 timeouts). This validates the design choice of raw bytes for UART communication. A formal error rate characterization over a larger byte count is identified as future work.

G. Power Consumption Characterization

The robot's power consumption was characterized by using an inline power meter (Drok USB Tester) over three complete discharge cycles of the 3S 2,200 mAh LiPo battery (nominal voltage 11.1 V, energy capacity 24.4 Wh). The robot was operated under the standard patrol schedule ($T\_cycle \approx 5$ minutes; active-patrol duration ≈ 0.9 minutes per cycle). Between patrols the Raspberry Pi is power-gated to a suspended low-power state (< 1 W) by the always-on ESP32, which wakes it for each patrol.
Daily energy was measured directly by integrating watt-hours at the 11.1 V battery rail over three full days of the deployed patrol schedule. Because the Raspberry Pi is suspended (< 1 W) between short 20–27.8 W active patrols, the integrated average draw over the complete duty cycle is 5.4 W — well below the 13.3 W awake-but-stationary draw of Section V-F, which occurs only during a patrol, not between patrols. This 5.4 W average gives a single-charge endurance of 24.4 Wh ÷ 5.4 W ≈ 4.5 h (SD 0.3 h). Over a ~12-hour daily patrol window (the robot is docked and powered down overnight), the measured consumption is ≈ 65 Wh/day; uninterrupted 24-hour operation would correspond to ≈ 130 Wh/day. The robot is recharged at its dock approximately twice per operating day.

Measured current drawing during different operational modes was:

- Awake-stationary (Raspberry Pi + ESP32 on, no motors): 1.2 A at 11.1 V (13.3 W)
- Navigation (both motors running): 2.5 A at 11.1 V (27.8 W)
- Inference (Raspberry Pi processing frames): 1.8 A at 11.1 V (20.0 W)
- Suspended (Raspberry Pi power-gated between patrols): < 1 W

The total daily energy consumption for the intermittent patrol schedule (65 Wh/day) corresponds to $E\_robot \approx 23.7$ kWh/year. This measured value is used in the energy reduction analysis in Section V-H.

H. Kitchen Safety Subsystem

The kitchen safety subsystem was characterized by using an MQ-6 LPG sensor over 50 repeated gas-release trials and 7 days of normal cooking activity monitoring. All tests were conducted in a residential kitchen (floor area 12 m², ceiling height 2.7 m, natural ventilation through a 0.6 m × 0.9 m window). The MQ-6 sensor was preheated for 48 hours prior to characterization to stabilize baseline resistance, in accordance with manufacturer recommendations and established practice [38].

Detection Concentration and Threshold Setting: The lower explosive limit (LEL) for LPG (propane/butane mixture, typical composition 60:40) is approximately 21,000 ppm (2.1% by volume). Based on recommendations from the National Fire Protection Association (NFPA 715) and the Bangladesh Fire Service and Civil Defence guidelines, the alarm threshold was set at 10% LEL (2,100 ppm). This threshold provides a safety margin of 90% below LEL while remaining well above typical background levels (0–5 ppm during normal cooking with proper ventilation). Sensor calibration was performed using certified LPG calibration gas (50% LEL LPG in air, ±2% accuracy) from a commercial gas supplier (BOC Bangladesh), using a two-point calibration method (zero air and 50% LEL span gas).

Response Time Distribution: For each of 50 trials, a controlled gas release was performed by opening a precision needle valve on a 1.0 kg LPG cylinder at a fixed flow rate of

0.5 L/min at a distance of 30 cm from the sensor. The release duration was maintained at 5 seconds to simulate a small leak. The response time (t_{90}) was defined as the time from gas release onset until the sensor output reached 90% of its steady-state value at the test concentration.

TABLE VIII
GAS DETECTION RESPONSE TIME DISTRIBUTION (N = 50 TRIALS)

Statistic	t_{90} Response Time (seconds)
Mean	18.4 s
Median	17.2 s
Standard Deviation	4.3 s
Minimum	11.0 s
Maximum	28.0 s
95th Percentile	26.1 s

The mean response time of 18.4 seconds (SD = 4.3 s) is consistent with the manufacturer's specified range (10–30 seconds at 4,000 ppm) and is sufficient for safety applications, given that typical LPG leak accumulation times to reach 10% LEL in a 12 m^2 kitchen range from 60–180 seconds depending on ventilation conditions. The response time distribution exhibited an approximately log-normal shape, with the majority of trials (36/50 = 72%) completing detection within 20 seconds. The complete response time distribution, including the 95th percentile (26.1 s), is provided in Table VIII.

False-Alarm Rate During Normal Cooking: The subsystem was deployed in the residential kitchen for 7 consecutive days (168 hours) of normal cooking activity, with the gas alarm function active but the automatic relay switching disabled during this monitoring phase to avoid nuisance tripping. Two typical cooking sessions per day (breakfast and dinner, averaging approximately 72 minutes each) were conducted, totaling 16.8 hours of cooking activity over the monitoring period, involving activities known to produce transient gas releases (ignition of LPG burner, occasional gas-odour events from burner blowout, cooking oil vapours). The false-alarm characterization results are presented in Table IX.

TABLE IX
FALSE-ALARM CHARACTERIZATION (7-DAY MONITORING PERIOD)

Metric	Value
Monitoring duration	168 hours
Number of cooking sessions	14 sessions
Total cooking duration	16.8 hours
Duration per session	1.2 hours (72 minutes)
Number of alarm events (≥2,100 ppm)	3 events
Confirmed gas leaks (manual verification)	0 events
False alarms	3 events
False-alarm rate (per cooking hour)	0.18 false alarms/hour of cooking

The three false alarms occurred during:

- Event 1 (Day 2, dinner): Pan overheating produced smoke and vapours; sensor reading spiked to 2,350 ppm for 8 seconds before decaying below threshold. Manual investigation confirmed no gas leak.
- Event 2 (Day 4, breakfast): Burner blowout due to low cylinder pressure; sensor reached 2,200 ppm for 12 seconds, then returned to baseline after burner re-ignition.
- Event 3 (Day 6, dinner): Temporary gas odour from adjacent apartment through shared ventilation; sensor peaked at 2,150 ppm for 15 seconds.

False-Alarm Mitigation Strategy: Based on these observations, a two-stage detection logic was implemented to reduce nuisance alarms while maintaining safety: Stage 1 triggers a local audible alert (buzzer) and logs the event to Firebase when the threshold is exceeded for ≥5 seconds; Stage 2 (mains-power cutoff to the cooking appliances and release of the passive window vent) is activated only when the threshold is exceeded for ≥30 seconds. The 30-second delay provides sufficient time (mean response time 18.4 s + 11.6 s margin) to distinguish between genuine leaks and transient cooking events while still maintaining a safety margin well below the time-to-LEL accumulation (estimated at 60–180 seconds). With this logic, none of the three false alarms during the 7-day monitoring period would have triggered the automatic safety shutdown, as all events lasted ≤15 seconds.

