International Journal Software Engineering and Computer Science (IJSECS)
5 (2), 2025, 645-654
Published Online August 2025 in IJSECS (http://www.journal.lembagakita.org/index.php/ijsecs)
P-ISSN: 2776-4869, E-ISSN: 2776-3242. DOI: https://doi.org/10.35870/ijsecs.v5i2.4856.

RESEARCH ARTICLE **Open Access**

Mosque Parking System: An Implementation Using Camera and IoT

Lalu Ahmad Naufal Gina Gahara *
Department of Informatics, Faculty of Industrial Technology, Universitas Islam Indonesia, Sleman Regency, Special Region of Yogyakarta Province, Indonesia.
Corresponding Email: laluahmadnaufal003@gmail.com.

Kurniawan D. Irianto
Department of Informatics, Faculty of Industrial Technology, Universitas Islam Indonesia, Sleman Regency, Special Region of Yogyakarta Province, Indonesia.
Email: k.d.irianto@uii.ac.id.

Received: June 7, 2025; Accepted: July 10, 2025; Published: August 1, 2025.

Abstract: The rapid technological advancement has transformed various aspects of daily life, including parking management in mosque environments. Mosque parking administration often proves inefficient, particularly during religious holidays, leading to vehicle congestion, disorganized parking arrangements, and worshiper delays. This research develops an Internet of Things-based parking capacity calculation system using ESP32-CAM cameras that automatically detect vehicles. The system displays real-time information through a 20x4 I2C LCD and website interface, showing vehicle count, available slots, status (entry/exit), and timestamp data, with storage capabilities in Firebase. Designed using a Human-Centered Design approach, the system was implemented at Darussalam Ganjuran Mosque parking area and evaluated through black box testing. Results revealed 66.67% detection accuracy, successful real-time parking capacity information display, and effective vehicle history storage. Performance limitations included reduced accuracy in low lighting conditions, difficulty tracking high-speed vehicles, and camera resolution constraints. The system allows users to access current parking conditions while enabling mosque administrators to monitor parking areas efficiently.

Keywords: Internet of Things; Real-time; Mosque; Parking System; Firebase; ESP32-CAM; HCD.

1. Introduction

Information technology advances have transformed public services, shifting parking management from manual security oversight to computer-automated approaches [1]. This evolution matters particularly in mosques as centers for Islamic community life—spanning social, educational, and spiritual activities [4][11]—where visitor numbers spike during the five daily prayer times [5]. When worshippers arrive for prayers, especially during Friday congregations and religious holidays, parking areas quickly become overwhelmed. The search for available spaces often turns chaotic, with vehicles circling repeatedly or creating impromptu parking arrangements that block others. At Al-Ikhlas Mosque in Malang, research shows the parking area satisfies merely 65% of peak-hour demands [7], forcing many to park haphazardly along nearby streets. Similarly, surveys conducted at Al-Falah Mosque in Surabaya reveal that 72% of worshippers experience prayer delays because they spend precious minutes searching for parking [6]. Many arrive frustrated and distracted, diminishing the spiritual experience they seek.

Mosque administrators face equally challenging circumstances. Without accurate data on parking utilization, they struggle to manage traffic flow during peak periods. Security personnel often become overwhelmed directing vehicles, especially during special events when attendance multiplies. The absence of real-time space availability information triggers traffic jams extending beyond mosque grounds and encourages unauthorized parking in neighboring areas [9]. Environmental concerns also arise as vehicles circle continuously, releasing unnecessary emissions in residential neighborhoods [10]. Traditional parking management approaches rely heavily on human intervention—security guards manually directing traffic and worshippers communicating through word of mouth about available spaces. Such methods prove increasingly inadequate as mosque attendance grows and communities expand. The disconnect between available technology and current practices represents a missed opportunity for religious institutions to enhance service quality while reducing environmental impact. ESP32-CAM and IoT solutions offer an integrated answer to these challenges [11][12]. By leveraging affordable camera modules with machine vision capabilities, mosques can implement automated systems that detect and count vehicles while communicating availability to incoming worshippers. The technology promises to transform the arrival experience, allowing community members to focus on spiritual matters rather than parking logistics.

