JRAEE

Journal of Robotics, Automation, and Electronics Engineering
Vol. 3, No. 2, 2025, pp. 206-213
e-ISSN 3025-3780
https://journal.uny.ac.id/v3/jraee

Design of an Automatic Number Plate Recognition System Using PaddleOCR Method and Monitoring Application for Automated Parking

Derizqiadi Thoriq[a,1,*], Arya Sony[b,2]

[a,b]Dept. of Electrical and Electronics Engineering, Vocational Faculty, Universitas Negeri Yogyakarta
[1]derizqiadithoriq.2021@student.uny.ac.id; [2]arya.sony@uny.ac.id
[*]Corresponding Author

ARTICLE INFO

Article History
Received 10 Aug 2025
Revised 29 Aug 2026
Accepted 31 Aug 2026

ABSTRACT

Manual parking systems suffer from weaknesses such as inefficiency, the risk of human error, and the prevalence of illegal parking and unauthorized fees. Therefore, an automated parking system is needed to improve efficiency, reduce human error, and facilitate integrated monitoring and management of parking. This system is designed to perform automatic number plate recognition (ANPR) and monitor parking activities, thereby reducing human error and enhancing user convenience. The system is designed using the PaddleOCR method, a deep learning-based OCR framework capable of reading text on vehicle license plates with high accuracy. The license plate recognition process begins with the detection of the license plate location, followed by reading the plate using the PaddleOCR method combined with image pre-processing and post-processing techniques. The resulting data from the license plate reading is transmitted to the Firebase and SQLite databases and monitored through the MIT App Inventor application. By integrating Raspberry Pi 5, a webcam, an OLED display, and an LED, this system is expected to optimize automated parking systems. Test results show that the developed system has excellent performance. The PaddleOCR model, combined with pre-processing and post-processing techniques, achieved a license plate character reading accuracy of 87.5%. This success was supported by the YOLOv10n detection model, which detected license plates with 90% accuracy under real-world testing conditions. All hardware components, such as the OLED display, LED, and buzzer, functioned with 100% reliability in providing notifications, and the system successfully transmitted data consistently to the Firebase, SQLite databases, and the monitoring application without any issues.

ABSTRAK

Sistem parkir manual memiliki kelemahan seperti inefisiensi, risiko human error, serta maraknya parkir liar dan pungutan liar. Oleh karena itu, diperlukan sistem parkir otomatis yang dapat meningkatkan efisiensi, mengurangi human error, dan memudahkan pemantauan serta pengelolaan parkir secara terintegrasi. Sistem ini dirancang untuk melakukan pembacaan plat nomor kendaraan secara otomatis (ANPR) dan memantau

Keywords
Automated parking;
ANPR;
PaddleOCR;
Monitoring;
Raspberry Pi 5

ISSN 3025-3780 (online)
ISSN 3025-4590 (printed)
Journal of Robotics, Automation, and Electronics Engineering
Vol. 3, No. 2, 2025, pp. 206-213

aktivitas parkir, sehingga dapat mengurangi human error dan meningkatkan kenyamanan pengguna parkir. Sistem dirancang dengan memanfaatkan metode PaddleOCR, sebuah metode OCR berbasis deep learning yang mampu membaca teks pada plat nomor kendaraan dengan akurasi tinggi. Proses pembacaan plat nomor dimulai dengan pendeteksian lokasi plat nomor kendaraan, dilanjutkan dengan pembacaan plat nomor kendaraan dengan metode PaddleOCR yang telah digabungkan dengan teknik pre-processing dan post-processing gambar. Data hasil pembacaan plat nomor dikirim ke database Firebase dan SQLite, serta dipantau melalui aplikasi MIT App Inventor. Dengan integrasi Raspberry Pi 5, kamera webcam, layar OLED, dan LED, sistem ini diharapkan dapat mengoptimalkan sistem parkir otomatis. Hasil pengujian menunjukkan bahwa sistem yang dikembangkan memiliki kinerja yang sangat baik. Model PaddleOCR yang dikombinasikan dengan teknik pre-processing dan post-processing berhasil mencapai tingkat akurasi pembacaan karakter plat nomor sebesar 87,5%. Keberhasilan ini didukung oleh model deteksi YOLOv10n yang mampu mendeteksi plat nomor dengan akurasi 90% dalam kondisi pengujian nyata. Seluruh perangkat keras seperti layar OLED, LED, dan buzzer berfungsi 100% dalam memberikan notifikasi, serta sistem berhasil mengirimkan data secara konsisten ke database Firebase, SQLite, dan aplikasi monitoring tanpa kendala.

