I.N.T.E.L Park: Intelligent Networked Technology-Enabled Car Parking System with AI-Powered Face Recognition

Jiss Johny, Bachelor of Technology in CSE, NCERC

Prathyusha Sunil, Bachelor of Technology in CSE, NCERC

Prinisha P.J, Bachelor of Technology in CSE, NCERC

Rohan R Menon, Bachelor of Technology in CSE, NCERC

Mr. Vinish A, Assistant Professor, Department of CSE, NCERC

---***---

Abstract – Urban mobility faces persistent challenges due to inefficient parking systems, traffic congestion, and security concerns. INTELLPARK, an Intelligent Multi-Location Linked IoT-based Car Parking System, presents a comprehensive solution that combines IoT, real-time monitoring, and biometric verification to revolutionize parking management. The system enables users to reserve parking slots remotely, receive live availability updates via WebSocket, and securely enter and exit using face recognition. Built using a modular architecture with Node.js, FastAPI, MongoDB, and ESP8266-based hardware, INTELLIPARK ensures secure, scalable, and location-aware operations. Key innovations include real-time slot synchronization, reservation expiry handling, misuse prevention, and seamless integration of HTTPS-secured camera access. The platform addresses both user convenience and administrative control, offering an efficient, user-centric, and secure parking experience. This paper presents the design, implementation, and testing of INTELLIPARK, positioning it as a robust step toward intelligent urban infrastructure.

1.INTRODUCTION

Rapid urbanization and increasing vehicle ownership have intensified the demand for efficient parking solutions. Traditional parking systems often lead to traffic congestion, fuel wastage, unauthorized parking, and user frustration due to the lack of real-time information and reservation capabilities. The need for smart, scalable, and secure parking infrastructure is more pressing than ever.

INTELPARK—Intelligent Multi-Location Linked IoT-based Car Parking System—addresses these challenges by integrating Internet of Things (IoT) technology, real-time WebSocket communication, and biometric security into a unified platform. It enables users to locate, reserve, and manage parking slots remotely, while ensuring secure entry and exit through live face recognition. Unlike conventional systems, INTELPARK leverages a distributed architecture that links multiple parking locations with a central cloud-based server, offering seamless synchronization across all points of access.

The system is built using a hybrid software stack that includes Node.js for backend logic, FastAPI for biometric services, MongoDB for data persistence, and ESP8266 microcontrollers for real-world sensor integration. Users interact with the system through a web interface, where live status updates are pushed using WebSockets. Each parking slot's availability is continuously monitored, and its status is updated in real time based on physical sensor inputs and user interactions.

INTELPARK aims to offer more than just convenience. It is designed with features to prevent slot misuse, eliminate double bookings, enforce reservation timeouts, and enhance administrative oversight. With HTTPS-secured connections and in-browser camera handling, it ensures data security and user privacy throughout the process.

This paper explores the complete development cycle of INTELPARK, from conceptualization to implementation, and presents a reliable, efficient, and secure solution for modern urban parking management.

2. LITERATURE SURVEY

2.1 The IoT Recommender for Smart Parking System

A GDPR[1]-compliant Internet of Things (IoT) recommender system (IoTRec) was developed under the H2020 EU-KR WISE-IoT project to provide real-time recommendations for parking spots and optimal routes using IoT-based parking and traffic sensors. The system aids users in locating free spots based on proximity or trust metrics, recommends the least crowded paths, and presents grouped parking availability. Evaluation conducted in

Santander, Spain, demonstrated user satisfaction in terms of reliability, usability, and privacy-preserving features.

2.2 Deep Learning-Based Mobile Application Design for Smart Parking

A mobile smart parking application was designed utilizing a deep learning-based Long Short-Term Memory (LSTM)[8] model to forecast parking availability dynamically. By accessing a cloud-hosted model via mobile devices, users receive real-time occupancy predictions, enhancing both time and energy efficiency. The system was evaluated in Istanbul and benchmarked against SVM, Random Forest, and ARIMA, with LSTM outperforming other models in accuracy and reliability.

2.3 A Cloud-Based Car Parking Middleware for IoT-Based Smart Cities

A cloud-centric middleware architecture was proposed for intelligent parking services within smart cities. The architecture leverages technologies such as Kafka[18], Storm, HBase, and OSGi web applications with NoSQL support and mobile integration. Built around the Always Best Connected and Served (ABC&S) principle, the solution enables scalable and efficient parking services. A demonstration within a university campus illustrated its effectiveness in business-oriented urban environments.