Standard parking systems fail to provide up-to-date information during prayer rushes [3][9], undermining efficient land use and disrupting worship activities [5][6]. When hundreds of worshippers converge simultaneously for prayers, the lack of organized parking information creates a bottleneck that affects the entire worship experience. Late arrivals disrupt prayer proceedings, while the stress of finding parking diminishes the sense of tranquility essential to religious practice. Our research develops an ESP32-CAM and IoT parking calculation system responding to the needs of both worshippers and mosque staff. The system aims to create harmony between technological capability and religious practice, enhancing rather than intruding upon the sacred atmosphere of worship spaces. By addressing practical concerns through unobtrusive technology, the system supports rather than distracts from spiritual activities.

We focus on three main goals: (1) Using ESP32-CAM with background subtraction [13] to detect vehicles in real-time through digital images [12]; (2) Building dual interfaces (LCD displays on-site and web dashboards) showing space data, availability, vehicle types, and timing information [4][10]; and (3) Using Firebase as a central database to map usage patterns [8]. The system aims to reduce parking search time by 70%, based on results from similar systems [1]. The technical implementation balances sophistication with affordability. ESP32-CAM modules offer remarkable capabilities at modest cost, making the system accessible even to mosques with limited budgets. Background subtraction algorithms enable reliable vehicle detection without requiring expensive hardware, while Firebase provides a scalable database solution that accommodates varying mosque sizes and attendance patterns.

The Human-Centered-Design (HCD) approach guides our work through five stages [11]: (1) Understanding users through direct observation of parking behaviors and talking with mosque managers and worshippers [5][7]; (2) Defining system requirements based on what we learned; (3) Brainstorming architectural designs using ESP32-CAM and IoT; (4) Building working prototypes with ESP32-CAM → Firebase → LCD and website connections using background subtraction [13] and (5) Testing the system against benchmarks of ≥90% detection accuracy and ≤3 second response times for car/motorcycle identification [3].

During the empathy phase, we spent several weeks observing parking patterns at Darussalam Ganjuran Mosque, noting peak arrival times, common frustrations, and worshipper behaviors. Interviews with mosque administrators revealed concerns about traffic management during special events, while conversations with regular attendees highlighted the stress associated with uncertain parking availability. These insights shaped our technical requirements and interface design decisions. Our system offers five key benefits: (1) Technical innovation through ESP32-CAM + Firebase integration for mosque parking [11][13]; (2) Better management through smoother parking flow and reduced congestion [9]; (3) Spiritual support by helping worshippers arrive on time [5][6]; (4) Environmental improvement by reducing emissions from circling vehicles [10]; (5) Academic value as a case study of HCD in religious facilities with measured results [2][11].

In everyday use, the system gives worshippers real-time information via LCD screens positioned at mosque entrances [4] . These displays show current capacity, available spaces, and recommended entry points, allowing drivers to make informed decisions before entering the parking area. Meanwhile, administrators monitor capacity trends through web interfaces, gaining valuable insights for future planning and immediate management decisions. The system respects privacy concerns by focusing on vehicle detection rather than identification. Cameras track movement patterns without capturing license plate information or identifying individuals, maintaining the anonymous nature of worship attendance while still providing useful aggregate data. As mosque communities continue growing alongside technological capabilities, such systems represent a thoughtful integration of innovation with tradition. The goal remains enhancing rather than replacing human interaction, using technology as a tool to foster community connection rather than diminish it.

2. Related Work

To ensure effective research implementation, we gathered data through literature reviews from relevant journals. This review aims to obtain supporting references for system planning and identify research problems. Based on analysis of previous studies, significant differences in component usage were found. To ensure system design compatibility, we compared these critical components as shown in the following table:

Table 1. Comparison of Previous Research

Study	Microcontroller	Camera Hardware	Software	Research Location
[18]	Arduino UNO	Webcam	Python, Open CV	Mechatronics Engineering and Automation Systems Laboratory, Mechanical Engineering Department PNUP and PNUP Postgraduate Research Laboratory
[19]	NodeMCU 8266, ESP32-CAM	ESP32-CAM	VCL Media Player, Arduino IDE	-
[20]	NodeMCU 8266, ESP32-CAM	ESP32-CAM type OV 2640	Fritzing, Arduino IDE, Visual Studio Code	Nurul Iman Mosque parking area, Huta Sidorejo
[21]	Arduino Mega 2560, ESP32-CAM	ESP32-CAM type OV 2640	Telegram, Web Server, Roboflow, OCR	-
[22]	ESP32	ESP32-CAM	Arduino IDE, Xampp, Visual Studio Code, Telegram, Web Browser	Prof. Dr. Hazairin, SH University

Table 1 shows various parts used in previous research. While some studies implemented ESP32-CAM, other utilized parts varied, as did the software types.

