Smart Parking System

S Chandra Sekhar[1], Varsha N[2], Kavitha H K[3], Aparna N[4]
[1]*Asst. Prof., Department of Electronics and Communication Engineering, APS College of Engineering, Bengaluru, India*
[2,3,4]*Department of Electronics and Communication Engineering, APS College of Engineering, Bengaluru, India*

***Abstract*—The rapid growth in the number of vehicles has created a significant demand for efficient and automated parking solutions. Traditional parking systems commonly depend on security personnel, manually operated barriers, and paper or token-based records, which can result in delays, human errors, and difficulty in obtaining current information about available parking spaces. To address these limitations, this paper proposes an Automatic Smart Parking Management System that integrates and automates major parking operations, including advance slot reservation, vehicle identification, entry verification, parking-space allocation, occupancy monitoring, wrong-slot detection, parking-time calculation, digital payment, and automated vehicle exit. setting of the prototype shows that the proposed system can effectively perform vehicle identification, parking-slot allocation, occupancy monitoring, wrong-slot detection, and automatic barrier operation. The results demonstrate the potential of the proposed architecture to reduce manual involvement, improve parking security, minimize unnecessary waiting and congestion, and provide a better parking experience.**

***Index Terms*—Smart Parking, Automatic Number Plate Recognition (ANPR), IoT, ESP32, Ultrasonic Sensor, Slot Booking, Vehicle Navigation, OpenCV, Embedded Systems**

I. INTRODUCTION

The rapid growth of cities and the increasing use of private vehicles have created significant difficulties in managing parking spaces. Many commercial buildings, offices, and residential communities still depend on manual parking procedures, where security staff verify vehicles, record information manually, and guide drivers to available spaces. This approach can cause delays, especially during busy periods, and makes efficient parking management difficult. It also provides limited control over unauthorized vehicles entering or occupying reserved parking slots.

The integration of Internet of Things (IoT) and computer-vision technologies enables parking operations to be automated from entry to exit.

Ultrasonic and infrared sensors can continuously monitor the availability of individual parking spaces, while camera-based Automatic Number Plate Recognition (ANPR) systems can identify vehicles without relying on RFID cards, parking tickets, or manual verification. By connecting these technologies to a cloud-based web application, users can book parking spaces before arrival, obtain guidance to their allocated slot, and complete payments digitally. This integrated approach helps reduce waiting time and minimizes congestion at parking entry and exit points.

Automatic Smart Parking Management System that combines embedded hardware, Automatic Number Plate Recognition (ANPR), a robotic vehicle, and a real-time web-based dashboard. The proposed system demonstrates the automation of the complete parking process, including advance booking, vehicle entry verification, guided navigation, parking-slot monitoring, detection of incorrect slot usage, duration-based fee calculation, and automated exit. The prototype is developed using affordable and easily accessible components, providing a practical foundation that can be further expanded for implementation in larger multi-level parking facilities.

II. LITERATURE SURVEY

The increasing number of vehicles in urban areas has created a need for efficient parking management. An IoT-based smart parking system can use cameras and computer-vision techniques to

monitor parking spaces and identify vacant locations. The detected parking information is updated continuously so that users can easily check available spaces through an Android application. The application provides current parking information and helps users locate suitable spaces without spending excessive time searching. [1]

Smart parking technology provides real-time information about available parking spaces and helps improve the overall parking experience. With the continuous increase in vehicles and limited parking areas, managing parking efficiently has become an important challenge in cities and educational institutions. Traditional manual parking methods often cannot provide updated information about vacant spaces, which may lead to unnecessary.[2]

This study presents an intelligent parking solution designed to improve parking management and space utilization in urban areas. The system uses a network of sensors to continuously monitor the status of parking spaces and provide real-time occupancy information. It automates vehicle entry and exit operations, which can help reduce congestion and unnecessary movement within parking facilities. In addition, the system incorporates a route guidance feature that assists drivers in locating vacant parking spaces through updated navigation information.[4]

