AI Based Seamless Vehicle License Plate Recognition Using Raspberry Pi Technology

Alvis Abreo
CHRIST (Deemed to be University)
Bengaluru, India
Email:
alvis.abreo@mca.christuniversity.in

Mayur C
CHRIST (Deemed to be University)
Bengaluru, India
Email:
mayur.c@mca.christuniversity.in

Somnath Sinha
CHRIST (Deemed to be University)
Bengaluru, India
Email:
somnath.sinha@christuniversity.in

***Abstract*— This research presents the implementation of an innovative Vehicle Management System designed specifically for the Christ University Project "CampusWheels." The system incorporates cutting edge technologies, including YOLOv8 and Tesseract OCR, for robust license plate recognition. Addressing the unique challenges faced by Christ University in managing and securing vehicular movements within the campus, this project becomes crucial as the number of vehicles on campuses continues to grow. It not only provides an effective solution to these challenges but also introduces innovative methodologies, marking a significant departure from conventional campus management practices. The paramount importance of this project lies in its ability to enhance campus security through real-time vehicle monitoring and identification. The utilization of YOLOv8 for vehicle detection and Tesseract OCR for license plate recognition ensures a high level of accuracy in identifying and tracking vehicles entering and leaving the campus. This precision significantly contributes to the prevention of unauthorized vehicle access, a common security concern on educational campuses. Moreover, the system's ability to streamline traffic f low and improve efficiency in parking and access control addresses practical issues faced by campus administrators and security personnel.**

***Keywords*—Indian Number Plate Recognition, Raspberry Pi, YOLOv8, Tesseract OCR, Customized Vehicle Management**

I. INTRODUCTION

This paper introduces a cost-effective solution, leveraging Raspberry Pi, YOLOv8 object detection, and Tesseract OCR for comprehensive vehicle image capture and license plate recognition. In addressing security and efficient vehicle management, the innovative system transforms traffic monitoring across diverse domains, including campuses, parking facilities, and urban environments.

The Raspberry Pi 4, with 8GB RAM, serves as the central computational unit, enabling efficient video analysis and model inference [1]. Increased RAM capacity enhances concurrent task handling, crucial for resource-intensive algorithms like YOLOv8 and Tesseract OCR. The 5MP Raspberry Pi Camera integrates seamlessly, providing high-quality video capture for accurate vehicle detection and license plate recognition [2].

YOLOv8, a high-performance single-stage object detection model, is renowned for its accuracy and efficiency, making it a popular choice due to its superior performance compared to other models [3]. As the latest iteration in the YOLO series, it employs a grid-based approach, predicting bounding boxes and class probabilities for each cell using a convolutional neural network (CNN) architecture. Notable enhancements, including a path aggregation network (PAN) and a spatial attention mod ule (SAM), contribute to increased accuracy by capturing more contextual information. YOLOv8's efficiency allows it to process images at over 100 frames per second, making it suitable for real-time applications in autonomous vehicles and security systems [4].

Tesseract OCR, a robust open-source optical char acter recognition (OCR) engine by Google [5], is instrumental in extracting alphanumeric details from license plates in the Vehicle Management System. Renowned for its adaptability and maintained by Google, Tesseract employs advanced machine learning algorithms, ensuring precise text recognition from images. Its open-source nature guarantees continual updates, enhancing its effectiveness in various OCR applications.

This cost-effective solution centralizes on Rasp berry Pi for precise vehicle image capture, seamlessly integrating the Raspberry Pi Camera for both image capture and local server hosting. Utilizing the YOLOv8 object detection algorithm ensures accurate positioning of license plate regions, followed by automated character extraction for license plate recognition, achieving a balance of accuracy and efficiency. The research paper explores the practicality and effectiveness of this streamlined approach, high lighting its potential to redefine security practices and enhance operational efficiency across diverse domains.

II. LITRATURE REVIEW

The proposed ER-ALPR system [6] integrates image preprocessing, YOLOv8 for license plate frame detection, a virtual judgment line for frame passage determination, M-YOLOv8 for character recognition, and an auxiliary judgment system. Real-life tests in Taiwan showcase its impressive performance, achieving 97 percent license plate character recognition during the day and 95 percent at night, contributing significantly to real-time ALPR challenges and advancing traffic management with computational efficiency through the AGX system. Another notable system [7], designed by A. O. Agbeyangi et.al, addresses contemporary traffic management and law enforcement needs by leveraging Raspberry Pi, OpenCV, and OCR for streamlined vehicle identification and traffic regulation enforcement. As the number of vehicles rises, the system's cost-effectiveness and accessibility make it valuable for various traffic management scenarios, with experimental results highlighting superior performance under diverse conditions and potential contributions to automated license plate recognition.

