© June 2026 | IJIRT | Volume 13 Issue 1 | ISSN: 2349-6002

From Circling to Parking: A Low-Cost Wi-Fi Smart Parking System That Guides Drivers to Open Spots in Real Time

Mohamed Nafeez M[1], Dr. P. Sivasankar[2]

[1]*Electronics and Communication Engineering Puducherry Technological University*

[2]*Professor Electronics and Communication Engineering NITTTR Chenna*

***Abstract*—Urban drivers do not find time and slots for their car parking in busy areas like malls, IT offices etc. To avoid this error the parking mechanism with automatic tracking of parking slots with the help of IoT system have been implemented. For this system three layers of IoT is used first the perception layer where it involves the sensors which sense the data, next the transport layer where the sensed data passes with the help of Wifi communication protocol and the HTTP POST with the backend data, then comes the physical layer which involves the node backend and the react frontend with the express.js framework. The Application layer also involves the EC2 instance in Amazon Cloud Service platform. This demo helps to analyse the parking efficiency and fuel maintenance of the vehicles in the parking systems. It is the cost-effective system built with the ESP8266, the Hybrid sensors and the LCD 16x2 display (for advertising).**

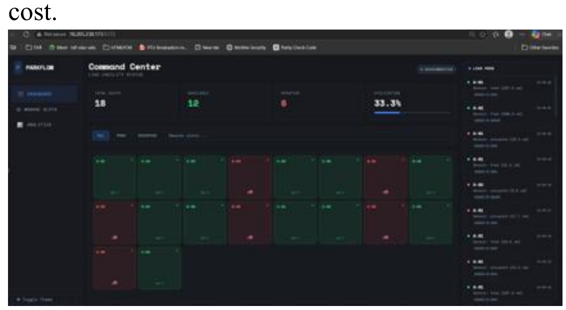
***Index Terms*—IoT, Smart Parking, Hybrid Sensors, AWS EC2, ESP8266, Wi-Fi**

I. THE CORE CHALLENGE: SEARCH TIME, NOT JUST AVAILABILITY

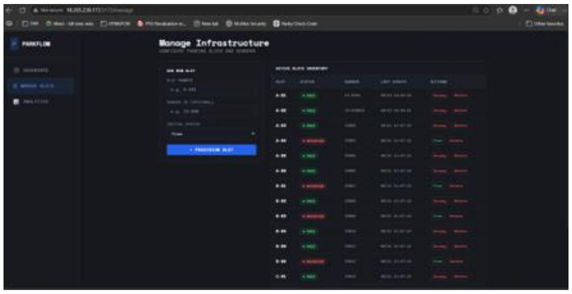
Many people assume that the biggest challenge in parking is the lack of available spaces. However, the actual problem is difficulty in finding the parking slots. Drivers usually do not know where free parking spots are until they reach the parking area. Studies have shown that drivers in large cities spend an average of around several minutes in searching of parking during a single trip. This unnecessary searching contributes to heavy traffic in roads, with some reports suggesting that nearly 30% of traffic is caused by vehicles circling in search of parking. As a result, fuel consumption increases, carbon emissions rise, and wastage of time. Traditional parking management methods, such as human attendants, signboards, and entry gate systems, mainly respond after a vehicle arrives at the parking facility. These systems provide little assistance to drivers before they enter the parking area, often forcing them to drive through the same rows multiple times looking for an empty space.

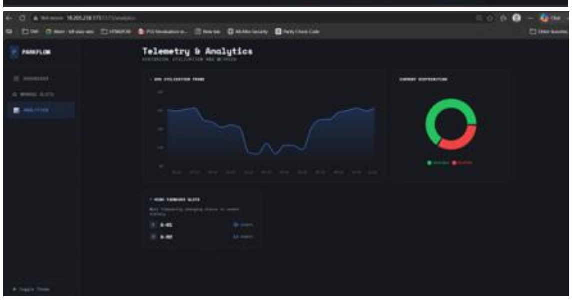
Use of this distributed sensor offers a more effective solution. Small sensors installed in parking slots can continuously monitor occupancy and send real-time information to a central system. Drivers can then check available spaces through a mobile application or web interface and proceed directly to a free slot. This reduces unnecessary driving, lowers traffic congestion, and helps decrease fuel usage and environmental impact.

This concept tells that the system uses affordable and widely available hardware components, including an ESP8266 Wi-Fi microcontroller, an HC-SR04 ultrasonic sensor, and a compact LCD display. A single unit can monitor up to four parking spaces, transmit occupancy information through Wi-Fi, and display the number of available slots. The estimated cost of each node is affordable, making the system suitable for large-scale deployment at a relatively low cost.



Several smart parking solutions have been proposed in previous studies, using technologies such as infrared sensors, magnetic detectors, cameras, and ultrasonic sensors. Among these options, ultrasonic sensors are particularly suitable for indoor and covered parking environments because they are not affected by lighting conditions. They are also considerably less expensive than camera-based systems while still providing reliable detection accuracy





This paper presents three main contributions:

(1) The complete design and implementation of the ParkFlow system, including hardware, firmware, and server components.