Finding suitable parking in crowded commercial locations such as shopping malls, airports, and large business complexes has become increasingly challenging due to the growing number of vehicles and limited availability of properly organized parking facilities. Inefficient parking management can lead to traffic congestion, longer waiting times, increased fuel consumption, and inconvenience for visitors.[6]

The proposed system focuses on developing an intelligent parking management solution that allocates available parking spaces according to user requirements in office and residential environments. The system is implemented using Internet of Things (IoT) technology, with the ESP8266 microcontroller serving as the main control unit. It incorporates a reservation-based parking mechanism that enables authorized users to reserve and utilize available parking spaces during specified time periods.[3]

Parking congestion is a common problem on university campuses because of limited parking capacity and ineffective management practices. These issues can result in longer search times, unnecessary fuel consumption, and increased traffic within the campus. To address these challenges, an intelligent parking management system was developed for the Nainital region of Uttarakhand. The proposed approach uses Internet of Things (IoT) technology to monitor and manage the movement of tourist vehicles within designate.[7]

This study focuses on developing an intelligent parking management system for residential communities using mini-server technology. The proposed system is designed to improve the organization and utilization of limited parking spaces while providing a secure and convenient parking solution for residents.

It aims to simplify parking management, reduce conflicts over available spaces, and make better use of the limited parking capacity in residential areas.[5]

A camera installed above the parking area continuously records video of the parking spaces. The captured footage is analyzed using a YOLO-based object detection method to identify whether individual parking spaces are occupied or vacant. The processed occupancy information is then uploaded to an AWS cloud platform, where it is made available through a Progressive Web Application for real-time monitoring and status updates. [9-3]

Efficient car parking plays an important role in managing transportation at tourist destinations.

The increasing number of visitors can significantly affect traffic conditions, particularly in areas with limited parking facilities. Therefore, this study focuses on examining the importance of implementing an efficient parking management system to handle vehicle demand.[3]

III. METHODOLOGY

The system is developed to manage parking activities in an automatic manner. Sensors monitor each parking space and determine whether it is available or occupied. A camera is used to capture the vehicle number plate, which is then processed to identify the

vehicle. The obtained details are stored in a cloud database and used for real-time parking updates. Users and administrators can access parking information through a web-based dashboard. The system can also identify vehicles parked in incorrect slots and record their parking time. A simulated online payment facility is included to calculate the parking charge according to the duration of parking.

A. BLOCK DIAGRAM

The architecture is separated into four responsibilities: a reservation and authorization service, an image-processing service, a stationary sensing/actuation controller, and a moving 4WD controller. The division prevents time-consuming video analysis from blocking microcontroller sensor acquisition and means a robot command cannot directly change a verified booking. A slot is assigned one of the logical labels available, booked, occupied and incorrectly occupied, while RESERVED may be retained as a user-facing alias for a future reservation. Status transitions should be validated by the server, not merely copied from browser inputs.

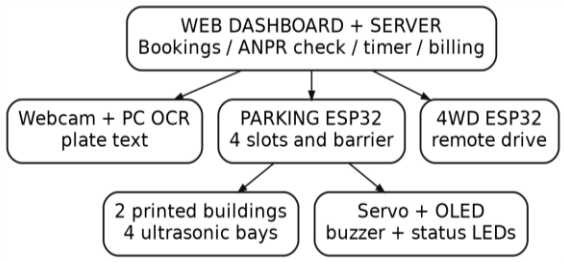


Fig 3.1 Block diagram for Control Station

The web application stores vehicle identity, bookings, assigned building/floor/slot, transaction state and audit history. At entry, the PC-hosted ANPR service submits a normalized OCR candidate to the backend. Only a valid registered plate with an active time-window reservation and an unoccupied assigned slot receives an entry grant. The parking ESP32 actuates the boom barrier and publishes four independent occupancy readings. A second ESP32 is a proposed practical implementation choice for controlling the 4WD vehicle wirelessly; neither board count nor exact pin assignments is fixed by the quotation [7].