Table 2. Microcontroller Comparison

Model	Operating Voltage	Digital Pins	Analog Input Pins	Clock Speed	WiFi
Arduino Mega 2560	5V	54	16	16 MHz	None
Arduino UNO	5V	14	6	16 MHz	None
ESP32-CAM	5V	34	6	16 MHz	Yes
ESP32	3.3V	34	18	240 MHz	Yes
NodeMCU 8266	3.3V	11	1	80 MHz	Yes

Table 2 compares several microcontroller types frequently used in previous research related to Internet of Things systems. This comparison forms the basis for selecting ESP32-CAM as the main part in our research. ESP32-CAM offers advantages in pin count, Wi-Fi connectivity, and comes equipped with an internal camera module. Additionally, this module can operate with 5V voltage thanks to its built-in regulator, making it more practical and efficient for supporting visual, real-time vehicle detection systems.

Table 3. Testing with Vehicles

Model	Operating Voltage	Function	Connectivity
ESP32-CAM	5V	Surveillance, vehicle detection	Wi-Fi, Bluetooth
Webcam	5V	Face image display	USB

Table 3 shows a comparison of camera types used in previous research. Based on this comparison, our research selected ESP32-CAM because it integrates camera and Wi-Fi modules, making it more practical than webcams that require additional devices. The camera serves as a medium for visually detecting vehicles and transmitting data wirelessly, meeting the needs of the Internet of Things-based parking system developed in our research.

3. Research Method

This research employs a Human-Centered Design (HCD) approach implemented through three iterative phases – observation, design, and evaluation [23] – to ensure the developed system meets the needs of its users, namely mosque congregants and administrators. The system focuses on calculating parking capacity by utilizing ESP32-CAM cameras connected to the Internet of Things (IoT) within an edge computing architecture [23]. With the HCD approach, this research emphasizes deep understanding of users through direct observation, accurate problem formulation based on actual pain points, and iterative technology-based solution development. Specifically, this research prioritizes real-time vehicle capacity monitoring: ESP32-CAM cameras connected to the internet capture vehicle images, then this data is processed using image processing algorithms to calculate parking capacity. The capacity calculation results are then automatically transmitted via MQTT protocol, so accurate information can be delivered to users and mosque managers through integrated interfaces consisting of on-site LCD displays and web dashboards [23].

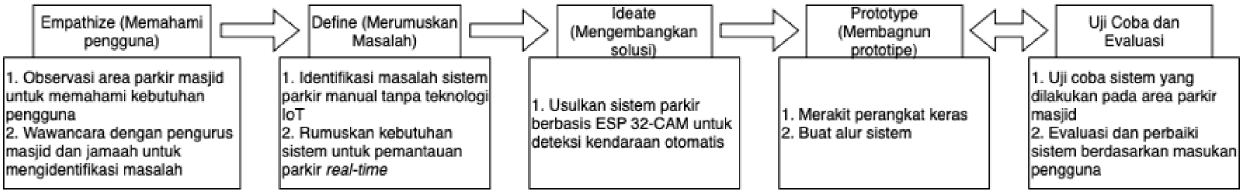

Figure 1. Design Flow.

3.1 Research Design

To ensure a structured research process, the following main stages were implemented:

1) Empathize (Understanding Users)
In the initial stage, observations were conducted in the mosque parking area to collect data about field conditions. Additionally, interviews were conducted with two respondent groups: mosque administrators and congregants. The main focus was identifying common problems, such as difficulty finding parking and minimal real-time capacity information.
2) Define (Formulating the Problem)
After initial data collection, the next step was formulating the core problem. From data analysis, it was found that the current parking system does not use integrated technology (such as IoT) to detect vehicles or manage parking capacity. This issue directed researchers to design an IoT-based system that can automatically detect vehicles and provide real-time parking capacity information.
3) Ideate (Developing Solutions)
This stage produced solutions based on the previous problem formulation. The solution is implementing an IoT-based parking capacity calculation system using ESP32-CAM cameras. This system is designed to automatically detect vehicles when entering and exiting the mosque parking area and count the number of parked vehicles. The aim of this solution is to improve parking management efficiency, provide convenience for congregants in accessing parking information, and assist mosque administrators in managing the parking area.
4) Prototype (System Prototype Development)
Based on the proposed solution, the next stage was building a system prototype. Several important points in prototype development include:
   a) Hardware Integration
   The core system component is the ESP32-CAM camera module, installed at parking entrance and exit points. The camera functions to capture vehicle images and connects to Wi-Fi to transmit image data. An ESP32-CAM MB microcontroller board is used to provide power and program the ESP32-CAM. A breadboard is used to arrange and connect electronic circuits. A 20x4 I2C LCD screen was added to display parking information to users. Several jumper cables are used to connect power and communication paths between these components.
   b) IoT Components and Software
   Managing communication between hardware and server. When the ESP32-CAM captures vehicle images, these images are sent through the IoT network to a central server. On the server side, a background subtraction algorithm is implemented using Python to detect moving objects (vehicles) in real-time [13]. This algorithm works by forming a static background model from several initial frames as reference. Each new frame is then compared with the background model using absolute differencing operations to identify moving objects (foreground). Next, morphological closing operations (dilation followed by erosion) are applied to reduce noise. Object contours are identified, and only contours meeting minimum area criteria for vehicles (cars/motorcycles) are validated [13].

Vehicle centroid positions are tracked to determine entry or exit status based on their movement relative to virtual lines at the parking entrance. Vehicle identification result data – including vehicle type, status (entry/exit), and time – is then stored in the Firebase database in real-time.

c) Data Presentation
Processed data is stored in the Firebase database in real-time. The web interface then retrieves this data to present parking capacity information directly. The website displays the number of vehicles, total vehicle slots, entry and exit times, vehicle status (entry/exit) which are all displayed in real-time, and there are options to manually add slots and vehicles. Meanwhile, the LCD displays concise information such as total vehicle slots (cars and motorcycles), detected vehicle type, vehicle movement status (entry/exit), and vehicle entry or exit time.

5) Testing and Evaluation
After prototype completion, testing and evaluation were conducted to ensure the system works properly. Trials were conducted in the mosque parking area with various image capture scenarios to verify detection constraints. Input from users (mosque administrators and congregants) was used as a guide for prototype refinement to provide reliable and timely parking capacity information.

In this research, components were selected according to system requirements and the advantages of each tool, then designed and arranged as follows:

Table 4. Components Used

No	Tool Name	Quantity
1	ESP32-CAM	1
2	Jumper Cables	9
3	Breadboard	1
4	LCD I2C 20x4	1
5	ESP32-CAM MB	2

In this process, the system was designed based on previously planned stages. The system began with assembling hardware such as ESP32-CAM, ESP32-CAM MB, LCD I2C 20x4, and breadboard, then after components were assembled into one unit. After that, the system was configured to connect with Firebase to display vehicle information in real-time through LCD and website. Below is the final form of the assembled device:

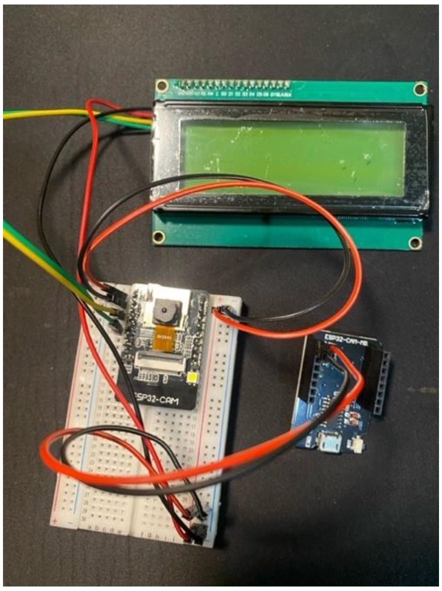

Figure 2. Complete Parking Camera System Assembly

The testing phase was conducted to ensure that the camera-based parking system functions optimally in detecting vehicles, sending data to Firebase, and displaying real-time information through LCD and website. The main objective is to identify errors to prevent operational losses and improve system quality [15]. Black box testing method was applied as functional testing based on client specifications without source code access, which evaluates the system exclusively from the end-user behavior perspective [14]. Techniques used include boundless value analysis for extreme inputs and equivalence partitioning for test case grouping—approaches proven effective in cause-effect graphs for complex scenarios—approaches that have proven effective in validating integrated IoT systems such as sensor response accuracy and real-time data synchronization in studies [16]. The main advantage of black box testing is its ability to identify aspects that have not been met from software requirements specifications [15]. However, this method has the weakness of being unable to perform comprehensive testing due to the tester's limited knowledge about the system's internals [15]. Test results will form the basis for correcting detection errors, thereby improving system quality to become optimal and effective in actual operations.