1. Introduction

The issue of transportation management has become an increasingly pressing concern in Indonesia, especially with the significant growth in vehicle numbers. According to the Central Statistics Agency (BPS), the number of motor vehicles reached 157,080,504 in 2023, an increase of 5.95% from the previous year [1]. This growth has led to a higher demand for parking spaces and increased complexity in their management, particularly in urban and tourist areas. Manual parking systems, which are still common, suffer from inefficiencies, long queues, and human error in recording, leading to potential revenue loss. Furthermore, these challenges are compounded by the prevalence of illegal parking, which causes severe traffic congestion in areas like Ubud, Bali [2], and Lembang [3]. As a solution, an automated system is needed to create a more efficient and reliable parking management process.

This research proposes an automated system based on the Internet of Things (IoT) and Artificial Intelligence (AI) to monitor vehicle entry and exit in real-time. The core of this system is Automatic Number Plate Recognition (ANPR), a technology that uses Optical Character Recognition (OCR) to automatically read vehicle license plates from images or video [4]. The system is implemented on a Raspberry Pi 5, chosen for its powerful processing capabilities in a compact form factor, making it suitable for edge computing applications. For license plate detection, the YOLOv10n algorithm is utilized due to its excellent balance of high accuracy and computational efficiency, which is critical for real-time performance on resource-constrained devices [5]. For character recognition, the PaddleOCR framework is employed, selected for its high accuracy, which is built upon an effective CRNN (Convolutional Recurrent Neural Network) architecture, as demonstrated in similar systems [6].

Previous related research has been widely conducted. For instance, Narendra et al. [7] developed a smart parking system using YOLOv4 and KerasOCR, achieving high accuracy but with relatively slow recognition speeds. Kusanatoro et al. [8] successfully implemented Tesseract OCR for a parking system, while other recent studies by Salsabila [9] combined YOLOv8 with EasyOCR. Although these studies show significant progress, there are still opportunities to develop a highly integrated system using the latest models on resource-constrained hardware with a more robust data backend. Therefore, the novelty of this research lies in the development of an integrated ANPR system that combines the efficient YOLOv10n detection model

ISSN 3025-3780 (online)
ISSN 3025-4590 (printed)

with a custom-trained, high-accuracy PaddleOCR recognition model, all implemented on a Raspberry Pi 5 and supported by a real-time monitoring application with a reliable dual-database architecture (Firebase and SQLite). The contribution of this research is to design and build an automatic and real-time system for license plate recognition and parking monitoring. This system is able to improve the efficiency of parking management, reduce dependence on manual processes, and provide a reliable framework for data logging and remote monitoring.

2. The Proposed Algorithm

The core of this research is a multi-stage algorithm designed to process video frames and output a validated license plate string in real-time. This method ensures high accuracy by combining a state-of-the-art detection model with a robust pipeline for image enhancement and text validation. The process is divided into four main stages:

2.1 The first stage

The first stage is license plate detection and isolation. The process begins when a video frame is captured by the webcam. This frame is fed into the custom-trained YOLOv10n model, which is responsible for detecting vehicles and precisely locating the license plate within the frame by generating its bounding box coordinates. This identified area is then cropped from the main image to create a smaller Region of Interest (ROI) that contains only the license plate. This isolation is a critical first step to reduce computational load and focus the subsequent processes only on the relevant data.

2.2 The second stage

The second stage is image pre-processing for OCR enhancement. The raw, cropped license plate image is often not of sufficient quality for accurate character recognition due to factors like low resolution, noise, or poor lighting. To overcome this, the ROI undergoes a sequence of enhancement techniques:

- Upscaling: The image resolution is increased by a factor of three using bicubic interpolation. This process enlarges the characters and makes their details clearer for the OCR model.
- Denoising: The upscaled image is processed using the Non-Local Means denoising algorithm to remove visual artifacts and noise while preserving the sharp edges of the characters.
- Contrast Enhancement: Contrast Limited Adaptive Histogram Equalization (CLAHE) is applied to the image. This technique improves local contrast, making the characters stand out distinctly from the background, even in challenging lighting conditions.

2.3 The third stage

The third stage is character recognition with PaddleOCR. The enhanced image is then passed to the custom-trained PaddleOCR model for character recognition. The model, built on a CRNN (Convolutional Recurrent Neural Network) architecture, analyzes the image and extracts a sequence of characters. The output of this stage is a raw text string representing the license plate, along with a confidence score for each detected character segment.