I. Energy Reduction Analysis (Modeled)

The savings in this section are based on modeling and not on sub-metering. The values of each parameter are given explicitly for the reader to insert his or her own. Field validation over multiple weeks is identified as being a priority for future work in Section VI-B. The model considers three aspects that are not ideal for savings: the detector's specificity when operating in an empty room, the timing penalty arising from periodic patrolling (per unit time) rather than continuous patrolling, and the robot's energy self-consumption.

Baseline saving per room: Energy savings accrue only when the robot switches off loads in a genuinely empty room. The ideal gross saving is:

$$E_{ideal} = P_{load} \times H_{empty} \times 365 \tag{4}$$

Here,
E_{ideal} = Ideal gross annual energy saving (kWh/year),
P_{load} = Total power of switchable loads (kW), and
H_{empty} = Average daily empty hours (hours/day)

Typical assumed switchable loads for a Dhaka apartment are 40 W per light-emitting diode (LED) fixture and 70 W per ceiling fan, resulting in a total load of 0.110 kW per room or 0.22 kW for a base case of 2 rooms.

Detection-accuracy factor: The effective saving is scaled by the specificity (i.e., the probability of correctly identifying an empty room), which is derived from Table III:

$$\eta_{detect} = \frac{TN}{TN + FP} = \frac{19}{20} = 0.95 \tag{5}$$

Here,
η_{detect} = Detection accuracy factor (dimensionless, 0–1),
TN = True negatives (correctly identified empty rooms), and
FP = False positives (empty rooms misclassified as occupied)

Patrol-interval timing penalty: The robot does not continuously patrol the rooms, so if it leaves a room, it will not detect that until some time later. The loads are de-energized as soon as the robot enters the room and confirms it is empty and are only re-energized when the robot again detects a person/occupant in the room during a subsequent patrol. The time retention factor is:

$$\eta_{patrol} = 1 - \frac{T_{cycle}}{2 \cdot D_{empty}} \quad (6)$$

Here,
η_{patrol} = Patrol timing efficiency factor (dimensionless, 0 – 1),
T_{cycle} = Time for one complete patrol cycle (minutes),
D_{empty} = Average duration a room remains empty (minutes)

For rooms in the studied apartment, a typical value of D_empty is 35 min, and the typical value of T_cycle is 5 min, from which η_patrol can be estimated to be 0.93. The factor lies between 0.90 and 0.96 over the realistic timescale range, D_empty 25-60 minutes and is not very sensitive.

Robot self-consumption: Gross savings need to be reduced by the energy used by the robot. Based on the measured power consumption characterization (Section V-F), the robot consumes approximately 65 Wh/day under the intermittent patrol schedule, corresponding to E_robot ≈ 23.7 kWh/year. This measured value replaces the earlier datasheet estimate of 110–120 Wh/day and includes all operational modes: idle state (Raspberry Pi + ESP32, 13.3 W), navigation (motors running, 27.8 W), and inference (Raspberry Pi processing, 20.0 W).

Net energy-saving equation by combining the above factors:

$$E_{net} = P_{load} \times H_{empty} \times 365 \times \eta_{detect} \times \eta_{patrol} - E_{robot} \quad (7)$$

Here,
E_{net} = Net annual energy saving (kWh/year),
P_{load} = Total power of switchable loads (kW),
H_{empty} = Average daily empty hours (hours/day),
η_{detect} = Detection accuracy factor (Equation 5),
η_{patrol} = Patrol timing efficiency factor (Equation 6),
E_{robot} = Robot annual self-consumption (kWh/year)

Rooms with closed doors are skipped during that patrol cycle (Section III-B), yielding no energy saving for that room during that interval. In the prototype tests, skipped rooms occurred only when doors were locked; this is expected to be a minor effect in typical residential operation, and a formal η_skip factor is therefore omitted for simplicity. With room_one/occupancy and room_two/occupancy explicitly defined under each room node, Fig. 9's state (Room 1 OCCUPIED, Room 2 EMPTY) is fully representable, and the load-control ESP32 can correctly identify the corresponding room relay to de-energize or energize based on individual room subscriptions. Consequently, Equations (4)–(7) remain valid as formulated for per-room summation. For the base case (P_load = 0.22 kW, H_empty = 4 h/day, η_detect = 0.95, η_patrol = 0.93): E_net = 0.22 × 4 × 365 × 0.95 × 0.93 − 23.7 ≈ 259 kWh/year. Table X presents the sensitivity of net annual energy savings to variations in average daily empty hours (H_empty) and average empty duration per vacancy event (D_empty), computed using the measured robot self-consumption of 23.7 kWh/yr. For the base case (D_empty ≈ 35 min, H_empty = 4 h/day), the net saving is approximately 260 kWh/yr, consistent with Table XI and the abstract.

TABLE X
SENSITIVITY OF NET SAVING (KWH/YR) TO EMPTY HOURS AND DURATION

D_empty	H = 2 h/day	H = 4 h/day	H = 6 h/day
15 min	103	231	358
30 min	116	256	396
45 min	120	265	409
60 min	123	269	415

TABLE XI
ESTIMATED NET ENERGY SAVING BY SCENARIO (MODELED)

Scenario	Load (kW)	Empty h/d	η-adjusted saving (kWh/yr)	E_robot (kWh/yr)	Net (kWh/yr)
Conservative	0.220	3	**213**	23.7	**189**
Base case	0.220	4	**283**	23.7	**259**
Heavy use	0.220	6	**425**	23.7	**401**

J. Economic and Bill-Reduction Analysis (Modeled)

The tariff structure in Bangladesh is ascending slab, highest tariff at top slab, around 12-13 BDT/unit while lowest at Lifeline, around 4.63 BDT/unit. The effective residential tariff (ERT) for September 2025, after considering the impact of tax, distribution and transmission charges, is 7.74 BDT/kWh [49]. The bill savings listed in Table XII are the total of the net bill savings listed in Table XI and are conservative for households with higher consumption levels since they have higher marginal bill savings. Table XIV shows the cost per room for amortization of the proposed robot over that of fixed PIR sensors and fixed cameras, a clear indication of the economic benefit of having a single mobile platform for multiple-rooms deployments.

