



Article

A Personal Microcomputer as an Access Control Management Platform in Road Transport

Przemysław Wójcik  and Tomasz Neumann * 

Faculty of Navigation, Gdynia Maritime University, 81-225 Gdynia, Poland; p.wojcik@wn.umg.edu.pl

* Correspondence: t.neumann@wn.umg.edu.pl

Abstract: For many years, the use of new computer systems to control various elements of everyday human life has been observed. Separate systems manage access control; others are used to control blinds and roller shutters; and others manage systems in the garden. Many of these systems can be integrated using available systems. This paper presents an example of an access control management system based on the Raspberry Pi microcomputer and shows an analysis of its performance, accuracy, and possibility of improvement. This study used official devices manufactured by the Raspberry Pi Foundation; however, it is possible to create a similar system with custom parts. This project used open-source software. The authors argued that it is possible to create an autonomous vehicle access control system using microcomputers and optical character recognition technology. Using simple devices, the plate recognition system was built and tested, proving the thesis that it is possible to build an access control system using available devices. This also confirms the thesis that microcomputers can be used to control other systems in the human environment.

Keywords: transportation; microcomputer; access control; Optical Character Recognition; Raspberry Pi; licence plate recognition



Citation: Wójcik, P.; Neumann, T. A Personal Microcomputer as an Access Control Management Platform in Road Transport. *Appl. Sci.* **2023**, *13*, 9770. <https://doi.org/10.3390/app13179770>

Academic Editors: Daniela Anna Misul, Gabriele Di Blasio and Alfredo Gimelli

Received: 25 June 2023

Revised: 18 August 2023

Accepted: 21 August 2023

Published: 29 August 2023



Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Autonomous and smart management systems have been used in an increasing number of projects over the past few years. Technologies like optical character recognition (OCR) have been implemented in access control devices, smart building systems, and more. Furthermore, scientific progress has created the possibility of constructing compact devices capable of handling advanced systems. Microcomputers, such as Raspberry Pi or Arduino, present computing power similar to that of some older stationary computers while using less power, materials, and space. This allows for decreased costs of building and maintaining some systems and a decreased amount of energy needed for them to work, potentially making it easier to implement some solutions using renewable energy while maintaining the desirable performance of a particular system.

This paper presents an example of an access control management system based on the Raspberry Pi microcomputer and shows an analysis of its performance, accuracy, and possibility of improvement. In this project, official devices manufactured by the Raspberry Pi Foundation were used; however, it is possible to create a similar system with custom parts. This project used open-source software.

2. Literature Review

In recent years, the ever-expanding Internet of Things (IoT) has become more empowered to revolutionise our world with the advent of cutting-edge features and intelligence in an IoT ecosystem. The distributed nature of the IoT and its rapid increase on a large scale raise many security and privacy issues. Access control is one of the major challenges currently addressed through centralised approaches that may rely on a third party, and they are constrained by availability and scalability, which may result in a performance

