Autonomous Parking Bay Design using ANPR

Antipas Chinodakufa

\*School of Information Sciences, Harare Institute of Technology   
P. O. Box BE 277, Belvedere, Harare

Zimbabwe

[1h190317j@hit.ac.zw](mailto:1h190317j@hit.ac.zw)

**Abstract**— This paper presents the design and implementation of an autonomous parking system that uses automatic number plate recognition (ANPR) and a payment platform to manage parking services in Zimbabwe. The system uses a camera and an optical character recognition (OCR) algorithm to scan the number plate of a car and start a timer when it enters a parking bay. The system also queries a database for the customer account associated with the number plate and updates it with the parking cost and records the car service and fee information. The customer can pay for the parking using PayNow, a third-party payment platform that is integrated with the system. The system also generates a report of the parking activity using the data from the database. The system aims to reduce errors and frauds, improve efficiency and speed, enhance convenience and safety for customers, lower costs and labor, and provide accurate and reliable data for reporting and analysis. The paper describes the software tools and techniques used to design, develop, test, and implement the system.

**Keywords**— ANPR, parking, autonomous, license, duration, cost

1. Introduction

Parking is an essential service for any urban area, especially for supermarkets that attract a large number of customers. However, parking systems in Zimbabwe are mostly human intensive and manual, which leads to various problems such as errors, frauds, inefficiency, inconvenience, high costs, and poor data quality. To address these problems, this paper proposes an autonomous parking system that uses automatic number plate recognition (ANPR) and a payment platform to manage parking services for any owner of parking space in Zimbabwe. The system uses a camera and an optical character recognition (OCR) algorithm to scan the number plate of a car and start a timer when it enters a parking bay. The system also queries a database for the customer account associated with the number plate and updates it with the parking cost and records the car service and fee information. The customer can pay for the parking using PayNow, a third-party payment platform that is integrated with the system. The system also generates a report of the parking activity using the data from the database. The system aims to reduce errors and frauds, improve efficiency and speed, enhance convenience and safety for customers, lower costs and labor, and provide accurate and reliable data for reporting and analysis. The paper describes the software tools and techniques used to design, develop, test, and implement the system. The paper also evaluates the performance and benefits of the system compared to the current system.

1. Related Work

The section presents an analysis on ANPR and the different ways used for ANPR.

1. ANPR

Automatic Number Plate Recognition system (ANPR) is a way of identifying vehicles by their licence plates using image processing. ANPR is useful for managing parking areas because each vehicle has a unique licence plate that can be used for identification [1]. ANPR also has other uses and examples are controlling the border, measuring time for a journey, controlling access to private property, highway tolling and monitoring traffic on the roads. [2].

ANPR has four main stages which are image acquisition, number plate detection, character segmentation and character recognition [3] [4] [5] [7] [8]. ANPR does the following process:

i. Image taken from camera

ii. Image pre-processed

iii. Number plate detection

iv. Character segmentation

v. Character recognition

vi. Output of character.

[3] [5] [7]

Stage 1 – image acquisition – this is where a camera gets a video or an image and in case of video, it is changed to frames in a pre-processing step. It is also where grey scale conversion and noise filtering happen to make number plate recognition better [5]. In [7], pre-processing has 4 steps, grey scaling, noise removal, edge detection and dilation.

Stage 2 – number plate detection – This is how to get the licence plate area by cutting vehicle image for later recognition system to check [6]. Reducing processing time for licence plate detection is done by only processing pixel with licence plate features like colour, size, edge boundary and presence of character feature [4]. Licence plates have high contrast areas and these yellow and black or white and black areas help with licence plate recognition. [5][12].

Stage 3 – character segmentation – This is splitting the licence plate image again and again until every character is its own different sub-image [4].

Stage 4 – character recognition – This is where the characters in sub images made in segmentation are recognized. In [4], four methods are explained, ANN, Template matching, Tesseract OCR engine and Support Vector Machine (SVM). Template matching is the only recognition tool used in [5] [21]. On the other side, [11] used SVM and ANN as OCR tools to a 90% character recognition in wet condition. But because of the limited nature of template matching, the Tesseract OCR engine has an advantage for its high recognition accuracy and having the ability to retrain a character dataset [4].