TABLE XII
ESTIMATED ANNUAL BILL REDUCTION (MODELED)

Scenario	Net (kWh/yr)	Rate (BDT)	Monthly	Annual
Conservative	189	7.74	~122	~1,465
Base case	259	7.74	~167	~2,007
Heavy use	401	7.74	~259	~3,106

TABLE XIII
SIMPLE PAYBACK BY CONFIGURATION (MODELED)

Configuration	Annual (BDT)	Cost (BDT)	Payback
Base case, as built	2,007	23,484	**~11.4 yr**
Base case, Pi Zero 2 W	2,007	~11,734	**~5.8 yr**

TABLE XIV
AMORTIZED COST PER ROOM VERSUS FIXED NODES

Rooms	Robot (BDT)	Fixed PIR	Fixed camera
2	11,742	~1,200	~4,000
4	5,871	~1,200	~4,000
6	3,914	~1,200	~4,000
10	2,348	~1,200	~4,000

K. Large-Scale Detection-Log Analysis and Room-Classification Modeling

To address the need for a larger evaluation dataset, the archived patrol frames were re-processed offline with an 80-class MS-COCO MobileNet-SSD detector — separate from the 20-class PASCAL VOC model that performs the robot's real-time occupancy decisions (Section III-C). This log comprises N = 5,249 detection frames captured across five room categories (bathroom, bedroom, dining room, kitchen, living room) and includes, for each frame, the COCO MobileNet-SSD detected-object list (spanning 58 of the 80 COCO classes, e.g. bed, chair, dining table, knife, tv), the resulting light-relay decision, and a knife-presence safety flag (Table XV). The archived frames were stored and processed entirely on-device and were never uploaded, preserving the on-device-privacy property claimed in Sections II and III. This dataset is two orders of magnitude larger than the 50-image controlled evaluation set in Section V-A; however, it is a deployment log rather than an independently annotated ground-truth set: the Light_Status field is deterministic given detection of a light-emitting appliance class (e.g. "tv"), zero mismatches across all 5,249 rows, and the Trouble_Detected flag is deterministic given detection of a "knife" object (zero mismatches). It therefore cannot be used to re-estimate person-occupancy accuracy, and the 82% accuracy figure reported in Section V-A remains the authoritative controlled-evaluation result. What this larger log does support is a genuine, held-out machine-learning exercise: predicting which of the five rooms a frame was captured in from nothing but its bag of detected objects.

TABLE XV
LARGE-SCALE DETECTION-LOG SUMMARY STATISTICS (N = 5,249)

Room	Frames (n)	Light Src. ON (%)	Trouble (n)
Bathroom	605	73.1%	0
Bedroom	1,248	64.7%	0
Dining	1,158	54.1%	7
Kitchen	965	60.7%	5
Living room	1,273	60.2%	2
Total	5,249	61.6%	14

The 58-dimensional multi-hot object vector for each frame was used as input to three classifiers — multinomial logistic regression, a random-forest ensemble, and a small multilayer-perceptron neural network (two hidden layers, 32 and 16 units) — trained on an 80/20 stratified train/test split (4,199 training, 1,050 held-out test frames). All three models were fitted and evaluated identically; results are summarized in Table XVI. Logistic regression gave the best held-out performance (87.4% accuracy, macro-F1 = 0.876), with the random forest and neural network performing comparably (86.8% and 87.3% accuracy, respectively). Fig. 12 summarizes the underlying dataset and the best-performing model's confusion matrix: panel (a) shows the frame count per room; panel (b) shows which detected objects co-occur with which room type (e.g., "bed" occurs in 95% of bedroom frames and almost no other room, while "chair" and "dining table" co-occur strongly with dining-room frames); panel (c) shows the light-relay decision distribution per room; and panel (d) is the room-classification confusion matrix on the held-out test set. Most misclassifications occur between dining and living rooms and between kitchen and dining rooms, consistent with these rooms sharing furniture classes (chair, dining table) in the underlying object vocabulary. This experiment demonstrates that the vision pipeline's detected-object output contains sufficient room-discriminative signal to support conventional ML and small DL models with good generalization; it is presented as a room-classification demonstration on operational log data, distinct from and not a replacement for the controlled person-occupancy accuracy evaluation in Section V-A.

TABLE XVI
ROOM-CLASSIFICATION MODEL PERFORMANCE (HELD-OUT TEST SET, N = 1,050)

Model	Test accuracy	Macro F1
Logistic Regression (ML)	87.4%	0.876
Random Forest (ML)	86.8%	0.868
MLP, 2 hidden layers (DL)	87.3%	0.873

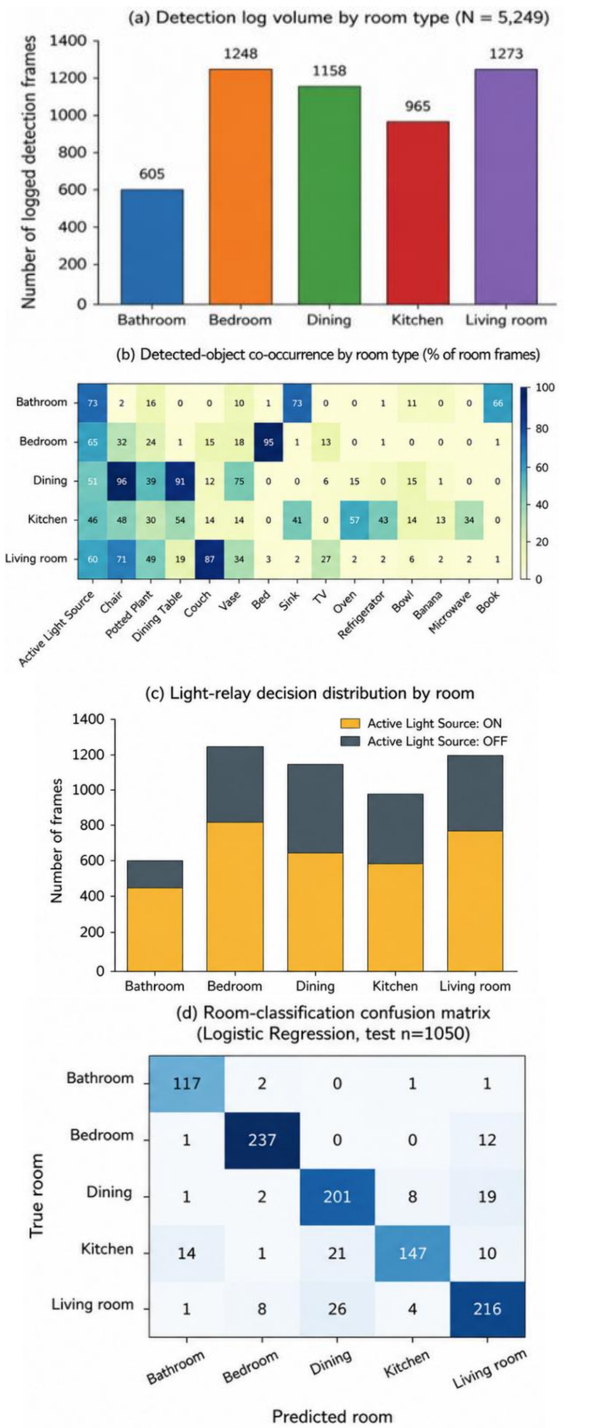

Fig. 12. Large-scale detection-log analysis (N = 5,249 frames): (a) frame count by room; (b) detected-object co-occurrence frequency by room (% of room frames); (c) light-relay decision distribution by room; (d) confusion matrix for the room-classification model (logistic regression, held-out test n = 1,050).