3.2 Research Location

This research was conducted at Darussalam Ganjuran Mosque, which was selected because it meets various criteria supporting the research objectives. This mosque has parking facilities spacious enough to accommodate vehicles (motorcycles and cars), making it easier to observe and test the designed parking system. The design and arrangement of the parking area at this mosque also support research needs: there are separate areas for motorcycles and cars as well as combined areas for flexible use. Additionally, Darussalam Ganjuran Mosque has a relatively high number of visitors, especially on religious holidays. These conditions provide opportunities to test the vehicle parking capacity calculation system in varied real conditions. Thus, this location is very appropriate for implementing and evaluating the IoT-based parking capacity calculation system developed in this research.

4. Result and Discussion

4.1 Results

The implementation of the camera-based parking system includes the application of functional system design and functional validation testing through black box testing. This method is a software evaluation approach oriented toward fulfilling functional specifications, where testers design various input scenarios and confirm system outputs against applied requirements [14]. In practice, black box testing demonstrates excellence in detecting discrepancies between the actual system performance and specified functional standards [15], particularly in validating real-time data transmission using MQTT protocol in IoT parking systems [17]. However, this approach has limitations as it cannot deeply audit errors in the internal logic layer due to restricted access to code structure [16].

4.1.1 Vehicle Detection Testing by ESP32-CAM

Figure 3. Camera Testing Using Cars.

Table 5. Car Testing Table.

No	Vehicle Type	Entry Detection			Exit Detection			Accuracy
	Attempt	Success	Fail	Attempt	Success	Fail	Attempt	
1	Car	1	Yes	-	1	Yes	-	100%
2	Car	2	-	Yes	2	-	Yes	0%
3	Car	3	Yes	-	3	Yes	-	100%
Total	Overall	3	2	1	3	2	1	(2/3) x 100% = 66.67%

Figure 4. Camera Testing Using Motorcycles.

Table 6. Motorcycle Testing Table.

No	Vehicle Type	Entry Detection			Exit Detection			Accuracy
	Attempt	Success	Fail	Attempt	Success	Fail	Attempt	
1	Motorcycle	1	Yes	-	1	Yes	-	100%
2	Motorcycle	2	-	Yes	2	-	Yes	0%
3	Motorcycle	3	Yes	-	3	Yes	-	100%
Total	Overall	3	2	1	3	2	1	(2/3) x 100% = 66.67%

This testing was conducted to determine the extent to which the system can accurately recognize passing vehicles. The trials were conducted three times using motorcycles and cars moving at an average speed of 10 to 15 km/h. In the first and third trials, the system supported by the ESP32-CAM camera showed good detection results that corresponded with vehicle movement. However, in the second trial, the system experienced decreased accuracy due to suboptimal lighting in the testing environment. This condition affected the camera's performance in accurately recognizing vehicle objects.

4.1.2 Website Testing

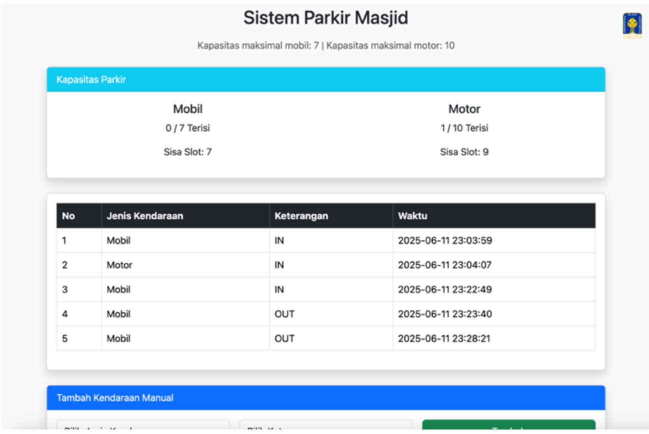

Figure 5. Testing on the Website.