1. ANPR tools and ways used

In [11], OpenCV was used as an image processing tool, SVM were used for contour detection on number plates and ANN were used for character recognition from number plates. The process they used had these stages: pre-processing, segmentation, morphological processing, number plate contour analysis, plate extraction and scaling, SVM, character segmentation, character contour analysis and then ANN. This mix of tools gave a 98.5% detection rate and a 90% character recognition rate.

In [12], the main tool they used was Faster R-CNN for number plate detection. They used template matching as an OCR way. The process went like this: needed database storage, input image capture, crop image to pick characters, change image to black and white, noise removal and comparing letters or numbers with template also called template matching. This got a 99.1% number plate detection accuracy.

In [14], a different way is used for ANPR which has three stages. The first is using the wavelet transform for contrast feature extraction. The reference line is then found, which is important in finding the wanted licence plate area. Lastly, licence plate adjustment is used in correctly finding the licence plate. This detection way was 92.4% right.

In [16], an edge detection algorithm was used for ANPR. It was based on things that vehicle licence plates (VLP) have like the height contrast to neighbour pixels that the contours of VLP characters have. Also, the contours of characters are closed always and the characters have a relationship. This way gave a standard measure of quality of 87.44%.

1. System Design
2. Functional Requirements
3. The system should detect the license plate of the vehicle entering the parking lot using ANPR with open CV.
4. The system should start and stop a timer for each vehicle based on the detection results.
5. The system should calculate the duration and cost of parking for each vehicle based on a predefined rate.
6. The system should deduct the cost from the balance of the customer's account linked to the license plate or generate an invoice for payment.
7. The system should generate reports on the occupancy, revenue, and payment status of the parking lot.
8. The system should allow the customer to log in to a platform using their license plate and view their transaction history and balance.
9. The system should allow the customer to pay online or offline using various payment methods.
10. Non-Functional Requirements
11. The system should be accurate and reliable in detecting and recognizing license plates under different lighting and weather conditions.
12. The system should be secure and protect the privacy and data of the customers and the parking owner.
13. The system should be user-friendly and provide a clear and intuitive interface for the customers and the parking owner.
14. The system should be efficient and fast in processing and transmitting data and generating feedback.
15. Architectural design



Fig. 1 A diagram showing the architectural design of Autonomous parking Bay

Fig 1 shows the architectural design of the autonomous parking bay system. It shows the hardware components and how they are all connected to each other.   
It shows how cameras will be connected to the detection system server which is connected to the universal database. The same database connects ParkPay (the payment system) and the Reporting system.