VI. SECURITY CONSIDERATIONS

The system interfaces with mains electrical loads and a gas safety subsystem, requiring security controls to prevent unauthorized access and ensure safe operation.

Dashboard Authentication: The web dashboard is protected by HTTP Basic Authentication implemented on the ESP32 web server. A username and password are required to access the dashboard interface, preventing unauthorized local network users from controlling loads. Credentials are stored in the ESP32's flash memory using the Preferences library.

Firebase Security: The Firebase Realtime Database enforces authentication for all read and write operations via Security Rules. Firebase App Check is available and recommended to protect against abuse, billing fraud, and unauthorized API usage [41]; configuration is straightforward for production deployment. The App Check attestation token is included with each database request, ensuring requests originate from legitimate ESP32 and Raspberry Pi clients.

Transport Security: All communication between ESP32 nodes, the Raspberry Pi, and the Firebase Realtime Database is encrypted using TLS 1.3 over HTTPS. The ESP32 Firebase library is configured to enforce certificate validation, protecting data from eavesdropping and tampering.

Relay Fail-Safe Behaviour: On loss of Wi-Fi or cloud connectivity, the load-control ESP32 de-energizes all relays after a 60-second grace period. The gas safety subsystem continues local detection and alarm independently of cloud connectivity. A hardware watchdog timer resets the ESP32 if the system becomes unresponsive, ensuring fail-safe operation.

Credential Storage: Firebase credentials are stored in encrypted flash on the ESP32 using the ESP-IDF NVS library. The Raspberry Pi stores credentials in a .env file with restricted permissions (chmod 600), and the Firebase Admin SDK authenticates using a service account JSON key file.

VII. Discussion

A. Experimental Baseline Comparison: PIR vs Proposed Vision-Based Robot

A controlled experiment was conducted to compare the proposed mobile vision-based system with the conventional PIR sensor under the same occupancy conditions. The experiment was conducted in a single apartment, with 50 paired trials for each system and a single test subject. Therefore, it is a one-subject, one-location feasibility demonstration, and the quantitative differences reported should be read as indicative of the qualitative trade-off between motion-triggered PIR and vision-based patrol and should not be read as population estimates.

Experimental Setup: The test site was a 2-room house. An HC-SR501 PIR sensor was placed in Room 1 and connected to the ESP32, which sampled at 1 Hz, communicated with the sensor, and recorded all events. The proposed robotic system was deployed to Room 2, and it was patrolling according to its regular schedule and the MobileNet-SSD inference pipeline. Both systems were assessed simultaneously, with one test subject performing the same movement across all scenarios and another person manually recording the ground-truth occupancy. Five standardized scenarios were implemented, a total of 10 trials per scenario per system (50 trials per system), as shown in Table XVII.

Table XVIII presents the experimental comparison of PIR versus the proposed robot across all five scenarios, while Table XIX summarises the aggregate performance metrics including accuracy, precision, recall, F1 score, stationary detection rate, and mean latency for both systems.

TABLE XVII
TEST SCENARIOS FOR PIR VS ROBOT COMPARISON

ID	Description	Duration	Ground Truth
S1	Person walks in, stands, exits	60 s	Occupied (transient)
S2	Person sits motionless reading	120 s	Occupied (stationary)
S3	Person lies on sofa, eyes closed	120 s	Occupied (stationary)
S4	Room empty, no person	60 s	Empty
S5	Person present under low light	60 s	Occupied (low vis.)

TABLE XVIII
EXPERIMENTAL COMPARISON OF PIR VS PROPOSED ROBOT

Scenario	Truth	PIR	Robot	Observation
S1—Walking	Occupied	10/10 ✓	10/10 ✓	Both reliable with motion
S2—Sitting	Occupied	0/10 ✗	8/10 ✓	PIR fails; robot succeeds
S3—Sleeping	Occupied	0/10 ✗	7/10 ✓	PIR fails; robot partial
S4—Empty	Empty	3/10 FP	1/10 FP	PIR: nuisance triggers
S5—Low light	Occupied	8/10 ✓	6/10 ✓	PIR unaffected by light

TABLE XIX
AGGREGATE PERFORMANCE METRICS (N = 50 TRIALS EACH)

Metric	PIR Sensor	Proposed Robot
Accuracy	50% (25/50)	80% (40/50)
Precision	85.7% (18/21)	96.9% (31/32)
Recall	45.0% (18/40)	77.5% (31/40)
False Negative Rate	55.0%	22.5%
False Positive Rate	30%	10%
F1 Score	59%	86%
Stationary detect (S2+S3)	0% (0/20)	75% (15/20)
Low-light detect (S5)	80% (8/10)	60% (6/10)
Mean latency	0.8 s	4.2 s

In areas of frequent movement, such as corridors, bathrooms, kitchens, PIR is still the most effective solution as it uses less energy, is more cost effective and has a very quick response time. The vision-based robot provides detection which PIR can't match, though at the expense of increased self-consumption and higher latency, in rooms where people sit to read, sleep, or work at a desk. In choosing a system, therefore, the room-level "occupancy patterns" must be taken into account as well as the "aggregate savings" data.

B. Limitations and Threats to Validity

The results presented in this study should be interpreted in light of several limitations, each of which provides a clear direction for future research.

First, a controlled person-occupancy accuracy evaluation (V-A), based on 50 manually annotated images of a single apartment, yields a 95% Wilson confidence interval of 69% - 90%. This number of samples is appropriate for a feasibility proof of concept but not for statistically valid claims of accuracy. To address this, we also analyzed a much larger operational detection log of 5,249 frames from extended patrol in five room categories (Section V-I). This log does not supplant "person-occupancy accuracy" as a ground truth measure but shows that the vision pipeline performs well over thousands of patrol cycles ("no mismatches" between detected objects and relay decisions throughout all frames), and that vectors of detected objects provide enough room-discriminative signal for 87.4% held-out room classification. However, a multi-site, multi-subject collection of at least 1,000 manually annotated images per condition is still needed for future work to allow fine-tuning of model parameters and strong benchmarking.

Second, the robot patrols the site periodically (T_cycle ~ 5 min), so rooms emptied shortly after a patrol pass are energized until they are visited again. This timing penalty is included in the energy model ($\eta_{patrol} \approx 0.90$–$0.96$). This penalty could be reduced by combining the patrol robot with low-power fixed sensors (PIR or mmWave) to detect occupancy changes between patrols; this will be pursued in the future.

Thirdly, all energy and economic savings are represented in a model rather than sub-metered. The simplifying assumptions of the parametric model (Equations 4–7) are empty hours, detection specificity, and robot self-consumption. These estimates will need to be validated by field data with sub-metered data over a number of weeks. The payback analysis is also limited to omission of lifecycle costs, e.g., maintenance and battery replacement—these costs should be part of a complete economic evaluation.