bottleneck. A novel solution to manage the delivery of lightweight and decentralised secure access control for an IoT system based on a multi-agent system and a blockchain has been proposed by [1]. Problems related to access control are very often discussed in the professional literature. The growing availability of mobile devices has led to the rising development of smart city services that share a huge amount of (personal) information and data. Without accurate and verified management, they could become severe backdoors for security and privacy. In this paper, we propose a smart city infrastructure able to integrate a distributed privacy-preserving identity management solution based on attribute-based credentials [2]. In [3], the authors propose a rule-based optimisation mechanism on an embedded edge platform to provide dynamic home appliance control and advanced intelligence in a smart home. The latest research includes those concerning buildings with nearly zero energy consumption [4,5]. One study affirmed that smart homes are not generally considered a safety-critical domain because the consequences of the artificial intelligence (AI) system's failure in a smart home setting are typically less severe than in other safety-critical domains such as transportation or healthcare [6]. In [7], the authors reviewed recent activities related to IoT-based energy systems. The work highlighted the potential areas to improve at different layers and reviewed communication technologies and standards related to energy systems. Some examples were discussed, including smart homes, smart power grids, and smart cities. Equally often, microcomputers are used to facilitate basic human activities. In [8], the authors demonstrate the capabilities of Raspberry Pi to perform electroencephalogram (EEG) biometric tasks in real-time by developing a portable, low-cost, real-time system that integrates all the necessary steps of subject identification from the acquisition of the EEG signals. The authors of [9] provide an analysis of the possibilities of using a Raspberry Pi microcomputer on a ship, while an overview of the components for an onboard control system of an autonomous ship is presented in [10]. In [11], a Raspberry Pi microcomputer embedded with image processing algorithms is introduced to realise edge-computing. In [12], the study thoroughly examines how blockchain-based decentralised architecture can potentially improve IoT access management. The authors of [13] propose combining zero-trust access control with an attribute-based encryption scheme against compromised devices in power IoT environments. Traditional access control schemes assume that internal devices are trusted in power IoT environments, thus giving compromised devices the chance to steal sensitive data [14]. A brief overview shows that the problem of control is an important one that is often described nowadays. New technologies and solutions are important elements that provide new opportunities. One such approach is presented in this paper.

3. Methodology for the Development of an Access Control Management System

An access control management system should be able to detect vehicles waiting for access, check linked databases for the existence of a detected vehicle's plate numbers, and, depending on the results, grant access and open the barrier or deny access and give proper information to the driver of the rejected vehicle. The system should also avoid closing the barrier when a vehicle is under the raised arm of the barrier.

The main goal of this project was to create the above-described system with the lowest possible cost and, at the same time, the lowest energy consumption. The designed system should be capable of fulfilling the conditions mentioned above with a reasonable rate of success. The accepted values for particular conditions are a 100% success rate in preventing the barriers from closing down when something is under the arm of the barrier and at least a 51% success rate in the proper reading of plate numbers. The time for response when the vehicle arrives should not be more than 10 s.

The motivations behind the values specified above are the following:

- Preventing the system from closing the barriers when something is under them is a matter of safety—any error may lead to accidents, vehicle, cargo, or infrastructure damage, and, in the worst cases, damage to health. Because of that, any rate below 100% is unacceptable.

- Even top OCR engines do not have 100% accuracy. When working on high-quality documents, the accuracy rate is usually around 99% per page. In describing the project, the OCR input would not be a scan but a picture taken in different weather and light conditions. The objects containing the searched data may also be seen in the particular images from slightly different angles. In those cases, the expected accuracy should be much lower. However, if the success rate turns out to be lower than 50%, it usually means that the system does not work.
- One of the main reasons behind automation is to reduce the time needed for a particular operation. In manually checking if a vehicle should have permission to enter, one should observe the car, visually inspect the plate numbers, write them in some sort of data browser, and, depending on the results, decide whether this vehicle does or does not have access to the particular area. With good perception and practical usage of cameras, computers and browsers should be able to do this quickly, but there is a relatively low probability of doing this in less than 10 s. Because of that, it seems to be a reasonable value for the automatic system to be considered effective.

Many OCR engines are available nowadays, some of which are open source and free, and others are commercial products. The paid versions of this software are usually better in speed, accuracy, and overall performance. Since the technology of OCR has been improving for decades, modern versions of free software usually also reveal acceptable-quality work. The additional goal of this project is to check if it is possible to create the described system using free open-source software and fulfil the necessary conditions mentioned above.