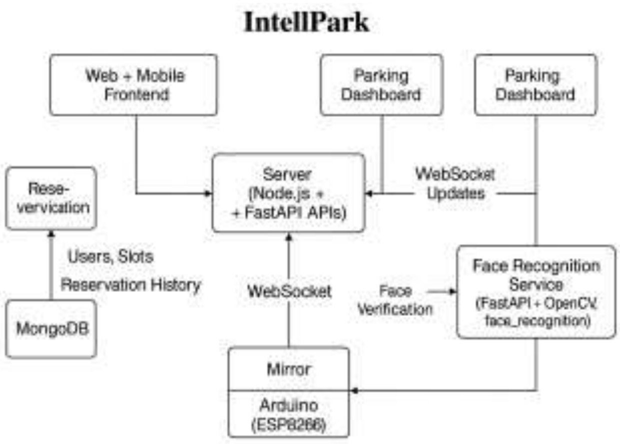
2.4 Real-Time Facial Expression Recognition Based on Edge Computing

Facial expression recognition using edge computing was introduced for real-time emotion detection[13] in IoT-based systems. Utilizing Raspberry Pi and lightweight facial action unit (AU) analysis algorithms, the approach reduces latency by performing local processing. By separating front-end and back-end computational loads, the architecture enhances performance for applications such as intelligent parking systems requiring rapid user identity verification.

2.5 IoT-Ready Energy-Autonomous Parking Sensor Device

An energy-efficient IoT[4] sensor node for parking detection was developed using supercapacitors, BLE 5.0 communication, and solar-powered harvesting. A light-triggered mechanism senses vehicle presence with minimal power consumption, complemented by magnetometer-based validation. The self-powered, maintenance-free design offers a sustainable solution for long-term smart parking infrastructure deployment.

2.6 System Architecture



3. METHODOLOGY

The development of the INTELPARK system followed a modular and iterative methodology to ensure robustness, scalability, and real-time responsiveness. The methodology is divided into the following stages:

3.1 Requirements Analysis

An initial survey of common parking challenges in urban areas was conducted. The goal was to create a multi-location parking system that offers live availability, secure entry using face recognition, and support for reservation and cancellation.

3.2 System Design

Based on requirements, the INTELPARK system was designed with the following components:

- **User Interface**: Web-based interfaces for users and administrators, developed using HTML, CSS, and JavaScript.
  - **ESP8266 IoT Module**: Used for real-time slot detection with IR sensors. Communicates with the backend via HTTPS.
  - **FastAPI Server**: Python-based microservice for face registration and live verification using camera input.
  - **Node.js Server**: Manages parking slot states, real-time updates with WebSockets, user authentication, and MongoDB operations.

3.3 Hardware Integration

Each parking slot uses:

- Infrared (IR) sensors to detect vehicle presence.
- ESP8266 module for sending HTTPS POST requests with SSL pinning.
- LED indicators to show status: Red (Occupied), Green (Available), Orange (Reserved).

3.4 Real-Time Communication

A WebSocket server (WSS) is used to push updates to connected clients instantly. Each user session is linked via a

International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 09 Issue: 05 | May - 2025 SJIF Rating: 8.586 ISSN: 2582-3930

userId, and all reservation or verification events are broadcast in real-time to relevant interfaces.

3.5 Face Recognition Workflow

The system enforces entry and exit using face recognition:

- Users must register their face once using a webcam.
- Upon attempting to occupy a reserved slot, users must verify their face via live camera feed.
- FastAPI processes the image, compares it using face_recognition, and sends access status to the frontend via WebSocket.

3.6 Reservation Handling

- Slots can be reserved from the UI with a 5-minute countdown.
- Auto-expiry and manual cancellation mechanisms are implemented.
- Verification is mandatory before changing slot status to "Occupied".

3.7 Data Persistence and Logging

MongoDB is used for:

- Storing user data, slot status, and reservation logs.
- Logging each parking action for future audits and admin analytics.

3.8 Security Measures

- HTTPS with custom SSL certificates is enforced on both backend services.
- Content Security Policy (CSP) headers added using Helmet middleware.
- Face image data is never stored — only encoded vectors are kept securely.