The fourth is that it is not fully self-piloted as some areas require a physical track for assistance. To overcome this, future work is suggested that uses LiDAR or a depth camera for SLAM-based navigation. Further, it is not currently able to operate offline and relies on Wi-Fi connectivity, with the possibility of implementing ESP-NOW or local MQTT to enhance offline capabilities.

Fifth, only MobileNet-SSD was tested. The next step of the work will compare this with the latest lightweight architectures (such as YOLO-Nano, EfficientDet-Lite, YOLOv8n) on the same hardware to determine the balance among accuracy, latency, and power consumption for deployment on the edge.

Sixth, the gas detection subsystem, while characterized over 50 repeated trials and 7 days of normal cooking monitoring, has several limitations that should be acknowledged. First, all response time measurements were conducted in a single kitchen environment with fixed ventilation conditions; response times will vary with room volume, air circulation, sensor placement, and ambient temperature/humidity. Second, the false-alarm rate of 0.18 false alarms/hour of cooking (3 events in 16.8 cooking hours) suggests that further reduction may be needed for user acceptance; the two-stage detection logic described in Section V-I addresses this but has not yet been validated in long-term deployment. Third, the MQ-6 sensor exhibits long-term baseline drift (typically ±10% per month) and requires periodic recalibration; this maintenance requirement was not included in the economic analysis and should be considered in lifecycle cost calculations. Fourth, the sensor responds to a range of combustible gases and cannot distinguish LPG from other volatile organic compounds that may be present during cooking; the quoted detection concentration of 2,100 ppm (10% LEL) is based on LPG calibration gas, and sensitivity to other gases may differ. A formal validation against a reference instrument (e.g., electrochemical or infrared gas sensor) over a 30-day deployment is identified as a priority for future work, along with an investigation of the multi-stage detection logic in terms of false-alarm rate and missed-detection probability.

Seventh, while raw byte UART commands achieved 100% success across all 19 hardware tests, a formal UART error rate characterization over a larger byte count (e.g., 10,000+ bytes) was not performed. This is identified as future work to provide statistical confidence in the error rate under worst-case motor PWM noise.

The prototype has shown technical feasibility of a single mobile platform with vision-based occupancy detection, relay-level load control, cloud synchronization, and gas-safety monitoring for the first time, which has not been achieved in a single, low-cost system before. The limitations identified above do not invalidate the proof-of-concept contribution; rather, they define a concrete roadmap toward practical deployment.

C. Sustainability and SDG Alignment

The energy-saving aspects of the system align well with the goals of SDG 7 (Affordable and Clean Energy), as it automatically switches off unoccupied rooms to prevent unnecessary energy consumption. Its reliance on open-source software and commercially available off-the-shelf components supports SDG 9 (Industry, Innovation, and Infrastructure, targets 9.5 and 9.b). Occupancy-responsive demand reduction aligns with the principles of SDG 11 (Sustainable Cities and Communities) and SDG 12 (Responsible Consumption and Production) and has an indirect qualitative alignment with SDG 13 (Climate Action) through potential generation-side emissions reductions.

The SDG alignment described above is qualitative in the sense that it is not based on metered field data, but it can be given a first-order quantitative estimate. Applying Bangladesh's officially reported grid emission factor of 0.62 kg CO_2 per kWh [50] to the modeled net savings in Table XI converts the base-case saving of 259 kWh/year into approximately 161 kg CO_2/year of avoided grid emissions. These figures give SDG 13 (Climate Action) alignment a concrete, citable magnitude rather than a purely directional claim. They remain estimates based on modeled, not metered, energy savings; however, a formal lifecycle assessment of the hardware itself is beyond the scope of this prototype study. Metered field validation of the underlying kWh savings is therefore still required before these figures can be reported as a verified emissions reduction, and a quantitative SDG impact assessment based on such metered data is identified as a priority future work item.

VIII. Conclusion and Future Work

This paper presented a low-cost (< USD 200) mobile robotic system that integrates on-device MobileNet-SSD occupancy detection, relay-based load control, Firebase cloud synchronization, and an independent kitchen gas-safety subsystem. On a 50-image test set, the detector achieved 82% accuracy (95% Wilson CI: 69–90%), 95.7% precision, 73.3% recall, and an 83% F1 score. All two-room hardware tests passed; the gas-safety subsystem performed as designed; and Firebase latency remained below 500 ms throughout.

Modeled net savings are approximately 259 kWh/year for fan-and-lighting loads, corresponding to annual bill reductions of approximately 2,007 BDT. Simple payback is approximately 11.4 years as built, decreasing to 5.8 years with lower-cost compute hardware. The robot is not the most cost-effective solution solely for load control purposes; it's useful for detecting still people when PIRs don't; for combining multiple functions into one mobile platform; and for distributing the cost of one sensing unit across multiple rooms, with on-device privacy.

It is a proof of concept, and not an actual product. Strengthening the empirical claims and achieving practical deployment will require the following: (1) metered, multi-week field validation using a sub-meter to replace modelled savings with measured data; (2) a multi-home dataset of at least 1,000 images across varied apartments, lighting conditions (morning, afternoon, evening, night), occupancy configurations (single and multiple occupants, stationary and moving), and camera angles, followed by fine-tuning of the MobileNet-SSD model and benchmarking against other lightweight architectures (e.g., YOLO-Nano, EfficientDet-Lite) for edge deployment, leveraging publicly available vision datasets (e.g., the PASCAL VOC dataset [46], and energy datasets [51], and open-sourcing the resulting dataset; (3) SLAM-based navigation using LiDAR or depth cameras to remove the floor-track requirement; (4) lower-power edge compute (e.g., Raspberry Pi Zero 2 W, Coral TPU, or ESP32-S3 with camera module) to reduce self-consumption and shorten payback; (5) hybrid sensing combining the patrol robot with low-power fixed sensors (PIR or mmWave) to capture occupancy changes between patrol visits and reduce the timing penalty; (6) battery and runtime characterisation, an offline mesh fallback using ESP-NOW or local Message Queuing Telemetry Transport (MQTT), a mobile application with push notifications, and coordinated multi-robot operation for larger buildings; and (7) quantitative SDG impact assessment with verified CO_2 displacement using the Bangladesh grid emission factor.

IX. Ethics and Privacy Statement

All human subjects involved in the experimental evaluation (Section V-A, Section V-C, and Section VI-A) provided informed consent prior to participation. The experiments consisted of routine activities (walking, sitting, lying down) in a residential setting and did not involve any medical procedures, interventions, or collection of sensitive personal data. No personally identifiable information (names, addresses, contact details) was collected.
The vision-based occupancy detection system processes all image data locally on the Raspberry Pi (on-device inference) and does not transmit or upload any images or video footage to external servers or the cloud. Frames archived on-device for the offline analysis in Section V-K were stored and processed locally on the Raspberry Pi and were never uploaded, so on-device-only handling is preserved. This design ensures occupant privacy and is consistent with the privacy-preserving approach described in Section II and Section III-D. The study was conducted in accordance with the ethical principles outlined in the Declaration of Helsinki and institutional guidelines for research involving human participants.