In the study outlined in [8], the impact of image segmentation techniques on ALPR system effectiveness is emphasized, comparing Canny Edge and Otsu Thresholding.

Using a Raspberry Pi for processing, the research evaluates their performance across different time segments, with Canny Edge outperforming Otsu Thresholding, achieving a 100 percent edge detection rate and contributing to an overall system accuracy of 72 percent. This underscores the importance of segmentation techniques in real-world time variations, offering insights for improved ALPR performance. Additionally, the proposed prototype in [9] introduces an automatic gate access control system, utilizing a camera, optical recognition, and a database for automated decision making. Leveraging Raspberry Pi, Pi camera, servo motor, and Python, the system demonstrates practicality and accessibility. Successful testing affirms its viability, presenting an innovative solution to security and access control challenges, particularly in scenarios with non-standardized license plates and image-based recognition requirements.

In the work by H. Padmasiri et al. [10], a resource-efficient ALPR system tailored for low resource edge devices showcases robust performance in challenging day and night scenarios. Achieving adaptability through optimized neural networks for different hardware configurations, the study represents a significant leap in addressing re source limitations in edge computing. Additionally, in the study outlined in [11], an efficient methodology integrates a camera with a gateway for precise vehicle detection and recognition. Complemented by IoT technology, the system enables comprehensive vehicle and number plate detection, contributing to enhanced security measures for access control and trespassing prevention.

The study in [12] introduces an innovative li cense plate recognition system with three key image processing stages: pre-processing, segmentation, and character recognition. Using Canny edge detection, it achieves a remarkable 93 percent accuracy in recognizing Arabic license plates. Implemented with ESP32 Cameras and Raspberry-Pi, the prototype demonstrates real-world effectiveness, offering ap plications in smart city development and traffic management to address urban challenges. Addition ally, the research in [13] significantly contributes to ALPR, providing a comprehensive review of deep learning methodologies and IoT sensor integration. The proposed system involves four key steps: Li cense Plate Extraction, Image Pre-processing, Character Segmentation, and Character Recognition. Innovative approaches for the first three steps, along with experimentation on four character recognition methods, offer valuable insights into the evolving landscape of ALPR, addressing contemporary traffic management and security system demands.

he system presented in [14] utilizes IoT for a cost-effective solution in identifying vehicles in irregular conditions, emphasizing community-based security. It provides comprehensive details, including license plates, make, model, colour, city, state, passenger capacity, and restrictions. The embedded Raspberry Pi system integrates various features like GPS, solar panels, 3G modem, Wi-Fi, camera, and motion sensors for autonomy. Testing validates its effectiveness in diverse conditions, offering practical and innovative surveillance and vehicle identification in remote regions. The emphasis on community based security enhances practical and cost-effective solutions for underinvested areas.

The proposed research work distinguishes itself within the realm of Automatic License Plate Recognition (ALPR) by amalgamating cutting-edge technologies and innovative methodologies tailored specifically for the unique challenges of real-world applications. Unlike the first paper [6], our research employs a training dataset comprising over 25,000 images capturing Indian vehicles and their associated license plates, ensuring adaptability to diverse scenarios and appearances. This extensive dataset facilitates a robust performance, enhancing the model’s precision in discerning and localizing license plates accurately. In contrast to the second paper [7], our approach leverages the YOLOv8 object detection algorithm, emphasizing its prowess in precisely identifying and tracking vehicles in real time, thereby contributing to superior recognition of license plates. Furthermore, unlike the fourth paper [9], our proposed system extends beyond gate access control, aiming for comprehensive vehicle management within campus environments.

Moreover, our research stands out from the sixth paper [11], which focuses on efficient access control, by encompassing the broader context of campus vehicle management, incorporating YOLOv8 for vehicle detection, and EasyOCR for license plate recognition. While the seventh paper [12] introduces an innovative design for Arabic license plate recognition, our research contextualizes the application within the educational setting, addressing the unique challenges faced by Christ University. Additionally, our work differentiates itself from the ninth paper [14], which emphasizes community-based security, by providing a specialized solution for campus vehicle management, optimizing efficiency and security through a combination of advanced technologies. In summary, our research work offers a nuanced and comprehensive approach to ALPR, tailored to the specific challenges and requirements of managing vehicular traffic within a campus environment.

III. Methodology

Fig 1. shows the brief methodology of the proposed system. The detailed description is as follows.