1. Use case diagram

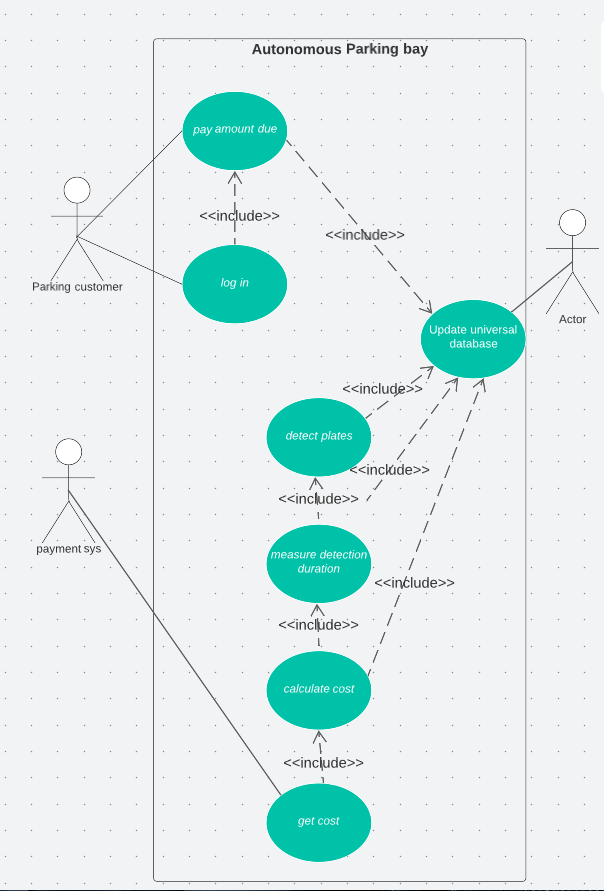


Fig. 2 Use case Diagram for Autonomous Parking Bay

1. Interface design

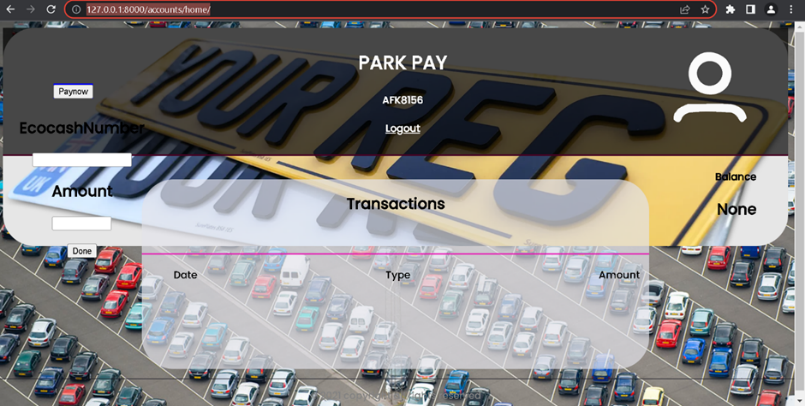


Fig. 3 Shows the interface of park pay

Fig 3 shows the interface design of ParkPay a major component of the Autonomous Parking Bay. ParkPay allows customers to pay for parking and also tracks their transactions on their own anywhere anytime. That means the parking attended won’t be needed to collect parking fees when such a system is present.

1. System Implementation
2. Tools used

The following tools were used to develop the Autonomous Parking Bay:

1. Image extraction – open cv & webcam
2. Image detection – Tensorflow /open cv
3. Image segmentation – easyOCR
4. Image recognition –EasyOCR
5. Duration measurement – python
6. Payment integration – Django framework and Paynow test API
7. Database – PostgreSQL
8. Frontend – HTML, CSS and JS
9. Detection system

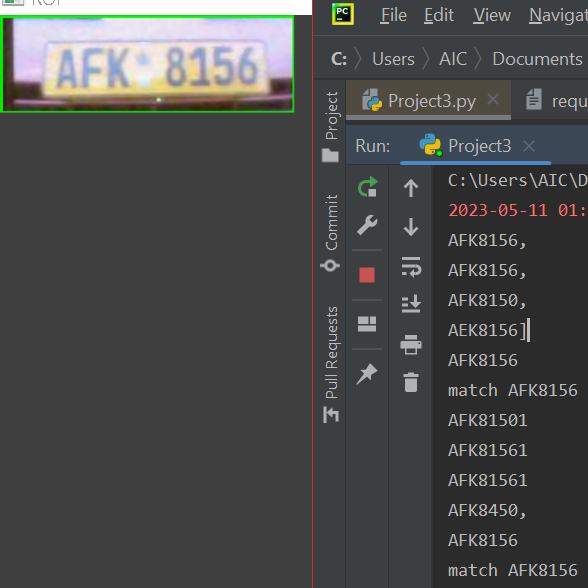


Fig. 4 Shows the detection system detecting plates

1. Reporting System

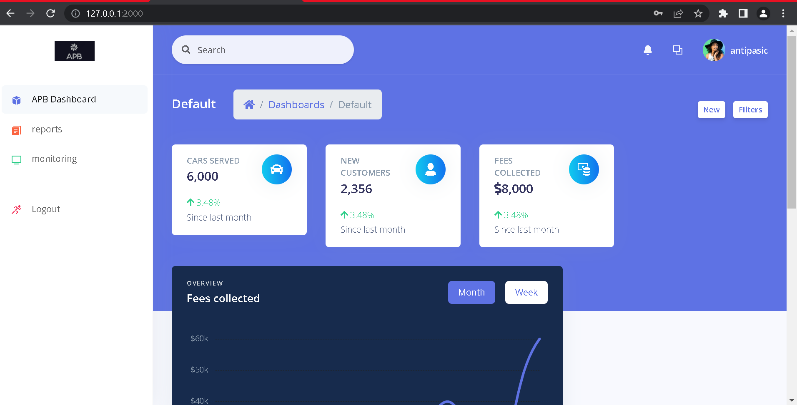


Fig. 5 Shows the reporting system dashboard which shows cars served, fees collected and new customers etc.