2.4 The fourth stage

The fourth stage is post-processing and validation. The raw text from PaddleOCR is often imperfect and requires cleaning and validation to ensure it conforms to the standard Indonesian license plate format. This is achieved through a final series of steps:

- Confidence Filtering: Text segments with a confidence score below a predetermined threshold (40%) are discarded to filter out unreliable recognitions.
- Sorting and Normalization: The remaining text fragments are sorted from left to right based on their coordinates and merged into a single string. This string is then normalized by converting it to uppercase and removing all non-alphanumeric characters.
- Regex Validation: The final, cleaned string is matched against a Regular Expression (Regex) pattern that defines the valid structure of an Indonesian license plate. Only strings that match this pattern are considered valid and are formatted with the correct spacing before being sent to the database. If the string does not match, it is discarded as an invalid reading.

3. Method

The research methodology followed the Engineering Design Process (EDP), which includes defining the problem, designing a solution, building a prototype, and testing. The system's architecture is designed to be a complete, self-contained unit for vehicle entry and exit monitoring.

3.1 System Design and Hardware

The system consists of input, processing, and output components.

- Input: Two webcams capture real-time video feeds, one for incoming vehicles and one for outgoing vehicles.
- Processing: A Raspberry Pi 5 with a 4GB LPDDR4X RAM serves as the central processing unit, running all computational tasks for detection and recognition.
- Output: The system provides user feedback through a 0.96-inch OLED display for showing vehicle information, a 5mm LED for success notifications, and a buzzer for error alerts. All components are housed in a custom 3D-printed enclosure for protection and stability. Power is supplied by a 5V/5A USB-C power adapter to ensure stable operation.

3.2 Software and Model Development

The software stack was developed in Python and integrated several key technologies.

1. License Plate Detection (YOLOv10n): The YOLOv10n model was chosen for its high efficiency and low computational overhead, making it suitable for deployment on the Raspberry Pi. The model was custom-trained on a dataset of 3,358 Indonesian license plate images from Roboflow using Google Colab with a Tesla T4 GPU. This training process tailored the model to accurately locate license plates in various conditions.
2. Character Recognition (PaddleOCR): The PaddleOCR framework was used for its high accuracy and multilingual support. A custom recognition model was developed by fine-tuning the pre-trained PP-OCRv3 model, which is based on a Convolutional Recurrent Neural Network (CRNN) architecture. The model was trained on a dataset of 6,342 images of individual characters from Indonesian license plates, also sourced from Roboflow.
3. Image Processing Pipeline: To maximize accuracy, a multi-stage image processing pipeline was implemented.
    - Pre-processing: Before recognition, the cropped license plate image undergoes upscaling to increase resolution, denoising to remove artifacts, and Contrast Limited Adaptive Histogram Equalization (CLAHE) to enhance character visibility.
    - Post-processing: After OCR, the raw text output is refined through confidence-based filtering, sorting of text fragments, and normalization (converting to uppercase and removing non-alphanumeric characters). Finally, a Regular Expression (Regex) is used to validate and format the text according to the standard Indonesian license plate structure.
4. Data Management and Monitoring: The system utilizes a dual-database approach. A Firebase Realtime Database is used for cloud-based, real-time data synchronization, while a local SQLite database serves as a backup to ensure data integrity during internet outages. A monitoring application was created using MIT App Inventor, which connects to the Firebase database to display real-time parking data on an Android device.

4. Result and Discussion

4.1 Hardware Implementation and System Design

The system was physically realized in a custom enclosure designed for practical deployment. The initial design was refined, resulting in a compact box with dimensions of 10 cm (length) x 7 cm (width) x 7.3 cm (height), fabricated using 3D printing technology. The enclosure was designed to precisely house the Raspberry Pi 5, with dedicated ports for USB and power access, as well as an exhaust vent for thermal management. The front panel integrates the user interface components: a 0.96-inch OLED display, a 5V buzzer, and a 5mm LED. Two webcams were mounted on top in opposing directions to capture incoming and outgoing vehicles. A custom PCB was created to neatly manage the connections for these output components. Subsequent hardware tests confirmed that all components of the Raspberry Pi 5, webcams, OLED, LED, buzzer, and power supply functioned correctly and reliably under operational loads.