Step 1. Video Capture and Motion Detection: This involves using the Raspberry Pi 8MP cam era for continuous campus monitoring, integrating motion detection algorithms to identify movement. When motion is detected, indicating the presence of a vehicle or other activity, frames are captured from the live video feed in real-time. This efficient surveillance mechanism conserves computational resources by focusing on relevant frames and allows the system to promptly respond to changes in the campus environment. The proactive engagement of motion detection makes the system an effective tool for campus monitoring and security.

Step 2. Frame Transmission to PC Server: After capturing frames, the Raspberry Pi 8MP camera uses Flask, a lightweight Python web framework, to efficiently transmit frames to a central PC server. This streamlined communication ensures fast and reliable data transfer for subsequent analysis. Flask’s versatility simplifies the process, highlighting the project’s emphasis on real-time connectivity and data flow, enhancing the overall effectiveness of the campus vehicle management system.

Step 3. Sequential Image Saving and Change Detection: Upon successful transmission of frames to the PC server, the frames undergo two key processes: sequential image saving and change detection. The frames are systematically saved in sequence, establishing a chronological record of the campus environment. This sequential storage not only preserves the historical visual data but also forms the basis for subsequent analyses. Simultaneously, a counter mechanism is initiated to detect and record any changes within the captured frames. This change detection process serves as a real-time monitoring tool, allowing the system to identify and document alterations in the campus surround ings. The counter updates dynamically, capturing and quantifying any modifications in the visual data. This dual process of sequential image saving and change detection provides a comprehensive approach to maintaining a historical record while actively monitoring the campus environment for any noteworthy events or alterations, contributing to the overall efficacy of the vehicle management system.

Step 4. YOLOv8 Object Detection and License Plate Identification: The heart of the system lies in the implementation of the YOLOv8 model, a state of-the-art object detection algorithm. The training methodology employed in this study is pivotal for the robust performance of our model. The training dataset comprised a comprehensive collection of over 25,000 images capturing Indian vehicles and their associated license plates. This extensive dataset facilitated the model's ability to discern various scenarios and diverse license plate appearances, con tributing to its adaptability in real-world conditions. The training process involved utilizing the YOLOv8 object detection algorithm, which effectively recognizes the presence of a vehicle in the captured image. Subsequently, the model extracts coordinates specifying the region containing the license plate, enabling precise localization.

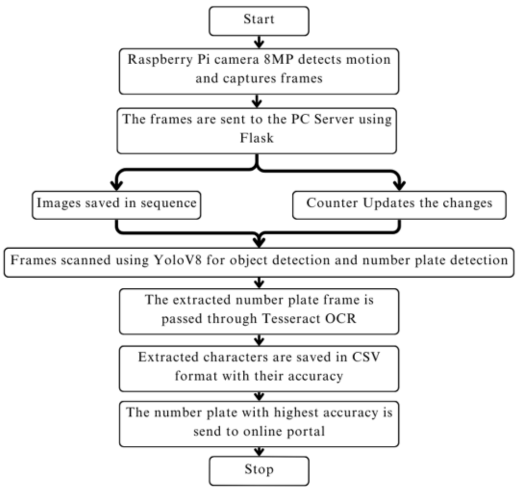

Fig. 1. Methodology of the Indian Vehicle License Plate Detection

Step 5. Tesseract OCR for Alphanumeric Extraction: Following the initial detection, the segmented part of the image containing the license plate is forwarded to another model equipped with Tesseract OCR. This additional model is specifically trained to recognize characters from the license plate images. Notably, the training encompassed the latest series of Indian license plates, such as the BH series, ensuring the model's proficiency in handling contemporary registration patterns. The utilization of Tesseract OCR, a powerful optical character recognition tool, enhances the system's accuracy in extracting alphanumeric information from the license plates, thereby completing the comprehensive license plate recognition process.

Step 6. CSV Format and Data Organization: Following the successful extraction of alphanumeric characters from the license plates, the next crucial step involves the systematic organization and storage of this information. The extracted alphanumeric characters are meticulously arranged and stored in CSV(Comma-Separated Values) format. CSV, being a widely used plain-text format, offers a structured and easily accessible means of organizing data. In this context, the CSV file serves as a repository for storing license plate details, including the specific alphanumeric characters identified by the system and their corresponding accuracy levels derived from the character recognition process. Each entry in the CSV file corresponds to a specific instance of license plate recognition, providing a concise and organized representation of the extracted information. This format enhances the system's efficiency by facilitating straightforward data management, retrieval, and analysis, contributing to the overall effectiveness of the Vehicle Management System implemented for Christ University's "CampusWheels" project.