(2) An evaluation of system accuracy and response time through experimental testing.

(3) A discussion of possible future improvements, such as communication protocol enhancements and edge-based analytics.

II. DESIGN PHILOSOPHY AND SYSTEM ARCHITECTURE

2.1 Three-Layer IoT Architecture

ParkFlow implements the ITU-T three-layer IoT model, separating sensing, transport, and application logic. This modularity simplifies testing, updates, and maintenance.

Layer	Function	Components
Sense	Measure distance, detect vehicles	HC-SR04 sensors, ESP8266 reads echo pulse
Transport	Transmit data over Wi-Fi	ESP8266 joins 2.4 GHz network, sends JSON via HTTP POST
Application	Store, process, display data	Server API, database, dashboard endpoints, 16×2 LCD

Layer independence allows server changes without firmware modifications, or sensor replacements without network code revisions.

2.2 Occupancy Detection Logic

Component	Role	Key Specifications
ESP8266 NodeMCU	Wi Fi microcontroller	80 MHz, 96 KB RAM, 4 MB flash
HC SR04	Ultrasonic distance sensor	2–400 cm range, ±3 mm accuracy
16×2 LCD (I2C)	Local display	I2C address 0x27
USB / 5 V supply	Power	3.3 V for ESP, 5 V for sensors
Wi Fi	Wireless network	2.4 GHz, 802.11 b/g/n

The HC-SR04 emits a 40 kHz sound burst upon receiving a 10microsecond trigger pulse. Sound travels to the nearest object and returns. The echo pin remains high during the return trip; pulse duration matches round-trip time.

Distance calculation:

$$\text{distance} = \frac{\text{travel time} \times \text{speed of sound}}{2} \quad (1)$$

2.3 Microcontroller Selection: ESP8266EX

The ESP8266EX integrates:

- 32-bit processor at 80 MHz
- 96 KB RAM, 4 MB flash
- Full Wi-Fi radio with TCP/IP support
- Arduino IDE compatibility
- Direct GPIO connection to sensor trigger/echo lines
- Built-in Wi-Fi stack (network association, IP assignment, HTTP sessions)

This integration will reduce cost, board size, and wiring complexity-ideal for cheap, compact, connected nodes.

2.4 Node Operation Loop

Each node executes five recurring steps:

1. Sense: Trigger each HC-SR04 sequentially, measure echo pulse duration
2. Decide: It converts echo time to distance, compare against threshold, apply majority-vote smoothing
3. Pack: Construct JSON with node ID, slot states (free/occupied), timestamp
4. Send: POST JSON to server API via Wi-Fi
5. Show: Update LCD with "free / total" count

Wi-Fi drops trigger exponential backoff retries. Failed HTTP requests queue messages for resending upon connection recovery, preventing data loss during brief network gaps.

2.5 Wiring Configuration

Each sensor requires two GPIO pins (trigger output, echo input):

- Sensor 1: trigger on D1, echo on D2
- Sensor 2: trigger on D5, echo on D6
- LCD (I2C): SDA to D3, SCL to D4

All sensors share the 5 V rail. The ESP8266 operates at 3.3 V via its internal regulator. This minimal wiring avoids extra components and maintains compact board design.

III. EXPERIMENTAL RESULTS AND ANALYSIS

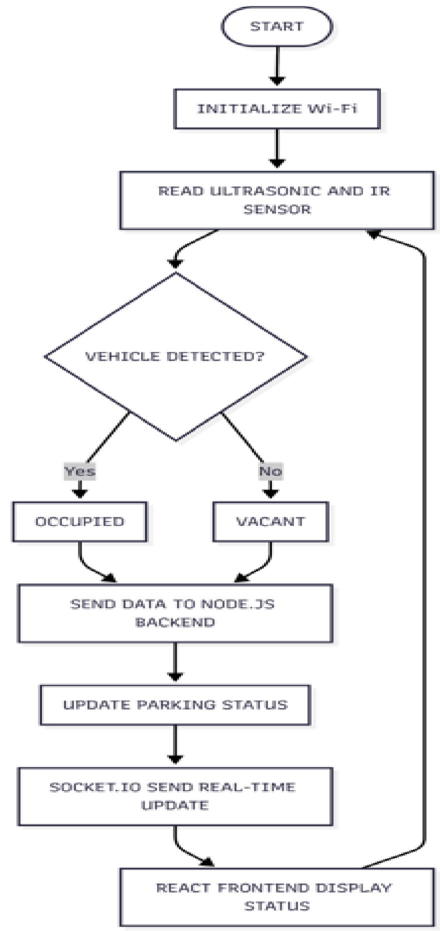
3.1 Comparative Analysis

Aspect	ParkFlow	RFID	Camera
Vehicle HW	None	Tag/car	None
Cost/slot	~$10	$30–50+	$100+
Load	Minimal	Low	High
Light Dep.	No	No	Yes
Feedback	LCD	None	App

ParkFlow eliminates vehicle-tag adoption barriers and avoids expensive image processing hardware/software. The local LCD provides immediate feedback even during server downtime a feature many competing systems lack.