 ISSN 3025-3780 (online)
ISSN 3025-4590 (printed)

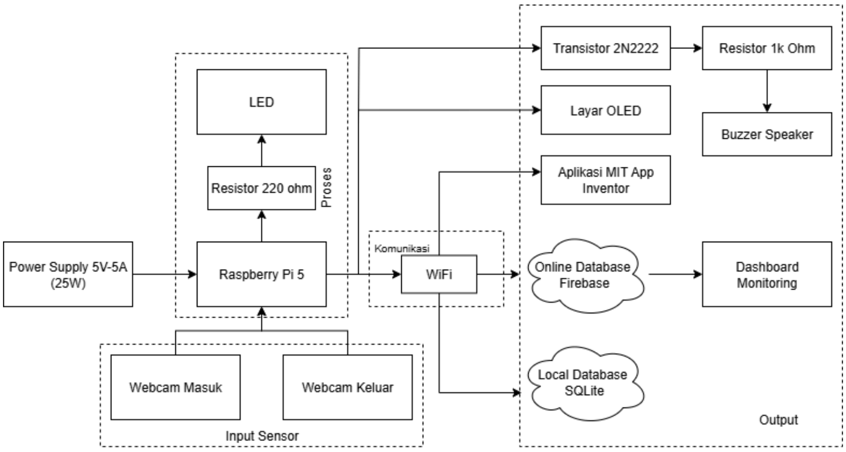


Fig. 1 Architecture of system components

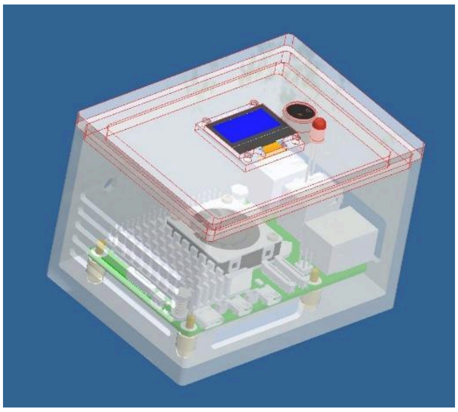


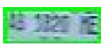
Fig. 2 Hardware components

4.2 Detection Model Performance (YOLOv10n)

The core of the vehicle detection mechanism is a custom-trained YOLOv10n model. The model was trained on a dataset of 3,358 Indonesian license plate images to ensure high relevance and accuracy. The training process on Google Colab yielded excellent performance metrics, including a precision of 0.953, a recall of 0.939, and a mAP50 of 0.98.

In real-world testing conducted on a public road, the model was evaluated on its ability to detect and correctly crop license plates from a live video feed. Out of 20 distinct vehicles that passed through the detection zone, the model successfully detected and isolated the license plates of 18 vehicles, achieving a detection accuracy of 90%. The failures were primarily attributed to motion blur caused by fast-moving vehicles, a limitation linked to the webcam's shutter speed rather than the model itself. For a typical parking environment where vehicle speeds are low, this accuracy is considered highly robust.

Table 1. YOLOv10n Number Plate Detection Result

No.	Number Plate Figure	Success	Failure	Confidence Score	Inference Speed
1.		V		0.89	288 ms
2.		V		0.92	197 ms
3.		V		0.88	244 ms
4.		V		0.66	255 ms
5.		V		0.72	288 ms
6.		V		0.79	320 ms

7.			V	0.14	889 ms
8.		V		0.92	531 ms
9.		V		0.77	391 ms
10.		V		0.84	398 ms
11.			V	0.11	911 ms
12.		V		0.94	570 ms
13.		V		0.90	219 ms
14.		V		0.91	224 ms
15.		V		0.68	213 ms
16.		V		0.82	229 ms
17.		V		0.84	203 ms
18.		V		0.81	197 ms
19.		V		0.83	235 ms
20.		V		0.76	233 ms

4.3 Recognition Model Performance (PaddleOCR)

For character recognition, a PaddleOCR model was custom-trained using a fine-tuning approach on a dataset of 6,342 images of characters from Indonesian license plates. The training resulted in a model with a validation accuracy of 82.81%, a low Normalize Edit Distance, and minimal loss, indicating that the model had learned the character patterns effectively.

To maximize real-world accuracy, a comprehensive image processing pipeline was implemented. This included pre-processing steps (upscaling, denoising, and CLAHE for contrast enhancement) to improve image quality before recognition, and post-processing steps (confidence filtering and Regex validation) to clean and format the OCR output. When tested in the field, this enhanced PaddleOCR system correctly read 7 out of the 8 license plates provided by the YOLOv10n model, achieving a final character recognition accuracy of 87.5%. This result confirms the effectiveness of combining a custom-trained model with a rigorous processing pipeline.