Step 7. Selection of Highest Accuracy Number Plate: In the final step of the process, the system undertakes the critical task of selecting the number plate with the highest accuracy from the extracted data. After the alphanumeric characters are organized and stored in CSV format, each entry is associated with a corresponding accuracy level, reflecting the reliability of the character recognition process. The system, in this step, intelligently identifies and chooses the license plate entry with the highest accuracy among all the captured frames. This selection is based on a comparison of accuracy levels associated with each entry. The chosen information, representing the most precise license plate details, is then forwarded to an online portal. This strategic selection process ensures that the online portal receives and archives the license plate information with the utmost accuracy. By transmitting only the data with the highest confidence in its correctness, the system contributes to the online portal's efficacy in real-time monitoring and management of vehicles within the campus environment, aligning with the overarching goals of the "CampusWheels" project at Christ University.

Keep your text and graphic files separate until after the text has been formatted and styled. Do not use hard tabs, and limit use of hard returns to only one return at the end of a paragraph. Do not add any kind of pagination anywhere in the paper. Do not number text heads-the template will do that for you.

IV. IMPLEMENTATION

The implementation was tested in the Christ University. The fig 2 and 3 shows image capture and the results delivered by the model. The detailed flow of implementation results are as below:

The first phase involves real-time video capture and motion detection using the Raspberry Pi 8MP camera. The camera continuously monitors the campus environment, and upon detecting motion, captures frames from the live video feed. These frames are then transmitted to a PC server using Flask, establishing a seamless connection for further processing. The systematic saving of frames in sequence and simultaneous change detection provide a chronological record of campus activities.

In the subsequent phase, the YOLOv8 model, known for its superior object detection capabilities, scans the systematically saved frames. YOLOv8 precisely identifies and tracks vehicles in real-time, contributing to enhanced security through accurate monitoring. Following object detection, the model is employed for license plate recognition. This Optical Character Recognition tool ensures a high degree of accuracy in extracting alphanumeric characters from the identified license plates.

Fig. 2. Vehicle entering the campus

The extracted alphanumeric characters are organized and stored in CSV format, offering a structured and easily accessible database for license plate details. To conclude the process, the system selects the number plate with the highest accuracy from the extracted data. This selected information is then transmitted to an online portal, where it is stored for real-time monitoring and management of vehicles within the campus environment.

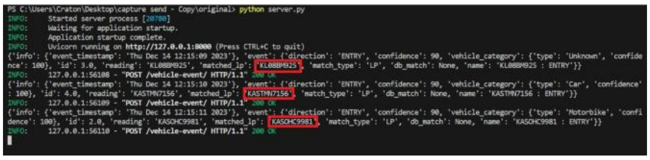

Fig. 3. The highlighted part of the output shows the characters extracted by the model.

Overall, the implementation of the ”CampusWheels” project represents a transformative step towards intelligent and technologically-driven cam pus management, addressing security concerns, optimizing vehicular traffic, and showcasing the potential of cutting-edge technologies in educational institutions.

V. Results and Analysis

In our experimentation, we conducted 25 tests for each case to evaluate license plate reading accuracy under varying distances. Case I, at 1 meter, achieved 77 percent accuracy with a 4 percent false reading rate. In Case II (1.25 meters), accuracy slightly decreased to 59 percent, with a 9 percent false reading rate. Extending to 1.5 meters in Case III resulted in 43 percent accuracy and a 12 percent false reading rate. Beyond 1.5 meters in Case IV, accuracy dropped to 22 percent, with a 15 per cent false reading rate. These findings emphasize the need for optimization in scenarios with greater frame distances.

For this study, the following system configurations were employed: Raspberry Pi with 8 GB RAM, Raspberry Pi Camera with 8 MP resolution, and a local desktop with a minimum of 8 GB RAM. These configurations were crucial in ensuring the robust ness and efficiency of the license plate recognition system under investigation.

The accuracy (percent) is calculated as: Accuracy = (Number of True Number Plates Read Correctly by Model / Total Number of Test Cases)×100

Similarly, the False Reading (percent) is calculated as: False Reading = (Number of False Number Plates Read by Model / Total Number of Test Cases)×100

Fig 4 represents the visualization of the results in a graph, while the Fig 5 represents the accuracy of the model computed in together in tabular format.

These results highlight the sensitivity of license plate reading to varying distances, emphasizing the need for optimization and further investigation, particularly in scenarios where the distance between the camera and the vehicle is greater.

Fig. 4. Graph showing the accuracy of the developed model.