This testing was conducted to test the system's ability to automatically record data of entering and exiting vehicles. The process was carried out by entering vehicle data based on type, then evaluating whether this information was correctly displayed on the website page. Results showed that the system was able to record vehicle movements in real-time, including arrival time, exit time, and available parking capacity for both motorcycles and cars.

4.1.3 LCD I2C 20x4 Testing

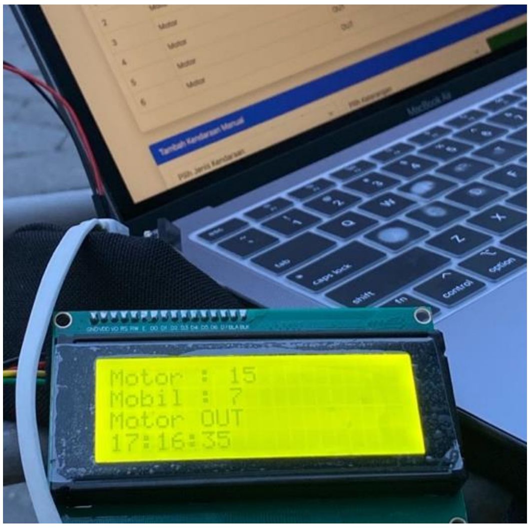

Figure 6. Testing with LCD I2C 20x4.

This testing aimed to ensure that the LCD I2C 20x4 could accurately display vehicle detection results. Data displayed included vehicle types (motorcycles and cars), vehicle status, and vehicle parking capacity, which were sent in real-time through I2C connection and microcontroller. Test results showed that the LCD display functioned well and was able to present information in accordance with the system's calculation results.

4.1.4 Firebase Testing

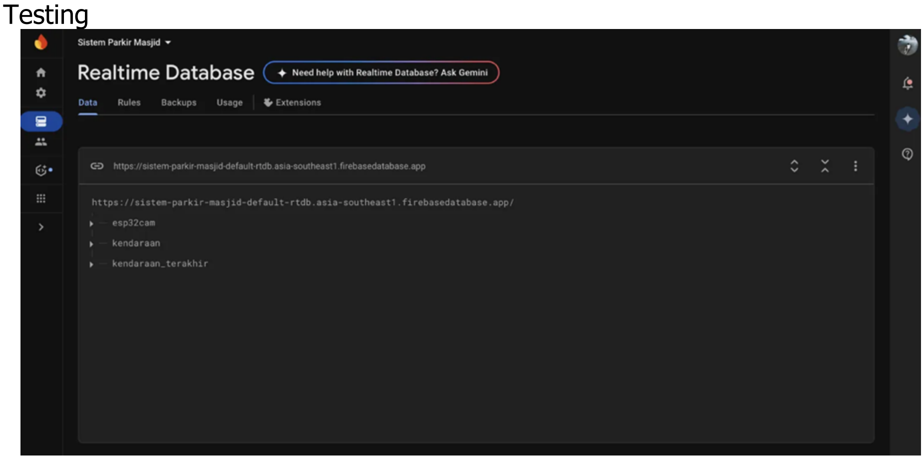

Figure 7. Firebase Database Testing

This testing was conducted to evaluate whether data from the ESP32-CAM camera could be properly sent to Firebase and displayed in real-time on the LCD and website interface. The testing process was carried out using an internet connection, considering the influence of network stability. Test results are displayed in Table 7, which shows that the data transmission process ran well, although it was influenced by the quality of the network used.

Table 7. Firebase Testing Table.

Data sent	Data received	Firebase	Website	LCD	Description
Data #1	Yes	Yes	Yes	Yes	Data successfully appeared on all display media
Data #2	No	No	No	No	Data did not successfully appear on all display media
Data #3	Yes	Yes	Yes	Yes	Data successfully appeared on all display media

4.2 Discussion

The ESP32-CAM-based parking system developed has undergone testing phases to evaluate the compatibility between designed functions and their field implementation. Based on black box test results, this system, which adopts the Internet of Things concept, is able to automatically detect vehicles in the mosque parking area. The identification process for vehicles passing through entry and exit lanes can proceed with reasonable precision, and the entire system operates in stable conditions throughout testing.