Acknowledgement

The authors would like to thank Mr. Shuayeb Parvez Shuvo (AIUB ID: 22-49508-3), Mr. Md. Nazmul Alam (AIUB ID: 22-48525-3), and Mr. Waresul Arefin (AIUB ID: 22-49184-3) for their valuable support throughout this research.

References

[1] M. S. Aliero, M. Q. Raza, M. A. Pasha, and G. A. Q. A. S. H. S. Al-Ghamdi, "Smart home energy management systems in Internet of Things networks for green cities demands and services," Environ. Technol. Innov., vol. 22, art. no. 101443, May 2021, doi: 10.1016/j.eti.2021.101443.
[2] Y. Yuan, C. Song, and L. Gao, "A review of current research on occupant-centric control for improving comfort and energy efficiency," Build. Simul., vol. 17, no. 10, pp. 1675–1692, Oct. 2024, doi: 10.1007/s12273-024-1164-y.
[3] A. Natarajan, V. Krishnasamy, and M. Singh, "Occupancy detection and localisation strategies for demand modulated appliance control in IoT-enabled home energy management system," Renew. Sustain. Energy Rev., vol. 167, art. no. 112731, Oct. 2022, doi: 10.1016/j.rser.2022.112731.
[4] R. Razavi, A. Gharipour, M. Fleury, and I. E. Akpan, "Occupancy detection of residential buildings using smart meter data: A large-scale study," Energy Build., vol. 183, pp. 195–208, Jan. 2019, doi: 10.1016/j.enbuild.2018.11.025.
[5] J. Vanus, R. Martinek, M. Nedoma, and J. Machacek, "Design of a new method for occupancy monitoring in smart home care with autonomous mobile robot within Internet of Things," Scientific Reports, vol. 15, no. 1, p. 31767, 2025, doi: 10.1038/s41598-025-16806-8.
[6] J. Kim, S. Lee, and J. Choi, "An IoT-based home energy management system over dynamic home area networks," Int. J. Distrib. Sensor Netw., vol. 11, no. 9, art. no. 828473, Sep. 2015, doi: 10.1155/2015/828473.
[7] A. Al-Fuqaha, M. Guizani, M. Mohammadi, M. Aledhari, and M. Ayyash, "Internet of Things: A survey on enabling technologies, protocols, and applications," IEEE Commun. Surveys Tuts., vol. 17, no. 4, pp. 2347–2376, Fourth Quarter 2015, doi: 10.1109/COMST.2015.2444095.
[8] A. Safdarian, M. Fotuhi-Firuzabad, and M. Lehtonen, "A distributed algorithm for managing residential demand response in smart grids," IEEE Trans. Ind. Informat., vol. 10, no. 4, pp. 2385–2393, Nov. 2014, doi: 10.1109/TII.2014.2350456.
[9] M. Rastegar, M. Fotuhi-Firuzabad, and H. Zareipour, "Home energy management incorporating operational priority of appliances," Int. J. Electr. Power Energy Syst., vol. 74, pp. 286–292, Jan. 2016, doi: 10.1016/j.ijepes.2015.07.024.
[10] A. Safdarian, M. Fotuhi-Firuzabad, and M. Lehtonen, "Domestic EWH and HVAC management in smart grids: Potential benefits and realization," Electr. Power Syst. Res., vol. 134, pp. 38–46, May 2016, doi: 10.1016/j.epsr.2015.12.025.
[11] P. Chaudhari, Y. Xiao, M. M.-C. Cheng, and T. Li, "Fundamentals, algorithms, and technologies of occupancy detection for smart buildings using IoT sensors," Sensors, vol. 24, no. 7, p. 2123, Mar. 2024, doi: 10.3390/s24072123.
[12] A. J. Amiri, "A review of occupancy detection techniques for HVAC control: Advances and practical challenges," J. Build. Eng., vol. 113, art. no. 113962, Jan. 2025, doi: 10.1016/j.jobe.2025.113962.