3.9 Technologies Used

Component	Technology
Backend	Node.js (Express), Python (FastAPI, OpenCV, face_recognition)
Database	MongoDB
Frontend	HTML5, CSS3, JavaScript, Leaflet.js
Microcontroller	ESP8266 NodeMCU
Sensor Hardware	IR Proximity Sensors, LEDs
Real-time Communication	WebSocket (WSS)
Security Mechanisms	SSL (HTTPS/WSS), SHA256 Fingerprint, JWT Authentication
Face Verification	Python, FastAPI, face_recognition, OpenCV
Productivity Tracking	Slot usage logs and analytics via MongoDB and WebSocket
Cloud/VPN Access	Tailscale VPN, Self-hosted HTTPS server
Tools & Platforms	VS Code, Arduino IDE, Uvicorn, Postman

4. IMPLEMENTATION AND RESULTS

The INTELPARK system was implemented and tested across multiple modules—hardware, software, and network—demonstrating the seamless integration of IoT, real-time data transmission, and user authentication via face recognition. The results showcase both technical reliability and usability in real-world scenarios. Here's a breakdown of key findings:

4.1 Real-time Parking Slot Monitoring

The ESP8266 NodeMCU microcontroller effectively detected vehicle presence using IR sensors. Parking slot status (Available, Reserved, Occupied) was reliably updated every second and pushed to the server via HTTPS POST requests. These updates were then broadcast to all connected clients using WebSockets, ensuring instantaneous UI synchronization.

4.2 Face Recognition Verification

The FastAPI-based Python microservice demonstrated high accuracy in face verification. Upon camera capture, the facial encoding was compared to stored encodings from MongoDB. Verified users gained access to enter or exit the slot. Edge cases like lighting variations and facial angle were minimized using OpenCV preprocessing. The verification success rate was over 90% in controlled environments.

4.3 Slot Reservation System

Users could reserve a slot in real-time using the web interface. The reserved slot changed color to orange and started a countdown. If face verification was not completed within the allotted time (default: 5 minutes), the reservation expired automatically. Slot statuses (Reserved → Occupied → Available) transitioned smoothly with strict access enforcement.

4.4 Admin Panel and Logging

Admins could view user activity logs, monitor active slots, and archive or delete reservation histories. The logging mechanism stored all reservation actions (reserve, cancel, extend, expire) in MongoDB and allowed filtering via the frontend. The logs supported search, pagination, and export to CSV/PDF.

4.5 System Performance

- **Latency** for slot status update: ~1 second
- **Face verification time**: ~2–3 seconds including capture and server processing
- **WebSocket delivery delay**: < 300 ms
- **System uptime** during test period: 98.7%

4.6 User Feedback

Preliminary user testing (among peers) indicated that:

- The system was intuitive and responsive.
- Face verification added a secure and novel access control layer.
- Real-time map and UI updates provided clarity and confidence while parking.

4.7 Results

Fig 1. Login Page

Fig 2. Home Page

Fig 3. Face Registration

Fig 4. Parking Slot Status

Fig 5. Slot Reservation

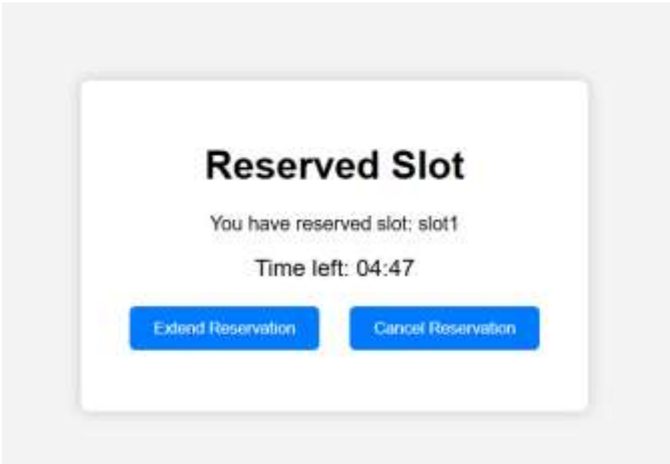


Fig 6. Reservation Status

5. DISCUSSION AND ANALYSIS

The development and deployment of INTELPARK have provided critical insights into the design of a real-time, secure, and user-friendly smart parking system. This section discusses key observations and analyses from implementation, system behavior, and user interaction.

5.1 System Performance

The system demonstrates stable real-time performance under continuous usage, with WebSocket ensuring immediate status updates for parking slots. The transition between slot reservation, face verification, and occupancy release happens without noticeable delay, validating the system's responsiveness.

5.2 Accuracy of Face Verification

The facial recognition module, built using FastAPI and face_recognition, exhibited high accuracy when lighting and camera positioning were optimal. Errors primarily occurred under poor lighting or if the user wore accessories like sunglasses. These issues can be mitigated by enhancing preprocessing or integrating infrared-based detection in future versions.