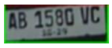
Table 2. PaddleOCR Custom Model Training Result

No.	Learning Rate	Accuracy	Normalize Edit Distance	Loss
1.	0.00001	0.000000	0.000000	143.720861
10.	0.000481	0.000000	0.969358	24.638519
20.	0.000462	0.000000	0.958364	17.827087
30.	0.000410	0.000000	0.933868	14.431591
40.	0.000343	0.171875	0.911756	8.994470
50.	0.000267	0.312500	0.844674	5.827691
60.	0.000189	0.476562	0.715365	3.282364
70.	0.000117	0.648437	0.581641	1.990310
80.	0.000058	0.687500	0.279009	1.612456
90.	0.000013	0.758749	0.138244	1.297640
100.	0.000001	0.828125	0.074398	1.015610

Table 3. Number Plate Recognition Testing Result

No	Number Plate	PaddleOCR Recognition	Confidence Score	Fragmen	Latency	Number Plate Figure	Result
1	AB 1336 VS	AB 1336 VS	99.02%	1	191 mx		Success

Journal of Robotics, Automation, and Electronics Engineering
Vol. 3, No. 2, 2025, pp. 206-213
ISSN 3025-3780 (online)
ISSN 3025-4590 (printed)

2	AB 1474 QA	AB 1474 OA	83.11%	1	380 ms		Fail
3	AB 1498 XZ	AB 1498 XZ	98.68%	1	197 ms		Success
4	AB 1794 EY	AB 1794 EY	94.03%	1	186 ms		Success
5	AB 1556 IN	AB 1556 IN	99.84%	1	203 ms		Success
6	AB 1862 KV	AB 1862 KV	98.43%	1	446 ms		Success
7	H 1791 PT	H 1791 PT	99.85%	1	202 ms		Success
8	AB 1580 VC	AB 1580 VC	99.31%	1	214 ms		Success

4.4 Integrated System Testing and Data Transmission

The fully integrated system was tested in a real parking lot scenario to evaluate its end-to-end performance. The system successfully managed the entire workflow, including detecting vehicles, reading license plates, triggering hardware notifications, and transmitting data. For every successfully detected vehicle, the hardware notifications (OLED display, LED, and buzzer) functioned with 100% reliability.

Crucially, the system's IoT functionality performed flawlessly. Data for every detected event was successfully sent to both the cloud-based Firebase database and the local SQLite backup database, achieving a 100% data transmission success rate. Furthermore, the data displayed on the MIT App Inventor monitoring application perfectly matched the records in the Firebase database, confirming a consistent and reliable data flow from the device to the end-user.

Table 4. Automated Parking System Testing Result

No	Number Plate	Entry Reading	Exit Reading	Hardware Notification	Transmission to Database	Status
1	AB 1336 VS	Success	Success	Success	Success	Exit
2	AB 1474 QA	Fail	Success	Success	Success	Exit
3	AB 1498 XZ	Success	Success	Success	Success	Exit
4	AB 1794 EY	Success	Success	Success	Success	Exit
5	AB 1556 IN	Success	Fail	Success	Success	Entry
6	AB 1862 KV	Success	Fail	Success	Success	Entry
7	H 1791 PT	Success	Success	Success	Success	Exit
8	AB 1580 VC	Success	Success	Success	Success	Exit

4.5 Discussion

The results demonstrate the successful development of a highly effective and reliable ANPR system. The 90% detection accuracy of the YOLOv10n model and the 87.5% recognition accuracy of the custom PaddleOCR model are strong indicators of the system's capability in a real-world setting. The minor inaccuracies were primarily linked to hardware limitations (webcam speed) and challenging environmental conditions, which are less prevalent in a controlled parking scenario.

The discussion highlights the system's most significant achievement: its robust integration and reliability. The 100% success rate in hardware notifications and data transmission underscores the system's readiness for deployment. By successfully automating the process of vehicle identification and data logging, the system directly addresses the core problems of inefficiency and human error found in manual parking systems. The dual-database architecture provides a critical layer of data security and integrity, while the mobile application offers a practical tool for real-time monitoring. This research thus contributes a complete, field-tested solution that is both accurate and dependable for modernizing parking management.