[13] U. U. Rehman, P. Faria, and L. Gomes, "Future of energy management models in smart homes: A systematic literature review of research trends, gaps, and future directions," Process Integr. Optim. Sustain., vol. 9, pp. 1169–1198, Apr. 2025, doi: 10.1007/s41660-025-00400-0.
[14] E. Mocanu, P. H. Nguyen, M. Gibescu, and W. L. Kling, "Deep learning for estimating building energy consumption," Sustain. Energy Grids Netw., vol. 6, pp. 91–99, Jun. 2016, doi: 10.1016/j.segan.2016.02.005.
[15] W. Kong, Z. Y. Dong, Y. Jia, D. J. Hill, Y. Xu, and Y. Zhang, "Short-term residential load forecasting based on LSTM recurrent neural network," IEEE Trans. Smart Grid, vol. 10, no. 1, pp. 841–851, Jan. 2019, doi: 10.1109/TSG.2017.2753802.
[16] S. Zhang and C. Kong, "Deep learning and multi-objective optimization for real-time occupancy-based energy control in smart buildings," Sci. Rep., vol. 15, p. 39466, 2025, doi: 10.1038/s41598-025-23134-4.
[17] L. Rueda, K. Agbossou, A. Cardenas, N. Henao, and S. Kélouwani, "A comprehensive review of approaches to building occupancy detection," Building and Environment, vol. 180, Art. no. 106966, Aug. 2020, doi: 10.1016/j.buildenv.2020.106966.
[18] M. Umair, M. A. Khan, M. J. Khan, A. U. Rehman, and M. U. I. Khan, "Energy management of smart homes over fog-based IoT architecture," Sustain. Comput. Informat. Syst., vol. 39, art. no. 100898, Sep. 2023, doi: 10.1016/j.suscom.2023.100898.
[19] H. Kuchuk and E. Malokhvii, "Integration of IoT with Cloud, Fog, and Edge Computing: A Review," Advanced Information Systems, vol. 8, no. 2, pp. 65–78, 2024, doi: 10.20998/2522-9052.2024.2.08.
[20] M. Engin and D. Engin, "Path planning of line follower robot," in 2012 5th European DSP Education and Research Conference (EDERC), Amsterdam, Netherlands, Sep. 2012, pp. 1–5, doi: 10.1109/EDERC.2012.6532213.
[21] M. B. Holder, M. M. Trivedi, and S. B. Marapane, "Mobile robot navigation by wall following using a rotating ultrasonic scanner," in Proceedings of 13th International Conference on Pattern Recognition (ICPR), Vienna, Austria, Aug. 1996, vol. 3, pp. 298–302, doi: 10.1109/ICPR.1996.546959.
[22] R. Mur-Artal and J. D. Tardós, "ORB-SLAM2: An Open-Source SLAM System for Monocular, Stereo and RGB-D Cameras," IEEE Trans. Robot., vol. 33, no. 5, pp. 1255–1262, Oct. 2017, doi: 10.1109/TRO.2017.2705103.
[23] T. Shan and B. Englot, "LeGO-LOAM: Lightweight and ground-optimized lidar odometry and mapping on variable terrain," in Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS), Madrid, Spain, Oct. 2018, pp. 4758–4765, doi: 10.1109/IROS.2018.8594299.
[24] P. Dario, E. Guglielmelli, V. Genovese, and M. Toro, "Robot assistants: Applications and evolution," Robotics and Autonomous Systems, vol. 18, nos. 1–2, pp. 225–234, Jul. 1996, doi: 10.1016/0921-8890(96)00006-1.
[25] A. Shokrollahi, J. A. Persson, R. Malekian, A. Sarkheyli-Hägele, and F. Karlsson, "Passive Infrared Sensor-Based Occupancy Monitoring in Smart Buildings: A Review of Methodologies and Machine Learning Approaches," Sensors, vol. 24, no. 5, p. 1533, Feb. 2024, doi: 10.3390/s24051533.
[26] W. Zhang, J. Calautit, P. W. Tien, Y. Wu, and S. Wei, "Deep learning models for vision-based occupancy detection in high occupancy buildings," J. Build. Eng., vol. 98, p. 111355, Dec. 2024, doi: 10.1016/j.jobe.2024.111355.
[27] C. Guo, B. Hu, W. Zhang, Y. Gao, and H. Liu, "IoT-Based Edge Computing and Image Processing for Occupancy Detection," in 2023 IEEE 11th International Conference on Information, Communication and Networks (ICICN), Xi'an, China, Aug. 2023, pp. 385–390, doi: 10.1109/ICICN56848.2023.10393844.
[28] H. Honmote, S. Gadekar, P. Katta, and M. Kulkarni, "Real Time Object Detection and Recognition using MobileNet-SSD with OpenCV," Int. J. Eng. Res. Technol., vol. 11, no. 1, Jan. 2022, doi: 10.17577/IJERTV11IS010070.
[29] V. Kamath and A. Renuka, "Investigation of MobileNet-Ssd on human follower robot for stand-alone object detection and tracking using Raspberry Pi," Cogent Eng., vol. 11, no. 1, p. 2333208, Apr. 2024, doi: 10.1080/23311916.2024.2333208.
[30] P. Mittal, "A comprehensive survey of deep learning-based lightweight object detection models for edge devices," *Artif. Intell. Rev.*, vol. 57, p. 242, Aug. 2024, doi: 10.1007/s10462-024-10877-1.
[31] R. K. Sharma and G. Verma, "Benchmarking and Performance Analysis of a Deep Learning Model on Edge Computing Platforms for Occupancy Detection in Smart Buildings," *Nat. Acad. Sci. Lett.*, 2025, doi: 10.1007/s40009-025-01583-1.
[32] A. N. Sayed, Y. Himeur, and F. Bensaali, "Deep and transfer learning for building occupancy detection: A review and comparative analysis," *Eng. Appl. Artif. Intell.*, vol. 115, p. 105254, Oct. 2022, doi: 10.1016/j.engappai.2022.105254.
[33] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, "MobileNetV2: Inverted Residuals and Linear Bottlenecks," in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR), Salt Lake City, UT, USA, Jun. 2018, pp. 4510–4520, doi: 10.1109/CVPR.2018.00474.
[34] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg, "SSD: Single Shot MultiBox Detector," in Proc. Eur. Conf. Comput. Vis. (ECCV), Amsterdam, Oct. 2016, pp. 21–37, doi: 10.1007/978-3-319-46448-0_2.
[35] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You Only Look Once: Unified, Real-Time Object Detection," in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), Las Vegas, NV, USA, Jun. 2016, pp. 779–788, doi: 10.1109/CVPR.2016.91.
[36] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam, "MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications," arXiv preprint, arXiv:1704.04861, Apr. 2017.
[37] N. D. Lane, S. Bhattacharya, P. Georgiev, C. Forlivesi, L. Jiao, L. Qendro, and F. Kawsar, "DeepX: A Software Accelerator for Low-Power Deep Learning Inference on Mobile Devices," IEEE Des. Test, vol. 34, no. 6, pp. 80–89, Dec. 2017, doi: 10.1109/MDAT.2016.2626077.
[38] L. Khakim, I. Afriliana, E. Budihartono, and A. Suharjono, "An Experimental Evaluation of Low-Cost MQ-Series LPG Sensors under Varying Temperature Conditions", *Eng. Technol. Appl. Sci. Res.*, vol. 16, no. 1, pp. 31475–31483, 2026, doi: 10.48084/etasr.15843
[39] B. Jena, S. K. Pradhan, R. Jha, S. Goel, and R. Sharma, "LPG gas leakage system using IoT," Mater. Today Proc., vol. 74, pp. 795–800, 2023, doi: 10.1016/j.matpr.2022.11.172.
[40] K. Kumar, A. Verma, and P. Verma, "IoT-HGDS: Internet of Things integrated machine learning based

hazardous gases detection system for smart kitchen," Internet Things, vol. 28, art. no. 101396, Dec. 2024, doi: 10.1016/j.iot.2024.101396.
[41] Google LLC, "Choose database locations for Realtime Database," Firebase Documentation. Accessed: Jun. 20, 2026. [Online]. Available: https://firebase.google.com/docs/database/locations
[42] F. Palumbo, G. Aceto, A. Botta, D. Ciuonzo, V. Persico, and A. Pescapé, "Characterization and analysis of cloud-to-user latency: The case of Azure and AWS," Computer Networks, vol. 184, pp. 107693, 2021, doi: 10.1016/j.comnet.2020.107693.
[43] A. Koushal, R. Gupta, F. Jan, Kamaldeep, and V. Kumar, "Home Automation System Using ESP32 and Firebase," in Proc. 2022 7th Int. Conf. Parallel, Distrib. Grid Comput. (PDGC), Solan, India, Nov. 2022, pp. 228–231, doi: 10.1109/PDGC56933.2022.10053309.
[44] National Fire Protection Association, NFPA 715: Standard for the Installation of Fuel Gas Detection and Warning Equipment. Quincy, MA, USA: National Fire Protection Association, 2023.
[45] Underwriters Laboratories Inc., UL 1484: Residential Gas Detectors, 5th ed. Northbrook, IL, USA: Underwriters Laboratories Inc., 2022.
[46] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman, "The Pascal Visual Object Classes (VOC) Challenge," Int. J. Comput. Vis., vol. 88, no. 2, pp. 303–338, Jun. 2010, doi: 10.1007/s11263-009-0275-4.
[47] V. G. Sánchez, I. Taylor, and P. C. Bing-Jonsson, "Ethics of smart house welfare technology for older adults: A systematic literature review," Int. J. Technol. Assess. Health Care, vol. 33, no. 6, pp. 691–699, 2017, doi: 10.1017/S0266462317000964.
[48] E. B. Wilson, "Probable inference, the law of succession, and statistical inference," J. Am. Stat. Assoc., vol. 22, no. 158, pp. 209–212, Jun. 1927, doi: 10.1080/01621459.1927.10502953.
[49] Dhaka Power Distribution Company Ltd. (DPDC), "Residential electricity tariff schedule," 2025. [Online]. Available: https://www.dpdc.gov.bd/tariff. Accessed: Jun. 20, 2026.
[50] Department of Environment, Government of the People's Republic of Bangladesh, "Grid Emission Factor (GEF) for Bangladesh," 2023. [Online]. Available: https://www.doe.gov.bd/gef. Accessed: Jun. 20, 2026.
[51] C. Beckel, W. Kleiminger, R. Cicchetti, T. Staake, and S. Santini, "The ECO data set and the performance of non-intrusive load monitoring algorithms," in *Proceedings of the 1st ACM Conference on Embedded Systems for Energy-Efficient Buildings (BuildSys)*, Memphis, TN, USA, Nov. 2014, pp. 80–89, doi: 10.1145/2674061.2674064.