Case	Distance of Frames Captured (in meters)	Accuracy (%)	False Reading (%)
I	1	77	23
II	1.25	59	41
III	1.5	43	57
IV	>1.5	22	78

Fig. 5. Table showing the numerical representation of the accuracy obtained

VI. Future Scope

The future scope of this license plate recognition application involves integrating machine learning and deep neural networks to enhance accuracy across distances and lighting conditions. Real-time data analytics and predictive modeling can enable proactive decision-making in vehicle management and security. Collaborations with smart city initiatives could amplify its impact on urban mobility. Exploring cloud-based solutions and edge computing ensures scalability and adaptability to evolving technological landscapes.

VII. Conclusion

In conclusion, our research presents a comprehensive evaluation of license plate reading performance under different distances, revealing crucial insights into the system's sensitivity and limitations. The varying results across cases underscore the importance of optimizing the system for different scenarios, particularly when dealing with greater distances between the camera and the vehicle.

This study provides a foundation for further refinement of license plate reading systems, suggesting the necessity of adaptive algorithms or additional technologies to enhance performance across diverse spatial contexts.

VIII. References

[1] R. Model P. Ltd, "Buy A Raspberry Pi 4 B– Raspberry Pi,"RaspberryPi, https://www.raspberrypi.org/products/raspberry-pi-4 model-b/

[2] "Raspberry Pi Documentation- camera." https://www.raspberrypi.com/documentation/accessories/camera.html

[3] A. Bochkovskiy, "YOLOV4: Optimal speed and accuracy of object detection," arXiv.org, Apr. 23, 2020. https://arxiv.org/abs/2004.10934

[4] D. Reis, "Real-Time Flying Object Detection with YOLOv8," arXiv.org, https://arxiv.org/abs/2305.09972 May 17, 2023.

[5] Tesseract-Ocr, "GitHub- tesseract-ocr/tesseract: Tesseract Open Source OCR Engine (main repository)," GitHub. https://github.com/tesseract-ocr/tesseract

[6] C. Lin, C. Chuang, and H.-Y. Lin, "Edge-AI-Based Real Time Automated License Plate Recognition System," Ap plied Sciences, vol. 12, no. 3, p. 1445, Jan. 2022, doi: 10.3390/app12031445.

[7] A. O. Agbeyangi, O. A. Alashiri, and A. E. Otunuga, "Automatic Identification of Vehicle Plate Number using Raspberry Pi," IEEE Xplore, Mar. 2020, doi: 10.1109/icm cecs47690.2020.246983.

[8] A. Firasanti, T. E. Ramadhani, M. A. Bakri, and E. A. Z. Hamidi, "License Plate Detection Using OCR Method with Raspberry Pi," IEEE Xplore, Nov. 2021, doi: 10.1109/tssa52866.2021.9768252.

[9] S. A. Kahie, A. A. Nor, A. H. Hasan, A. M. Abdi, L. M. Hassan, and M. A. Mohamud, "A Smart Access Control for Restricted Buildings Using Vehicle Number Plates Recognition System," IEEE Xplore, Aug. 2021, doi:10.1109/esmarta52612.2021.9515752.

[10] H. Padmasiri, J. Shashirangana, D. Meedeniya, O. Rana, and C. Perera, "Automated License Plate Recognition for Resource-Constrained Environments," Sensors, vol. 22, no. 4, p. 1434, Feb. 2022, doi: 10.3390/s22041434.

[11] P. a. H. Vardhini, V. K. R. Yasa, and G. J. Raju, "Rasp berry Pi Vehicle Gateway System with Image Processing based Authorization Detection using IoT," 2021 IEEE International Conference on Electronics, Computing and Communication Technologies (CONECCT), Jul. 2021, doi:10.1109/conecct52877.2021.9622528.

[12] M. M. Abdellatif, N. H. Elshabasy, A. E. Elashmawy, and M. Abdel-Raheem, "A low cost IoT-based Arabic license plate recognition model for smart parking systems," Ain Shams Engineering Journal, vol. 14, no. 6, p. 102178, Jun. 2023, doi: 10.1016/j.asej.2023.102178.

[13] M. A. Jawale, P. William, A. B. Pawar, and N. Marriwala, "Implementation of number plate detection system for vehicle registration using IOT and recognition using CNN," Measurement: Sensors, vol. 27, p. 100761, Jun. 2023, doi: 10.1016/j.measen.2023.100761.

[14] L. A. Glasenapp, A. F. Hoppe, M. A. Wisintainer, A. Sartori, and S. F. Stefenon, "OCR Applied for Identification of Vehicles with Irregular Documentation Using IoT," Electronics, vol. 12, no. 5, p. 1083, Feb. 2023, doi:10.3390/electronics12051083