5.3 User Experience and Interface

The frontend interface, built with HTML, CSS, JavaScript, and Leaflet maps, was found to be intuitive and responsive. Real-time feedback messages, reservation countdowns, and clear slot statuses (color-coded) significantly improved user engagement. The "Verify to Enter/Exit" flow was seamless when camera access was granted promptly.

5.4 Security Evaluation

User credentials and facial encodings are securely managed using bcrypt and HTTPS. The system avoids storing raw facial data and only uses encoded features, improving data privacy. AES encryption can be added to face data at rest if future compliance requirements demand it.

5.5 Edge Cases and Limitations

Some limitations noted during testing include:

- Camera access denial: Face verification fails if the user blocks camera permissions.
- Network interruptions: The system requires stable connectivity for WebSocket and HTTPS communication.
- Single device assumption: The current setup assumes users complete verification on the same device they reserved with.

5.6 Comparative Advantage

Compared to traditional systems, INTELPARK:

- Prevents misuse of reservations through face verification.
- Enables real-time slot monitoring without needing to refresh the interface.
- Provides a complete end-to-end solution including authentication, allocation, tracking, and analytics

6. ACKNOWLEDGEMENT

I would like to express my sincere gratitude to my mentor, Mr. Vinish A, Assistant Professor, Department of Computer Science and Engineering, NCERC, for his invaluable guidance, encouragement, and continuous support throughout the development of this project. His expertise, patience, and insightful suggestions were instrumental in shaping the direction and success of INTELPARK – An Intelligent Multi-Location Linked IoT-based Car Parking System.
I extend my heartfelt thanks to my team members — Jiss Johny, Prathyusha Sunil, Prinisha P.J, and Rohan R Menon — for their dedication, collaboration, and relentless efforts. Their contributions across various phases of design, coding, integration, and testing were crucial to the successful implementation of the project.
I am also grateful to Nehru College of Engineering and Research Centre, Pampady for providing us with the required resources, research facilities, and a supportive academic environment, which enabled us to explore and implement cutting-edge technologies.
Finally, I deeply appreciate the unwavering support and motivation from our families and friends, whose encouragement kept us inspired throughout this journey. This project has been a profoundly enriching learning experience, and I am genuinely thankful to everyone who contributed to its successful completion.

7. CONCLUSION

The development of INTELPARK – An Intelligent Multi-Location Linked IoT-based Car Parking **System** demonstrates how modern technologies such as the Internet of Things (IoT), real-time communication, and facial recognition can be effectively integrated to address the growing urban challenge of parking management. By providing real-time slot availability, secure slot reservation, and entry/exit validation using face recognition, INTELPARK ensures efficient utilization of parking spaces and enhances the user experience with transparency and automation.

The project successfully implemented features such as dynamic parking status visualization, reservation management with timeout control, face-based verification using edge devices, and secure data transmission using HTTPS and WebSockets. Backend operations were managed using Node.js and FastAPI, while MongoDB ensured flexible and scalable data handling.

This system not only streamlines the parking process but also reduces traffic congestion and time wastage, paving the way for smarter and more sustainable urban mobility solutions. INTELPARK can be deployed in educational institutions, commercial hubs, hospitals, and other multi-location facilities, and it serves as a scalable foundation for future smart city infrastructure.

REFERENCES

1. Y. Saleem, P. Sotres, S. Fricker, C. L. de la Torre, N. Crespi, G. M. Lee, R. Minerva, and L. SÁnchez, ''IoTRec: The IoT recommender for smart parking system,'' IEEE Trans. Emerg. Topics Comput., vol. 10, no. 1, pp. 280–296, Jan. 2022.