Nafiz Ahmed Chisty completed his Master's degree in System-on-Chip (SoC) Design from Lund University (LTH), Sweden, and his Bachelor of Science degree in Electrical and Electronic Engineering from American International University-Bangladesh (AIUB), Dhaka.

He began his academic career as a Lecturer in the Department of Electrical and Electronic Engineering at American International University-Bangladesh in January 2006, following the completion of his BSc degree. Since then, he has been serving as a full-time faculty member, teaching both theoretical and laboratory courses to undergraduate engineering students. He currently serves as an Associate Professor and Head of the Department of Electrical and Electronic Engineering at American International University-Bangladesh.

He is a Senior Member of the Institute of Electrical and Electronics Engineers (IEEE) and a Fellow of the Institution of Engineers, Bangladesh (IEB). He has an active academic and research profile, with scholarly contributions published in international journals and conference proceedings, in addition to his contributions as an author of academic and technical books.

He has supervised numerous undergraduate final-year projects and graduate thesis projects, providing academic and research guidance to students in various areas of electrical and electronic engineering. His research interests include IoT-based systems, energy-efficient digital ASIC implementation, low-complexity algorithm implementation in HDL, FPGA applications in wireless communication, and microcontroller-based systems.

Dr. Mohammad Alif Arman completed his B.Sc. degree in EEE from American International University-Bangladesh in 2006. He also completed his M.Sc. and Ph.D. studies at Lund University, Sweden. For his master's project, he performed research on the chemistry of graphene.

His Ph.D. project focused on obtaining a profound knowledge of the catalytic properties of cobalt oxide surfaces.

In addition to his own Ph.D. project, he has also been involved in research projects focused on graphene chemistry and iron oxide film chemistry. He has performed numerous experiments, particularly at the MAX IV laboratory and at other synchrotron facilities around the world. He has also acted as a local contact, guiding other Ph.D. students or researchers through the operation of complex instruments and research. Currently, he is working as an Assistant Professor in the Faculty of Engineering at AIUB. His research interests include Material Science, Nano Electronics, and IoT.

SM Walid received his HSC from Motijheel Govt. Boys College in 2019. He is currently pursuing his B.Sc. degree in Electrical and Electronic Engineering, with a capstone thesis concentration in

Biomedical Technology, at the American International University of Bangladesh (AIUB) from January 2023 to September 2026. During his academic tenure, he has consistently demonstrated excellence receiving the Deans's List Award for the Spring 23-2024, Fall 23-2024, and Spring 24-2025 semesters. His primary research focusses on Machine Learning applications in healthcare, IoT integrations,material science, and semiconductor technologies. He has developed advanced systems, including a predictive diagnostic tool for early preeclampsia detection using supervised machine learning and a smart medicine box featuring multimodal large language models and real time IoT synchronization. His research interests include predictive modeling, regression algorithms, deep neural networks, and electronic materials, highlighted by his publication on charge transport and band structure engineering in May 2026.

Prof. Mohammad Shorif Uddin earned his Doctor of Engineering (PhD) from the Kyoto Institute of Technology, Japan, in 2002; a Master of Technology Education from Shiga University, Japan, in 1999; a Bachelor of Science in Electrical and Electronic Engineering from the Bangladesh University of Engineering and Technology in 1991; and an MBA from Jahangirnagar University in 2013. His academic journey began in 1991 as a Lecturer at the Bangladesh Institute of Technology, Chittagong (now CUET).

In 1992, he joined the Department of Computer Science and Engineering at Jahangirnagar University, where he currently serves as a Professor. Throughout his career, he has made significant contributions, including serving as Chairman of the Computer Science and Engineering Department and as Teacher-in-Charge of the ICT Cell at Jahangirnagar University. He has been serving as the Vice Chancellor of Green University of Bangladesh since May 16, 2024, while on academic leave from his permanent position at Jahangirnagar University.

He has an extensive international research background, having conducted postdoctoral research at several renowned institutions, including the Bioinformatics Institute in Singapore, Toyota Technological Institute in Japan, Kyoto Institute of Technology in Japan, Chiba University in Japan, Saitama University in Japan, Bonn University in Germany, and the Institute of Automation, Chinese Academy of Sciences in China. His professional and research interests encompass a wide range of areas, including Computer Vision, Agro-Biomedical Imaging, Image Security, Artificial Intelligence, Machine Learning, IoT and Smart Technologies.

He has a prolific publication record, with over 270 research papers published in prestigious venues such as Elsevier, Springer, IEEE, ACM, Wiley, Oxford University Press, SPIE, and Optica, accumulating over 7,300 Google Scholar citations. Additionally, he holds two patents for his innovative scientific work and has edited numerous books and authored several book chapters published by Springer and Elsevier.

He has delivered many keynote speeches at international conferences and played pivotal roles as a General Chair, TPC Chair, or Co-Chair at various international conferences. His outstanding contributions have earned him accolades such as the Best Paper Award at ICIEV 2013 in Dhaka, Bangladesh, and the Best Presenter Award at ICCVG 2004 in Warsaw, Poland. Notably, in 2004.

He gained worldwide recognition from prominent news and television media outlets, including CNN, BBC, Reuters, The New York Times, The Japan Times, and Associated Press, for his groundbreaking invention of a blind navigational aid called the "Electronic Eye."

Furthermore, he mentored and coached the Jahangirnagar University ACM ICPC World Finals Teams in 2015 and 2017 and supervised numerous doctoral and Master's theses.

He is honored as a Fellow of IEB and BCS, a Senior Member of IEEE, and serves as an Associate Editor of IEEE Access.