Position guidance uses a scaled parking map and a known route or simulated coordinates; it is not a claim of measured indoor GPS accuracy. A single ultrasonic transducer indicates presence, not the vehicle identity. Wrong-slot classification therefore needs an active booking-to-vehicle association or supervised one-vehicle-at-a-time operation.

B. WORKING

The system uses a moisture sensor to continuously monitor soil moisture levels and send real-time data to the ESP32. When the moisture level falls below a predefined threshold, the ESP32 automatically activates the relay to turn on the water pump, ensuring timely irrigation. Users also have the option of manual control by connecting to the ESP32 through the HC-05 Bluetooth module using an Android app to operate the pump as needed. Throughout the process, an LCD display provides live updates, showing sensor readings, pump status, and other important system information.

C. HARDWARE INTEGRATION

The proposed smart parking system uses different hardware components to monitor and control the parking process. The ESP32 microcontroller acts as the main controller of the system. It receives information from the sensors, controls the barrier and display, and provides Wi-Fi communication with the web application. Ultrasonic sensors are placed at each parking slot to measure the distance and identify whether a vehicle is present or the slot is empty. A webcam is used at the entrance to capture the vehicle number plate for automatic vehicle identification and verification.

A servo motor is connected to a 3D-printed boom barrier to control the entry and exit of vehicles automatically. An OLED display is used to show useful information such as slot availability, vehicle verification, entry status, and guidance messages. Buzzers provide audio alerts when an unauthorized vehicle enters or when a vehicle is parked in the wrong slot. A 4WD robotic vehicle with DC geared motors represents the user's car and demonstrates movement inside the parking area. A motor-driver module controls the speed and direction of the robotic vehicle. A battery pack and power-supply regulation circuit provide the required voltage to the different components.

D. SOFTWARE INTEGRATION

The proposed smart parking system uses Python with OpenCV to capture and process vehicle images

for number plate detection. An OCR-based Automatic Number Plate Recognition (ANPR) technique is used to identify and extract the vehicle number from the captured image. A Flask or FastAPI backend manages communication between the hardware, database, and web application through REST APIs.

The user and administrator dashboards are developed using HTML, CSS, and JavaScript to provide an easy and responsive interface. Firebase or a similar cloud database stores important information such as user details, parking bookings, vehicle records, and transaction history.

The ESP32 is programmed to monitor parking sensors, control the boom barrier, and provide Wi-Fi-based communication with the system. A QR-code-based dummy UPI payment interface is also included to simulate parking fee calculation and digital payment during the prototype demonstration.

IV. SYSTEM FLOW AND CONTROL LOGIC

The transaction identifier should persist between dashboard pages and device updates so that a delayed ultrasonic reading is not applied to the wrong user session. The booking process needs a conflict check for overlapping time intervals and should not allow a browser to overwrite another user's reserved space.

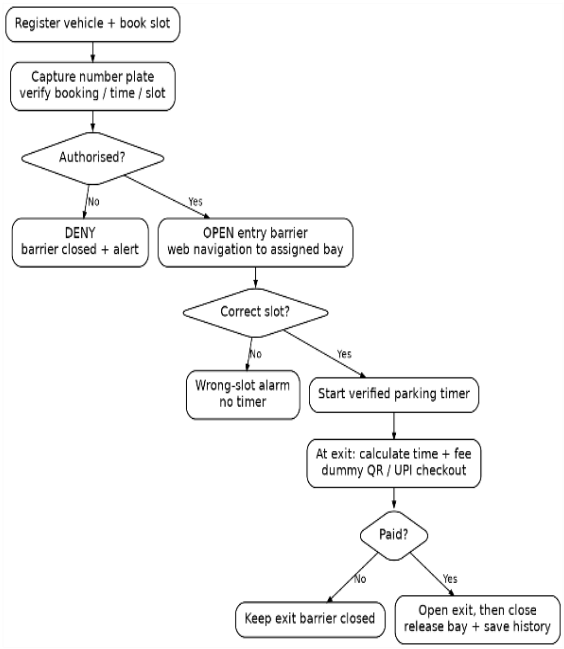


Fig.4 End-to-end transaction flow with rejection conditions.