In terms of accuracy, the ESP32-CAM camera demonstrates its ability to detect vehicles passing through entry and exit lanes. Nevertheless, the system still faces several technical constraints, including insufficient lighting, limited camera image resolution, and excessive vehicle speed. These factors can cause vehicles to not be identified correctly, which means system performance is highly dependent on the environmental conditions where the device is operated. Another function of this system is to automatically send detection data to the Firebase real-time database. All transmitted data is then displayed on the LCD screen and website directly. Information presented includes the number of vehicles, entry and exit times, parking capacity, and vehicle status, all of which can be monitored in real-time. This system has several advantages such as: (1) users can know the availability of motorcycle and car parking directly; (2) parking managers can easily monitor the number of vehicles and capacity through the website display; (3) all vehicle movement data is stored online and can be accessed anytime. The shortcomings found include: (1) dependence on internet network stability; (2) absence of live video feed features from the camera; (3) decreased detection accuracy in low light conditions; (4) camera limitations in managing images independently, thus still requiring additional devices for visual processing.

5. Conclusion

This research has successfully designed and implemented a vehicle parking system based on the Internet of Things by utilizing ESP32-CAM as the main detection tool. The system is able to automatically recognize vehicles entering and exiting the mosque parking area, then send detection data to Firebase to be displayed in real-time through websites and LCD displays. Implementation results show that this system provides significant benefits for parking managers, particularly in efficiently monitoring parking capacity. Additionally, users can also directly view information on parking space availability for both motorcycles and cars. Important information such as the number of vehicles, vehicle types, vehicle capacity, and entry and exit times can be easily monitored without manual intervention. Thus, the developed system has proven capable of supporting digital and real-time parking management.

References

[1] Maulana, R., Pradani, E. R. K., & Yuniahastuti, I. T. (2023). *Prototype smart parking berbasis KTM di lingkungan Universitas Islam Raden Rahmat*. Metrotech (Journal of Mechanical and Electrical Technology), 2(3), 105–111. https://doi.org/10.33379/metrotech.v2i3.3065

[2] Winarto, I. Z. L., Kholis, N., & Anifah, L. (2023). *Rancang bangun purwarupa aplikasi masjid pintar menggunakan platform Kodular berbasis internet di masjid*. Impression: Jurnal Teknologi dan Informasi, 2(2), 61–71. https://doi.org/10.59086/jti.v2i2.288

[3] Septriyaningrum, I. A., Nugrahadi, D. T., & Ridwan, I. (2016). *Perancangan dan pengembangan prototype sistem parkir*. Kumpulan Jurnal Ilmu Komputer (KLIK), 3(2), 146–155.

[4] Nugraha, F. (2016). *Manajemen masjid: Panduan pemberdayaan fungsi-fungsi masjid*. Lekkas.

[5] Hayati, A. M. U. (2020). *Shalat sebagai sarana pemecah masalah kesehatan mental (psikologis)*. Spiritualita, 4(2). https://doi.org/10.30762/spr.v4i2.2685

[6] Wadud, A. A. (2019). *Manajemen fasilitas parkir di Masjid Al-Falah Surabaya*. Masjiduna: Jurnal Ilmiah STIDKI Ar-Rahmah, 2, 12–25.

[7] Setlight, C. Y. (2022). *Evaluasi kinerja kapasitas parkir dan kebutuhan ruang parkir di Masjid Al-Ikhlas Kota Malang*. Anggapa Journal: Building Design and Architecture Management Studies, 1, 47–56.

[8] Putra, D. I., Aisuwarya, R., Ardopa, S., & Purnama, I. (2018). *Sistem cerdas reservasi dan pemantauan parkir pada lokasi kampus berbasis konsep Internet of Things*. Jurnal Teknologi dan Sistem Komputer, 6(2), 57–63. https://doi.org/10.14710/jtsiskom.6.2.2018.57-63

[9] Purwanto, H., & Prasetio, B. (2022). *Microcontroller based parking lot monitoring system prototype*. International Journal of Research and Applied Technology (INJURATECH), 2(1), 132–141. https://doi.org/10.34010/injuratech.v2i1.6742

[10] Rahman, S., & Bhoumik, P. (2019). *IoT based smart parking system*. International Journal of Advances in Computer and Electronics Engineering, 4(1), 11–16.