1. System Testing and Results

This section presents the testing of the overall system functionality and performance  
  
 Table 1 – test cases

|  |
| --- |
| **Test Case Description** |
| 1. Test if the system can detect and read the number plate of a car that parks in the parking bay and starts a timer. |
| 1. Test if the system can calculate the cost of parking based on the duration and rate and update the customer account with the cost. |
| 1. Test if the system can log in the user to the Payment System using the number plate and password or token and retrieve the transaction history for the customer account. |
| 1. Test if the system can pay for the parking using PayNow, a third party payment platform that is integrated with the Payment System and update the payment status in the Database. |
| 1. Test if the system can generate a report of the parking activity using the car service and fee information from the Database and display it to a dashboard for monitoring and analysis. |
| 1. Test if the system can handle multiple cars parking and leaving at different times and rates. |
| 1. Test if the system can handle invalid inputs such as invalid number plates, passwords, tokens, payment details, etc. |
| 1. Test if the system can handle errors such as Database not accessible, Payment System not available, PayNow not working, etc. |
| 1. Test if the system can meet the performance requirements such as response time, accuracy, reliability, scalability, security, etc. |

1. Results

Fig. 6 shows a chart of the system testing results

The fig 6 shows that the overall system testing result was **pass**, as all test cases met the acceptance criteria and all defects were resolved or accepted.

1. Conclusion

The current parking system is human intensive with two or more parking attendants at a single parking bay. They collect fees, note vehicle license plates and keep time. This system is therefore prone to errors, frauds and is generally inefficient. This was the basis for undertaking such a project to correct the inefficiencies.   
The solution is an Autonomous parking system that depends on ANPR and a universal database. The system comprises of four parts the payment system (ParkPay), the detection system, the reporting system and the universal database. The tools used to implement the system were OpenCV, EasyOCR, PostgreSQL, python and Django Framework.   
The system passed the system testing and has shown that there can exist an autonomous parking bay.

1. Acknowledgment

I would like to express my sincere gratitude to the following people who have supported me throughout this project:

My supervisor, Mr. Makondo, for his valuable guidance, feedback, and encouragement. He constantly checked up on us, enquired on our progress and demanded continuous improvement. He was a true mentor

Kelvin Zawala, for providing me with a stable and power internet connection that enabled me to experiment with different software and tools. He also assisted me with some technical problems and shared his insights and suggestions.

My family and friends, for their constant support and motivation. They have been very understanding and supportive of my efforts and challenges. They have also cheered me up and inspired me to keep going.