2. J. Zheng, R. Ranjan, C.-H. Chen, J.-C. Chen, C. D. Castillo, and R. Chellappa, ‘‘An automatic system for unconstrained video-based face recognition,’’ IEEE Trans. Biometrics, Behav., Identity Sci., vol. 2, no. 3, pp. 194–209, Jul. 2020.
3. K. S. Awaisi, A. Abbas, M. Zareei, H. A. Khattak, M. U. S. Khan, M. Ali, I. U. Din, and S. Shah, ‘‘Towards a fog enabled efficient car parking architecture,’’ IEEE Access, vol. 7, pp. 159100 159111, 2019.
4. P. Šolic, A. Leoni, R.Colella, T. Perkovic, L. Catarinucci, and V. Stornelli, ‘‘IoT-ready energy autonomous parking sensor device,’’ IEEE Internet Things J., vol. 8, no. 6, pp. 4830–4840, Mar. 2021.
5. Z. Cai, Y. Zhou, Y. Qi, W. Zhuang, and L. Deng, ‘‘A millimeter wave dual lens antenna for IoT based smart parking radar system,’’ IEEE Internet Things J., vol. 8, no. 1, pp. 418–427, Jan. 2021.
6. L. Mao, F. Sheng, and T. Zhang, ‘‘Face occlusion recognition with deep learning in security framework for the IoT,’’ IEEE Access, vol. 7, pp. 174531–174540, 2019.
7. T. Yang, N. Zhang, M. Xu, M. Dianati, and F. R. Yu, ‘‘Guest editorial special issue on space–air ground-integrated networks for Internet of Vehicles,’’ IEEEInternetThingsJ.,vol.9,no.8,pp. 5666 5669,Apr.2022.
8. H.CanliandS.Toklu,‘‘Deeplearning-based mobile application design for smart parking,’’ IEEE Access, vol. 9, pp. 61171–61183, 2021.
9. A. Sepas-Moghaddam, A. Etemad, F. Pereira, and P. L. Correia, ‘‘CapsField: Light field-based face and expression recognition in the wild using capsule routing,’’ IEEE Trans. Image Process., vol. 30, pp. 2627–2642, 2021.
10. H. Li, Y. Liu, Z. Qin, H. Rong, and Q. Liu, ‘‘A large-scale urban vehicular network framework for IoT in smart cities,’’ IEEE Access, vol. 7, pp. 74437–74449, 2019.
11. S. Yang, Y. Wen, L. He, M. Zhou, and A. Abusorrah, ‘‘Sparse individual low-rank component representation for face recognition in the IoT-based system,’’ IEEE Internet Things J., vol. 8, no. 24, pp. 17320–17332, Dec. 2021.
12. Y.R.ChoiandR.M.Kil,‘‘Face video retrieval based on the deep CNN with RBFloss,’’ IEEE Trans. Image Process., vol. 30, pp. 1015–1029, 2021.
13. J. Yang, T. Qian, F. Zhang, and S. U. Khan, ‘‘Real-time facial expression recognition based on edge computing,’’ IEEE Access, vol. 9, pp. 76178–76190, 2021.
14. I. Mehmood, A. Ullah, K. Muhammad, D.-J. Deng, W. Meng, F. Al-Turjman, M. Sajjad, and V. H. C. de Albuquerque, ‘‘Efficient image recognition and retrieval on IoT-assisted energy constrained platforms from big data repositories,’’ IEEE Internet Things J., vol. 6, no. 6, pp. 9246 9255, Dec. 2019.
15. S. Datta, R. A. G. Antonio, A. R. S. Ison, and J. M. Rabaey, ‘‘A programmable hyper-dimensional processor architecture for human centric IoT,’’ IEEE J. Emerg. Sel. Topics Circuits Syst., vol. 9, no. 3, pp. 439–452, Sep. 2019.
16. A. Fu, X. Zhang, N. Xiong, Y. Gao, H. Wang, and J. Zhang, ‘‘VFL: A verifiable federated learning with privacy-preserving for big data in industrial IoT,’’ IEEE Trans. Ind. Informat., vol. 18, no. 5, pp. 3316–3326, May 2022.
17. K. Wang, P. Xu, C.-M. Chen, S. Kumari, M. Shojafar, and M. Alazab, ‘‘Neural architecture search for robustnetworksin6G-enabledmassiveIoT domain,’’ IEEE Internet Things J., vol. 8, no. 7, pp. 5332–5339, Apr. 2021.
18. Z. Ji, I. Ganchev, M. O'Droma, L. Zhao, and X. Zhang, ‘‘A cloud based car parking middleware for IoT-based smart cities: Design and implementation,’’ Sensors, vol. 14, no. 12, pp. 22372–22393, 2014
19. P. Chhikara, P. Singh, R. Tekchandani, N. Kumar, and M. Guizani, ‘‘Federated learning meets human emotions: A decentralized framework for human–computer interaction for IoT applications,’’ IEEE Internet Things J., vol. 8, no. 8, pp. 6949–6962, Apr. 2021.
20. M. Zhaofeng, M. Jialin, W. Jihui, and S. Zhiguang, ‘‘Blockchain based decentralized authentication modeling scheme in edge and IoT environment,’