At the gate, OCR output is normalized for case and whitespace, then compared with registered records; confidence below a chosen threshold should fail safe or request review. Camera errors must not automatically convert the booking to an authorized entry.

A new reservation changes an available bay into a booked state using an atomic server-side check. At the entrance the camera acquires a plate, OCR produces a normalized string, and the back end validates identity, booking window and bay availability. Failure keeps the barrier closed and records a denial. Success displays the assigned space and authorizes a bounded open cycle. The driver remotely commands forward, reverse, left, right, stop and speed; an emergency stop and stale-command timeout are required for safe 4WD operation. If the assigned slot sensor becomes consistently occupied and can be associated with the arriving vehicle, the system records server-side start time. Occupancy of a non-assigned bay instead generates a wrong-slot alert without validating the parking session. Duration at exit is the difference between trusted end and start timestamps, after which a configurable time tariff produces a fee.

The QR/UPI user interface changes a simulated payment flag; it does not process funds. A successful server-approved simulated checkout permits the exit gate to open and then releases the slot.

State and timing rules

Let e_i indicate the filtered presence status at slot i. Then occupancy_count = $\Sigma\, e_i$ for i = 1,.4. For a parked session, $\Delta t = t_exit - t_start$, and a sample charge model is $fee = ceil(\Delta t / T) \times R$, where T is the billable interval and R is the configurable rate; neither numerical tariff nor OCR threshold is prescribed in the quotation. Gate permission follows a server-side predicate: (registered $\wedge$ booked $\wedge$ within_window $\wedge$ slot_available) for entry, and (identity_valid $\wedge$ payment_confirmed) for exit.

V. IMPLEMENTATION AND TEST PLAN

A. OCR and reservation matching procedure

At the shared entry camera, a video frame is acquired only when the robot is positioned in the expected number-plate capture region.

A suggested computer-vision chain is region localization, geometric adjustment, grayscale/contrast

enhancement, OCR text extraction and confidence filtering. Plate syntax is then normalized before lookup. This proposal uses general OpenCV and OCR techniques rather than asserting a trained custom detector or any measured recognition precision. A permitted decision checks the normalized plate against one active reservation, booked time constraints, and the allocated space. OCR should be reattempted or routed for review when the reading is low-confidence, and the database should retain a decision timestamp without exposing image files to unrelated users.

B. Slot detection and wrong-spot correlation

After a known vehicle is admitted, the backend records a temporary admission session. Sequential ultrasonic bursts estimate four slot distances; each reading is filtered across multiple samples and compared with a calibrated empty-bay baseline. The first stable detection on the assigned bay can be associated with the open session in a supervised demonstration. If an unassigned bay becomes occupied, a warning is generated instead. The method cannot uniquely identify multiple simultaneous cars from ultrasonic returns alone. Robust multi-vehicle use would need extra identity checkpoints, computer vision tracking, or RFID per bay, none of which is included as an assumed part of the quoted prototype.

C. Proposed service-to-device integration

The backend sends only a bounded open/close authorization to the gate controller and stores the returned acknowledgement. Ultrasonic readings are timestamped and debounced: several consistent measurements are required before the visible occupancy status is changed. For miniature vehicles, a bench calibration with and without a parked object should precede selection of the distance threshold.

The robot subscribes to a direction command and speed setting; if no fresh command arrives within a configured short interval, it breaks or coasts to a safe stop, depending on the driver wiring. The mobile or browser interface must provide a conspicuous STOP action.