1. References
2. Shahid, Umar. (2017). NUMBER PLATE RECOGNITION SYSTEM. 10.13140/RG.2.2.23058.71361.
3. Menon, A., & Omman, B. (2018). Detection and recognition of multiple license plate from still images. 2018 International Conference on Circuits and Systems in Digital Enterprise Technology (ICCSDET). https://doi.org/10.1109/iccsdet.2018.8821138
4. Simon, M., Shibwabo, B., & Mutua, K. (2017). An automatic number plate recognition system for car park management. International Journal of Computer Applications, 175(7), 36–42. https://doi.org/10.5120/ijca2017915608
5. Connie, L., On, C. K., & Patricia, A. (n.d.). A review of Automatic License Plate Recognition System in mobile based platform. CORE. Retrieved November 1, 2022, from https://core.ac.uk/display/235221828?utm\_source=pdf&utm\_medium=banner&utm\_campaign=pdf-decoration-v1
6. Babu, K. M., & Raghunadh, M. V. (2016). Vehicle number plate detection and recognition using bounding box method. 2016 International Conference on Advanced Communication Control and Computing Technologies (ICACCCT). https://doi.org/10.1109/icaccct.2016.7831610
7. Xu, L., Shang, W., Lin, W., & Huang, W. (2021). License plate detection methods based on opencv. 2021 21st ACIS International Winter Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing (SNPD-Winter). https://doi.org/10.1109/snpdwinter52325.2021.00012
8. Lubna, Khan, M. F., & Mufti, N. (2016). Comparison of various edge detection filters for ANPR. 2016 Sixth International Conference on Innovative Computing Technology (INTECH). https://doi.org/10.1109/intech.2016.7845061
9. Aung, K. P., Nwe, K. H., & Yoshitaka, A. (2019). Automatic License Plate Detection System for Myanmar vehicle license plates. 2019 International Conference on Advanced Information Technologies (ICAIT). https://doi.org/10.1109/aitc.2019.8921286
10. Kuo, H.-C., Lee, J.-S., & Kao, S.-T. (2009). An autonomous license plate detection method. 2009 Fifth International Conference on Intelligent Information Hiding and Multimedia Signal Processing. https://doi.org/10.1109/iih-msp.2009.222
11. Jiménez Zozaya, L. (1970, January 1). Automatic Number Plate Recognition. Handle Proxy. Retrieved November 1, 2022, from https://hdl.handle.net/2454/7126
12. Singh, A. K., & Roy, S. (2015). ANPR Indian system using surveillance cameras. 2015 Eighth International Conference on Contemporary Computing (IC3). https://doi.org/10.1109/ic3.2015.7346695
13. Ap, N. P., Vigneshwaran, T., Arappradhan, M. S., & Madhanraj, R. (2020). Automatic number plate detection in vehicles using faster R-CNN. 2020 International Conference on System, Computation, Automation and Networking (ICSCAN). https://doi.org/10.1109/icscan49426.2020.9262400
14. Sedighi, S., Nguyen, D.-V., & Kuhnert, K.-D. (2019). Implementation of a parking state machine on vision-based auto parking systems for Perpendicular parking scenarios. 2019 6th International Conference on Control, Decision and Information Technologies (CoDIT). https://doi.org/10.1109/codit.2019.8820634
15. Ching-Tang Hsieh, Yu-Shan Juan, & Kuo-Ming Hung. (2005). Multiple license plate detection for complex background. 19th International Conference on Advanced Information Networking and Applications (AINA'05) Volume 1 (AINA Papers). https://doi.org/10.1109/aina.2005.257
16. Farooqi, N., Alshehri, S., Nollily, S., Najmi, L., Alqurashi, G., & Alrashedi, A. (2019). Uparking: Developing a smart parking management system using the internet of things. 2019 Sixth HCT Information Technology Trends (ITT). https://doi.org/10.1109/itt48889.2019.9075113
17. Anoual, H., Fkihi, S. E., Jilbab, A., & Aboutajdine, D. (2011). Vehicle license plate detection in images. 2011 International Conference on Multimedia Computing and Systems. https://doi.org/10.1109/icmcs.2011.5945680
18. Mohammed Shariff, A. S., Bhatia, R., Kuma, R., & Jha, S. (2021). Vehicle number plate detection using Python and open cv. 2021 International Conference on Advance Computing and Innovative Technologies in Engineering (ICACITE). https://doi.org/10.1109/icacite51222.2021.9404556
19. M. Y. Aalsalem, W. Z. Khan and K. M. Dhabbah, "An automated vehicle parking monitoring and management system using ANPR cameras," 2015 17th International Conference on Advanced Communication Technology (ICACT), 2015, pp. 706-710, doi: 10.1109/ICACT.2015.7224887.
20. B. Gunay, M. Saltan, A. Ilgaz and Y. Albayrak, "Area-wide ANPR coverage with a small number of instrumented vehicles," 2015 International Conference on Models and Technologies for Intelligent Transportation Systems (MT-ITS), 2015, pp. 379-386, doi: 10.1109/MTITS.2015.7223283.
21. M. Y. Aalsalem and W. Z. Khan, "CampusSense — A smart vehicle parking monitoring and management system using ANPR cameras and android phones," 2017 19th International Conference on Advanced Communication Technology (ICACT), 2017, pp. 809-815, doi: 10.23919/ICACT.2017.7890230.
22. M. C. Sai, D. Chandwani and S. Bhava, "Advanced Vehicle Monitoring System with Multi-Object Automatic Number Plates Detection," 2019 Global Conference for Advancement in Technology (GCAT), 2019, pp. 1-4, doi: 10.1109/GCAT47503.2019.8978321.