4. Creating an Access Control System

4.1. Device Selection

As mentioned before, the computer and the camera used in this paper were Raspberry Pi-branded devices. However, it should also be possible to maintain a similar system based on some versions of Arduino or other single-board computers. Although there would be some differences in details, such as the programming language or way of connecting components, the overall algorithm of the program would be the same. Microcomputers from the main branch of Raspberry Pi devices have one particular advantage: they can handle ‘desktop-like’ operating systems, such as Raspberry Pi OS, Ubuntu MATE, Windows IoT Core, and more. This functionality extends the computer’s software power and makes working with it much more manageable. That is the main reason behind the decision to choose the Raspberry Pi as the foundation of the described project [15–17].

There are several generations of Raspberry Pi microcomputers, with multiple versions and models in each. The main branch contains four generations of devices. The one used in this project is Raspberry Pi 4 Model B with 2 GB of RAM (Raspberry Pi Foundation, Cambridge, UK). Older or more compact versions of Raspberry Pi should also be possible to use in this project, as long as they can work with the camera and have enough memory to perform some image processing. Some tests were undertaken with the previous-generation Raspberry Pi 3 Model B with 1 GB of RAM to check if it would have some influence on the performance. The results will be mentioned in the next chapter of this paper.

The camera used in the described project is an official Raspberry Pi High-Quality Camera with a 12 MPx Sony IMX477R Sensor (Tokyo, Japan), equipped with a 16 mm telephoto lens designed for that camera. The other official lens for the described camera, a 6 mm wide-angle lens, would be a better option depending on the planned distance between the camera and the monitored entry gate in the final form of the developed system. In this project, that distance was about 4 m. The 6 mm lens would be preferred when the space is lower than 4 m. It should also be possible to successfully use the older Raspberry Camera modules, especially since the sufficient resolution of processed images is 640×480 pixels. The overall costs may be lower, and there is no need to buy additional lenses, but there may be problems with purchase availability. It is also safe to assume that the latest Raspberry Pi Camera module generation—version 3, which came out in January—would also work. Regardless of choice, the camera connects to Raspberry Pi via a

CSI flat cable. The standard length of this cable is 30 cm. Because of that, the microcomputer has to be close to the camera. Placing it in the same case is a good option.

There are various ways of checking if the vehicle is below the arm of the opened barrier. One of the most popular options is to use induction loops. These devices installed on the surface of the road use electromagnetism to detect all metal objects placed above them. The significant advantages of this solution are its reliability and lifetime. However, the setup process generates significant costs due to the need for the road's surface modifications. It is possible to lower the overall cost of the described system by using a different object detection method. It has been used as an ultrasonic distance sensor in this project. It is a device capable of emitting and receiving high-frequency sound signals. Considering the time between signal emission and echo reception, it is possible to calculate the distance between the sensor and the detected obstacle. The sensor should be placed on the entry gate, under the arm of the barrier. With that, if the sensor detects an obstacle in a specified range, it would mean that something is under the component of the barrier, and the system should prevent it from closing down.

The devices used in this project are as follows:

- Raspberry Pi 4 Model B with 2 GB of RAM;
- Raspberry Pi HQ camera with a Sony IMX477R sensor;
- PT3611614M10MP 16 mm lens for the Raspberry Pi HQ camera;
- Ultrasonic distance sensor HC-SR04 (Cytron Technologies, Johor, Malaysia)

To provide safe communication between the Raspberry Pi microcomputer and the HC-SR04 ultrasonic sensor, resistors must be used. The described distance sensor communicates with Raspberry Pi through the GPIO port. However, the HC-SR04 is a 5 V device, and the GPIO port in Raspberry Pi microcomputers works with a voltage of 3.3 V. The echo pin of the distance sensor transfers the signal directly to the GPIO port of Raspberry Pi, so it must be lower before. To achieve this, two resistors must be used. The proportion of their resistance is calculated in the following equation:

$$U_o = U_z \times \frac{R_2}{R_1 + R_2} \quad (1)$$

$$3.3[\text{V}] = 5[\text{V}] \times \frac{R_2}{R_1 + R_2} \quad (2)$$

$$1.94R_1 = R_2 \quad (3)$$

where

U_o —desirable voltage of the signal entering the Raspberry Pi;

U_z —voltage of the signal exiting the HC-SR04 sensor;

R_1 —resistance of the first resistor;

R_2 —resistance of the second resistor.