D. Experimental protocol and reporting metrics

Test each sensor with repeated observations of both empty and occupied bays to estimate false-occupied and false-vacant events. For OCR, prepare a labelled image set with known plates at multiple light levels, tilts and entry distances; record exact-string correctness, false access approvals and per-image processing time. For control, capture the delay between browser request and measured servo actuation or wheel movement using time-stamped logs. A successful full-cycle trial requires registration, reservation, gate authorization, correct parking, a nonzero logged duration, dummy payment, exit authorization and the subsequent release of the same slot. Every rate or latency must be accompanied by a trial count and measurement procedure.

The implementation stack proposed in the quotation comprises ESP32 firmware, Python/OpenCV/OCR on a computer, Flask or Fast API, HTML/CSS/JavaScript and a cloud database such as Firebase [7]. The backend should treat raw sensor updates as observations, not as authorization. Only authenticated backend decisions may change booking and payment status. The 4WD control stream should expire to STOP after Wi-Fi loss, and the gate should remain closed under unverified identities or an unavailable service.

Evaluation requires actual trials under multiple lighting and approach angles, occupancy measurements per sensor, a command-to-motion delay log, unsuccessful transaction tests, and repeated end-to-end parking sessions. Plate exact-match accuracy should use correctly read plates divided by readable test plates, with the sample size disclosed. The provided material contains no such trial dataset, test photographs, latency distribution or measured success rates

VI. PROPOSED SYSTEM

The proposed system is designed to automate the major stages of parking management using a scaled-down prototype. The model consists of two open-top 3D-printed parking structures containing four individual parking spaces. A common entry and exit path is provided with a 3D-printed boom barrier controlled by a servo motor. A small four-wheel-drive robotic vehicle is used to represent the user's vehicle during system testing.

VII. EXPECTED RESULTS AND DISCUSSION

A. Evidence required before performance claims

A defensible experimental table would show the measured image count, success count, false

approvals, recognition duration, occupancy classification errors and full-cycle completions. A gate acceptance rate with no description of which plates were presented could be misleading; the system should separately count correctly recognized authorized vehicles and incorrectly authorized unknown vehicles. Latency should be measured end to end from the browser or OCR decision to the physical servo transition. The absence of logged results is a constraint of this design-stage manuscript and must not be interpreted as a successful validation.

B. Expected integration behavior

A structured results record should report sensor-specific thresholds; confusion counts for occupied-versus-empty readings; plate exact-match numerator/denominator; unauthorized-entry false acceptances; booking conflicts rejected; command response distributions; and successful end-to-end transaction cycles.

Presenting this information separately for each sensor and for the whole system would make bottlenecks visible. For example, a false-vacant observation should not alone override an active booking, and a correct plate string should not bypass the payment check at exit. The quotation and accompanying report specify intended behavior and pin-level design; neither supplies enough experimental evidence to populate these outcome cells honestly.

At the design level, the model should expose four distinct bay states, reject unbooked entry, guide an authorized model vehicle to its assigned slot, start timing only for verified correct occupation, report a dummy invoice and admit paid exit. These are specified expected outcomes, not demonstrated numerical results.

The system's principal strength is linking sensing, identity, reservation and payment into a traceable transaction. Its main limitations are OCR sensitivity to miniature plates, ultrasonic inability to identify vehicles, Wi-Fi dependence and simulated rather than live financial settlement. Deployment in a real parking area would additionally require durable hardware, fail-safe safety sensors, cybersecurity review and independently validated recognition accuracy.

VIII. RESULTS

The given picture shows the hardware implementation of an Automatic Smart Parking Management System. It is a miniature prototype developed to demonstrate automated vehicle parking and slot monitoring.

The system consists of a small parking structure, a robotic vehicle, sensors, electronic components, and a power supply. Ultrasonic sensors are placed near the parking slots to detect vehicle presence and identify available parking spaces. The sensors send information to the microcontroller, which processes the data and updates the parking status.