5. Conclusion

Based on the research and implementation conducted, it can be concluded that the design of an automatic number plate recognition system using the PaddleOCR method has been successfully realized by effectively integrating hardware and software components. The system demonstrated strong performance, with the YOLOv10n model achieving a 90% accuracy in license plate detection and the custom-trained PaddleOCR model attaining a character reading accuracy of 87.5% in real-world testing conditions. Furthermore, the system's performance in data transmission was proven to be highly functional and reliable; it successfully sent parking data for every detected vehicle to the Firebase and SQLite databases and the monitoring application with 100% success, ensuring data integrity and accessibility. The results achieved are consistent with the objectives stated in the introduction, where this system was proposed as a solution to the inefficiencies of manual parking. The successful implementation provides an automated, accurate, and integrated framework that effectively reduces dependency on manual processes. For future development, several improvements can be made. To enhance detection and recognition accuracy, especially for moving vehicles, it is recommended to use a camera with a higher specification, particularly a faster frame rate (FPS) and shutter speed, to minimize motion blur. Additionally, to ensure every vehicle is detected reliably, the system could be integrated with physical infrastructure such as a parking gate or road barrier, which would guide vehicles to pass through the optimal detection zone.

Author Contribution:

All authors contributed equally as main contributors to this paper. All authors read and approved the final paper.

Funding:

This research received no external funding.

Conflicts of Interest:

The authors declare no conflict of interest.

References

[1] Badan Pusat Statistik (BPS), “Jumlah kendaraan bermotor menurut provinsi dan jenis kendaraan (unit), 2023,” 2023. [Online]. Available: https://www.bps.go.id/id/statistics-table/3/VjJ3NGRGa3dkRk5MTlU1bVNFOTVVbmQyVURSTVFUMDkjMw==/jumlah-kendaraan-bermotor-menurut-provinsi-dan-jenis-kendaraan--unit---2022.html?year=2023. [Accessed: Aug. 12, 2024].

[2] P. P. G. Parwata, I. A. P. Widiati, and I. W. Artanaya, “Efektivitas penertiban parkir liar di kawasan obyek wisata Ubud,” *Jurnal Konstruksi Hukum*, vol. 2, no. 1, pp. 207–212, 2021, doi: 10.22225/jkh.2.1.2999.207-212.

[3] A. Zulhan, “Peran Dinas Perhubungan Kabupaten Bandung Barat (suatu studi tentang penertiban pelanggaran parkir di Kecamatan Lembang),” undergraduate thesis, Program Studi Ilmu Pemerintahan, Universitas Komputer Indonesia, Bandung, Indonesia, 2019.

[4] S. Joshi, P. Jejure, C. Jadhav, and V. Jankar, “Automatic number plate recognition,” *International Journal of Scientific Research in Science and Technology*, vol. 11, no. 5, pp. 439–448, 2024, doi: 10.32628/IJSRST2411476.

[5] A. Sundaresan Geetha, M. A. R. Alif, M. Hussain, and P. Allen, “Comparative analysis of YOLOv8 and YOLOv10 in vehicle detection: Performance metrics and model efficacy,” *Vehicles*, vol. 6, no. 3, pp. 1364–1382, 2024, doi: 10.3390/vehicles6030065.

[6] I. A. Kamal, F. A. P. Putra, F. M. Lasidi, W. Priharti, and W. A. Cahyadi, “Implementasi sistem gerbang otomatis dengan perpaduan teknologi pengenalan pelat nomor kendaraan dan pengenalan wajah,” *Jurnal Serambi Engineering*, vol. 9, no. 4, pp. 10479–10486, 2024.

[7] A. Narendra, S. R. Choudhury, A. Mishra, and I. Misra, “Chaurah: A smart Raspberry Pi based parking system,” *TechRxiv*, preprint, 2023, doi: 10.36227/techrxiv.21931398.v1.

[8] K. Kusnantoro, T. Rohana, and D. S. Kusumaningrum, “Implementasi metode Tesseract OCR (Optical Character Recognition) untuk deteksi plat nomor kendaraan pada sistem parkir,” *Scientific Student Journal for Information, Technology and Science*, vol. 3, no. 1, pp. 59–67, 2022.

[9] N. Salsabila and S. Sriani, “Enhancing automated vehicle license plate recognition with YOLOv8 and EasyOCR,” *Journal of Information Systems and Informatics*, vol. 6, no. 3, pp. 1577–1597, 2024, doi: 10.51519/journalisi.v6i3.848.

ISSN 3025-3780 (online)
ISSN 3025-4590 (printed)