The resistance values of the resistors used in this project were 1000 Ω and 500 Ω . It should fix the voltage value at 3.3 V. The connection diagram is illustrated in Figure 1.

Raspberry Pi 4's average power consumption is about 3 W. The maximum power usage is no more than 7 W. The older or more compact version of this microcomputer uses even less energy. For comparison, a typical mini PC uses between 7 and 40 W (there are some models with power consumption similar to or slightly above that of Raspberry Pi microcomputers; however, they are significantly more expensive), and a standard desktop office computer consumes about 150 W. A typical single-socket server uses up to 118 W. The required database could be stored directly in the storage memory of the microcomputer, and the server would serve only for backup purposes, which could reduce its power consumption. It is possible to make some extra power savings by using renewable energy. For example, the microcomputer could be equipped with a solar panel. To avoid damages related to the voltage fluctuations caused by differences in light hitting the board at a

particular time, the device should also be equipped with a battery, charge controller, and DC/DC converter. However, this modification is optional and was not used in the described project.

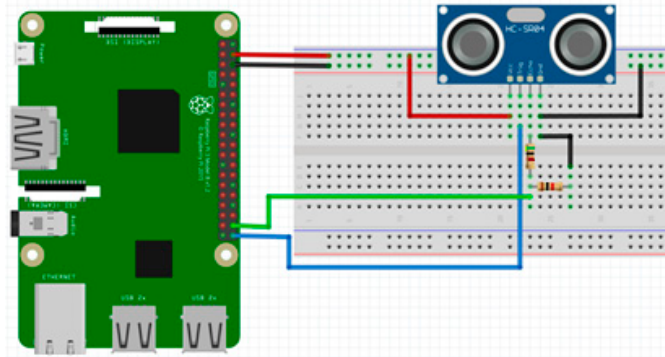


Figure 1. Schema of the proper connection between Raspberry Pi and the HC-SR04 ultrasonic distance sensor.

4.2. Software

As mentioned before, the additional goal of this project was to check the possibility of basing the described system on free, open-source software. The final version of the developed system contains two scripts in the Python language. The first of them controls the overall action algorithm and the work of barriers and ultrasonic sensors. The second is responsible for the camera, image processing, character recognition, and comparing the results with the database. To carry out such a comparison, the SQLite API for Python was used. The databases used in the project were written in SQL. The schema of the algorithm of the first of the Python scripts is pictured in Figure 2.

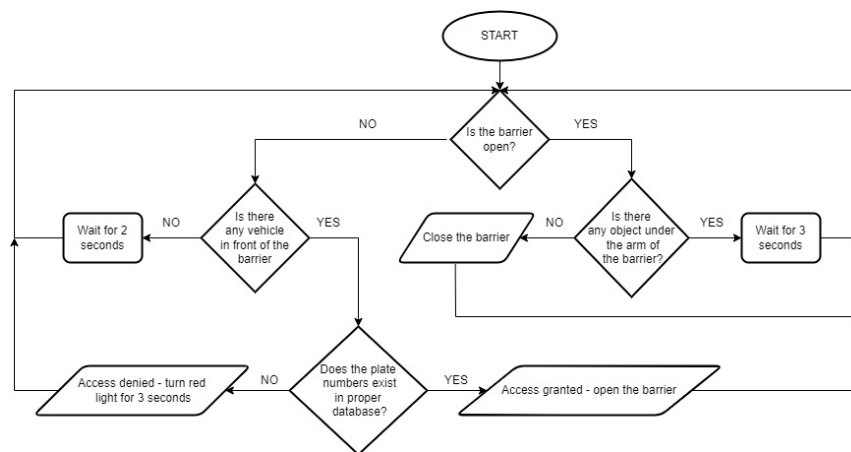


Figure 2. Schema of the algorithm of the first script.