A 4WD robotic vehicle represents the customer vehicle and demonstrates movement within the parking area. The electronic control circuit manages sensor operations and vehicle movement. LEDs are used to indicate the system status, while the power supply provides the required electrical energy.

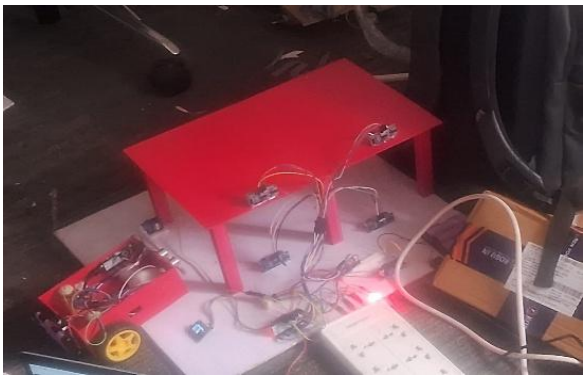


Fig 7: Input result

When a vehicle approaches the parking area, the sensors check the availability of parking slots. The microcontroller processes the sensor readings and identifies vacant and occupied slots. Based on the availability, the system helps guide the vehicle towards the assigned parking space. The vehicle movement and parking status are monitored through the control system. The ultrasonic sensors continuously monitor each parking slot and send updated information to the microcontroller. The available parking details are displayed on the web application, allowing users to identify free spaces easily. Once the vehicle reaches the assigned slot, the system confirms its parking position and updates the slot status as occupied. The system also records the parking time and maintains vehicle parking details.

This automated process reduces manual monitoring, saves time, and improves the overall efficiency of parking management.

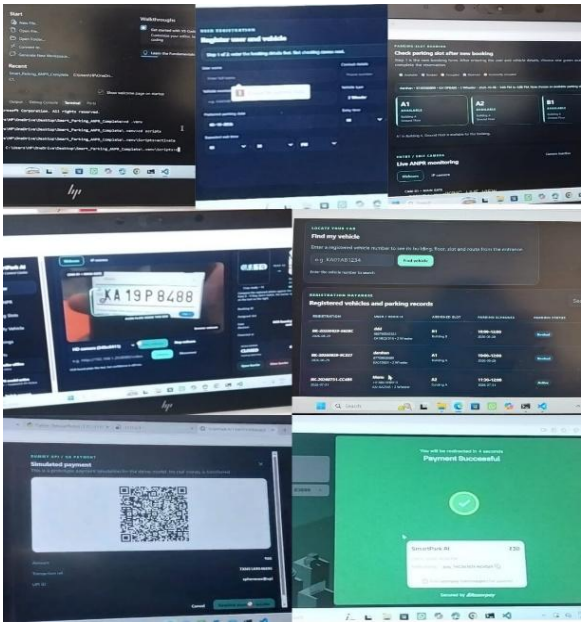


Fig 8: Output Result

The Automatic Smart Parking Management System works through a web application, ultrasonic sensors, a webcam, an ESP32 microcontroller, and a robotic vehicle. Initially, the user registers vehicle details and books an available parking slot through the web application. At the entrance, the webcam captures the vehicle number plate, and OCR verifies it with the registered details. If the verification is successful, the servo motor opens the boom barrier.

Ultrasonic sensors detect vacant and occupied slots, while the robotic vehicle moves towards the assigned parking space. The system records parking duration, calculates charges, and generates a QR code for payment. After successful payment confirmation, the exit barrier opens automatically.

The web application displays real-time parking availability, vehicle details, and booking records. It also helps users locate their parked vehicles and monitor payment status. Overall, the system reduces manual effort, saves time, and improves parking management efficiency.

IX. CONCLUSION

The expected demonstration can be judged at the level of state consistency rather than a single sensor output: a bay must not be simultaneously available and occupied; a vehicle with the correct OCR string but no valid booking must not pass; a paid record must be tied to the correct booking; and a cleared gate must not prematurely free an occupied bay. These conditions define actionable software invariants for acceptance testing. They can be checked using deliberately conflicting bookings, stale OCR strings, network interruptions, and wrong-slot placements.