HTTP was selected for development simplicity. In large production systems, MQTT would reduce message size by minimum percentage and maintain persistent connections, improving efficiency and reducing latency.

3.2 Flowchart



3.3 Scalability Considerations

One node supports up to four slots. For larger zones, multiple nodes connect to a zone controller that aggregates data before forwarding to the central server. Adding sensors to a single node requires an I2C GPIO expander (e.g., PCF8574) to expand pins without board redesign.

Security: Current implementation uses plain HTTP. Production deployments should adopt HTTPS with TLS encryption and require API keys/tokens for authentication, preventing unauthorized devices from sending fake data and protecting user privacy.

Wi-Fi congestion: In dense urban areas, hundreds of devices share the 2.4 GHz band. Large installations require careful channel planning or migration to Wi-Fi 6 (802.11ax) for stable performance.

3.4 Recommended Upgrades

Upgrade	Benefit
Switch to MQTT	Smaller messages, persistent connections reduce latency and power consumption
On-device analytics	Simple filtering/trend detection on ESP8266 reduces server message volume
Predictive availability	Historical data trains models predicting slot release times, helping drivers choose zones with higher success rates
Solar power	Small solar panel + ESP8266 deep-sleep modes enable wireless-powered outdoor nodes
City data integration	Connect to municipal open-data platforms for optimized pricing, capacity, and signage across the city
Extra sensors	Combine ultrasonic with infrared/magnetic sensors for improved reliability in edge cases
Mobile app	Real-time spot maps guiding drivers to nearest free slot

IV. CONCLUSIONS AND FUTURE WORK

The development of ParkFlow shows that a smart parking system can be implemented using low-cost and easily available hardware components. The system uses a Node MCU (ESP8266) microcontroller and an HC-SR04 ultrasonic sensor which has the high accuracy to detect the availability of parking spaces and provide updates to users in real time. During testing, the system demonstrated reliable performance with high detection accuracy and low response time.

A key advantage of the proposed system is its affordable, it can be deployed in large parking areas without requiring a significant investment. The LCD display also adds practical value by showing the number of available parking spaces even when the network connection is temporarily unavailable.

The results indicate that the proposed approach can help reduce the time spent searching for parking spaces and improve overall parking management. In addition, the modular architecture makes it easier to maintain and upgrade the system when new features are required.

In the future, the system can be further improved by adopting MQTT communication, integrating edge analytics, and exploring solar-powered operation for outdoor deployments. Testing the system in larger real-world parking environments will also provide a better understanding of its performance and scalability.

REFERENCES

[1] D. Shoup, The High Cost of Free Parking. American Planning Association; 2005.

[2] A. Zanella, N. Bui, A. Castellani, L. Vangelista, and M. Zorzi, "Internet of Things for Smart Cities," IEEE Internet of Things Journal, vol. 1, no. 1, pp. 22–32, 2014.

[3] S.-Y. Chou, S.-W. Lin, and C.-Y. Li, "Parking Lot Occupancy Detection Using Infrared Sensors," IEEE Internet of Things Journal, vol. 4, no. 2, pp. 480–491, 2017.

[4] M. Faheem, S. A. Mahmud, G. M. Khan, M. Rahman, and H. Zafar, "Smart Parking Systems and Sensors: A Survey," in Proceedings of the IEEE Student Conference on Research and Development (SCOReD), 2013.

[5] S. Valipour, M. Siam, M. Jagersand, and N. Ray, "Parking Lot Occupancy Using Long Short-Term Memory," in Proceedings of the IEEE International Conference on Image Processing (ICIP), 2016.

[6] V. Bharathi, and G. Vasudevan, "Smart Parking System Using IoT Infrastructure," International Journal of Engineering Research and Technology, vol. 6, no. 5, pp. 411–414, 2017.

[7] ITU-T, Overview of the Internet of Things, Recommendation Y.2060, International Telecommunication Union, 2012.

[8] Espressif Systems, ESP8266EX Datasheet. Shanghai, China: Espressif Systems; 2015.

[9] H. Al-Kharusi, and I. Al-Bahadly, "Intelligent Parking Management System Based on Image Processing," World Journal of Engineering and Technology, vol. 2, no. 2, pp. 55–67, 2014.

[10] S. Lim, and V. Rao, "A Review of Smart Parking Solutions," Transportation Research Part C: Emerging Technologies, vol. 96, pp. 334–353, 2018.

[11] B. Mishra, and A. Kertesz, "The Use of MQTT in M2M and IoT Systems," IEEE Access, vol. 8, pp. 201071–201086, 2020.