The second script is used as a subprocess when the barrier is open. It checks if there is any vehicle in front of the entrance and, if so, searches the proper database for the existence of its plate numbers. The results are returned to the first script, and the appropriate action is taken according to them. The schema of the algorithm of the second script is illustrated in Figure 3.

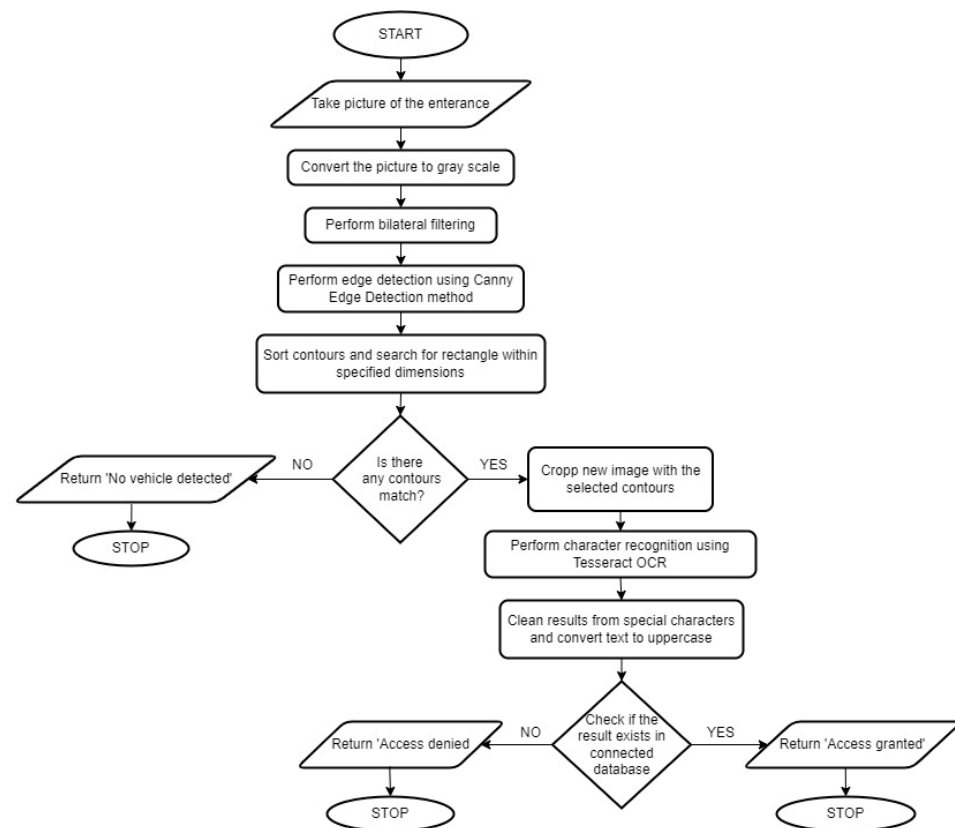


Figure 3. Schema of the algorithm of the second script.

Most of the image processing is performed with the OpenCV library. The script converts the image to greyscale, reduces noise by performing bilateral filtering, detects edges using the Canny Edge Detection method, and tries to find objects that could be the licence plate. If the program finds such an object, it isolates it and performs character recognition with the Tesseract OCR engine. Because the individual photographed vehicle may be at a slightly different angle relative to the camera and the picture may be taken during different weather and light conditions, there is a high probability of some errors occurring; however, it is also possible to prevent some of them. The most common bugs are mistakenly adding special characters and using lowercase. None exist on the licence plates of any country, so the described script deletes all special characters from the results and changes every letter to uppercase. After these actions, the final result is compared to the specified database. The correct information is returned to the first script based on the consequences of the second script's actions.