A quotation-grounded smart parking prototype has been presented with an implementable controller schematic, 4WD drive interface and end-to-end transaction workflow. Four ultrasonic-monitored bays, webcam-based number plate recognition, web booking/navigation and a payment-dependent exit gate together provide a credible integrated demonstration. Future work should report measured ANPR accuracy, occupancy false detection rates, barrier/robot response times and fault-recovery data before claiming validated performance.

X. FUTURE SCOPE

The Automatic Smart Parking Management System can be further improved by integrating advanced technologies such as Artificial Intelligence (AI), Machine Learning, and cloud computing. In the future, a mobile application can be developed to make parking slot booking and vehicle tracking easier. Advanced number plate recognition can improve vehicle identification and security. The system can also be extended to support multi-level parking facilities with real-time monitoring.

Integration of secure online payment gateways can make parking fee collection fully automatic. CCTV surveillance and smart notifications can provide better security and user convenience. Solar power can also be used to reduce energy consumption. Overall, these improvements can make the system more intelligent, reliable, secure, and suitable for large parking areas such as shopping malls, hospitals, colleges, and commercial buildings.

Additionally, the system can be connected with smart city infrastructure to manage multiple parking locations efficiently. Future upgrades can also include automatic fault detection and system maintenance alerts to improve reliability and performance.

REFERENCES

[1] B. Rajalakshmi, G. Sudha, S. Surya, H. R. M., J. A., and M. M. S. G., "IoT Smart Parking with Android App," in *Proc. 2025 International Conference on Computing and Communication Technologies (ICCCT)*, Chennai, India, 2025, pp. 1–6, doi: 10.1109/ICCCT63501.2025.11019695.

[2] D. C., S. B., V. Vijayan, D. R., K. J., and D. P. E., "Sensor-Based Energy Efficient Intelligent Parking System," in *Proc. 7th International Conference on Inventive Material Science and Applications (ICIMA)*, Namakkal, India, 2025, pp. 1043–1047, doi: 10.1109/ICIMA64861.2025.11073870.

[3] J. Kulkarni, S. Tambe, S. Deshmane, and P. Vasmatkar, "Smart Automated and Real-Time Parking Management System," in *Proc. 7th International Conference on Inventive Material Science and Applications (ICIMA)*, Namakkal, India, 2025, pp. 928–932, doi: 10.1109/ICIMA64861.2025.11074057.

[4] J. S., G. Chandrasekaran, M. S. G., Y. P., V. M., and S. M. S., "IntelliPark—Intelligent IoT-Based Parking System," in *Proc. 7th International Conference on Inventive Material Science and Applications (ICIMA)*, Namakkal, India, 2025, pp. 895–901, doi: 10.1109/ICIMA64861.2025.11073875.

[5] A. Ampuni, S. Fonataba, A. Fitrianto, and G. Wang, "Smart Parking System with Automatic Cashier Machine Utilize the IoT Technology," in *Proc. International Conference on ICT for Smart Society (ICISS)*, Bandung, Indonesia, 2019, pp. 1–4, doi: 10.1109/ICISS48059.2019.8969793.

[6] Suhartono, J.-L. Chiang, M. M. Sakanti, and C. B. D. Kuncoro, "Smart Residential Parking," in *Proc. 7th International Conference on Informatics and Computational Sciences (ICICoS)*, Semarang, Indonesia, 2024, pp. 438–443, doi: 10.1109/ICICoS62600.2024.10636897.

[7] T. Luong, Y. Shin, J. Park, S. Leeu, and H. Moon, "AI-Based Free Parking Space Detection System for Large Parking Areas," in *Proc. IEEE International Conference on Real-Time Computing and Robotics (RCAR)*, Toyama, Japan, 2025, pp. 1072–1077, doi: 10.1109/RCAR65431.2025.11139447.