[11] Alfikri, H., & Irianto, K. D. (2024). *Implementasi Internet of Things pada sistem parkir masjid*. EDUSAINTEK: Jurnal Pendidikan, Sains dan Teknologi, 11(4), 1907–1920. https://doi.org/10.47668/edusaintek.v11i4.1323

[12] Efan, S. (2021). *Sistem monitoring penempatan lokasi parkir kendaraan berbasis IoT*. Jurnal Syntax Admiration, 2(5), 930–950. https://doi.org/10.46799/jsa.v2i5.238

[13] Maulana, M. (2021). *Sistem monitoring slot parkir mobil dengan metode subtraction background berbasis Raspberry Pi* [Skripsi tidak dipublikasikan]. Politeknik Harapan Bersama Tegal.

[14] Uminingsih, M., Ichsanudin, M. N., Yusuf, M., & Suraya, S. (2022). *Pengujian fungsional perangkat lunak sistem informasi perpustakaan dengan metode black box testing bagi pemula*. STORAGE: Jurnal Ilmiah Teknik dan Ilmu Komputer, 1(2), 1–8. https://doi.org/10.55123/storage.v1i2.270

[15] Praniffa, A. C., Syahri, A., Sandes, F., Fariha, U., & Giansyah, Q. A. (2023). *Pengujian sistem informasi parkir berbasis web pada UIN SUSKA Riau menggunakan white box dan black box testing*. Jurnal Testing dan Implementasi Sistem Informasi, 1(1), 1–16. https://doi.org/10.55583/jtisi.v1i1.321

[16] Sonya, M. A., & Irianto, K. D. (2024). *Sistem pencahayaan otomatis pada smart home untuk lansia berbasis IoT*. Edusaintek: Jurnal Pendidikan, Sains dan Teknologi, 12(1), 189–207. https://doi.org/10.47668/edusaintek.v12i1.1366

[17] Pratama, R. F., Wicaksono, R. S. R., & Pramudhita, A. N. (2023). *Perancangan dan implementasi protokol MQTT pada sistem parkir cerdas berbasis IoT*. Jurnal Informatika dan Teknik Elektro Terapan, 11(3), 475–483. https://doi.org/10.23960/jitet.v11i3.3191

[18] Alfina, R., & Yusran, A. M. (2023). *Rancang bangun sistem keamanan parkir kendaraan roda dua dengan teknologi face recognition* [Tesis magister, Politeknik Negeri Ujung Pandang]. Repositori Institusi. https://repository.poliupg.ac.id/id/8294

[19] Riyanto, J., & Wasid, A. (2023). *Perancangan alat bantu parkir mobil berbasis ESP32-CAM dan sensor jarak VL53L0X menggunakan VLC*. Jurnal Informatika dan Komputasi: Media Bahasan, Analisa, dan Aplikasi, 17(1), 1–5. https://doi.org/10.56956/jiki.v17i1.173

[20] Ardana, M. D., Hartama, D., Wanto, A., Solikhun, S., & Lestari, S. P. (2023). *Implementasi sistem keamanan parkir kendaraan menggunakan sensor jarak HC-SR04 dan kamera cerdas protokol MQTT dengan Telegram*. Prosiding Seminar Nasional Teknologi Komputer dan Sains, 1(1), 383–389.

[21] Auliya, K., Yusfi, M., & Rasyid, R. (2023). *Sistem pemantauan slot parkir menggunakan sensor ultrasonik JSN-SR04T dan pengenalan plat nomor kendaraan dengan ESP32-CAM*. Jurnal Fisika Unand, 12(4), 534–540. https://doi.org/10.25077/jfu.12.4.534-540.2023

[22] Sari, D. H., Priyopradono, B., & Akhir, A. F. P. (2024). *Analisis dan perancangan sistem smart parking berbasis RFID dengan penggunaan kartu tanda mahasiswa (KTM) untuk mengurangi risiko pencurian kendaraan*. Jurnal Teknologi dan Sistem Informasi Bisnis, 6(4), 791–798. https://doi.org/10.47233/jteksis.v6i4.1561

[23] Fadillah, M. I., Fauzan, M., & Fadli, M. (2024). *Rancang bangun sistem monitoring parkir berbasis Internet of Things menggunakan metode human centered design (HCD)*. Jurnal Informatika dan Teknik Elektro Terapan, 3(2), 1907–1920. https://doi.org/10.35972/jitet.v3i2.15565