The ultrasonic sensor was installed as described earlier and set up to prevent barriers from closing down when it detects any objects between 10 and 160 cm. This range was chosen by comparing the possible vehicle width with the typical entrance width.

Another version of the second script has been prepared to facilitate the installation process. It works similarly to the original, although it shows a camera preview and performs all actions after the trigger by pressing the specified key. The preview helps set up camera properties and check if the image part with the licence plate is taken correctly. Figure 4 presents a Raspberry Pi desktop with a running help script.

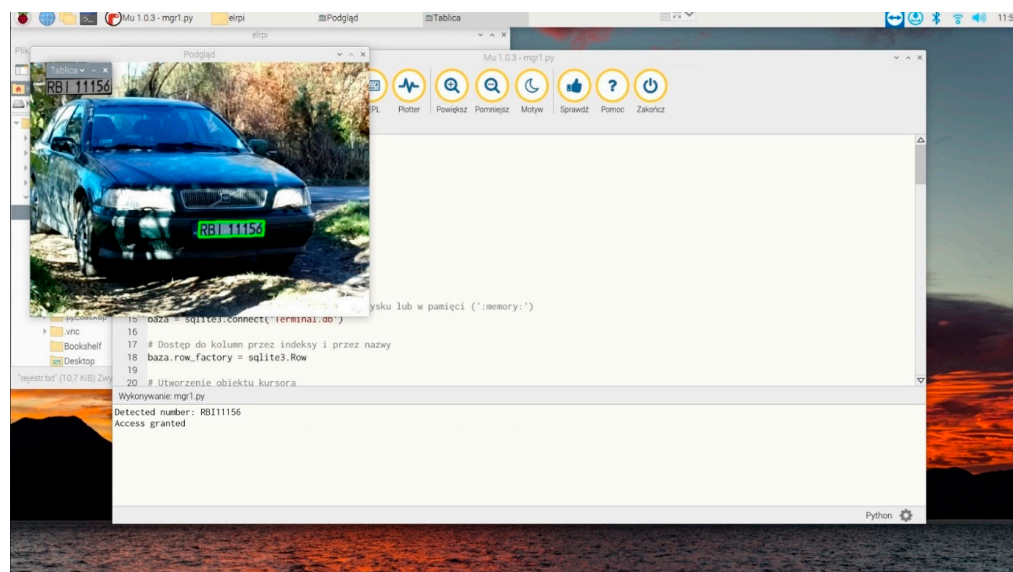


Figure 4. Raspberry Pi desktop running a help script. The program correctly determines licence plate position, recognises characters, and checks permission in the database.

4.3. Cost

Raspberry Pi Model B's current (Q1 2023) price with 2 GB of RAM oscillates around USD 50 among approved resellers. However, the Raspberry Pi Foundation has had some problems with supply chains for the past two years. This leads to frequent unavailability in official stores and increased prices at third-party resellers. Even though the fee should not exceed USD 100. The more compact versions of Raspberry Pi are proportionally cheaper. The camera module used in this project costs around USD 120 with a 16 mm lens included. The latest camera module, v3 is even more affordable, oscillating around USD 30. The HC-SR04 ultrasonic sensor used in this project usually costs around USD 2.

The overall cost of the device used in the described project was less than USD 200; with the camera module v3, it could be less than USD 100. However, this sum does not include the cost of the optional case and wires.

5. Analysis of the Results

The system was developed for EU-standard licence plates. The tests were performed with 20 different paper licence test plates, of which 3 were confirmed and 17 were fake, created mainly for this purpose and stuck on the real plates. To check if the background colour impacts the OCR accuracy, three fake plates had a green background (simulating the EV's licence plates), and another three had a yellow background (simulating licence plates from countries like, e.g., The Netherlands). Three false numbers were chosen to make it especially hard to read by the OCR engine, including combinations of letters and digits that are very similar, like '0' and 'O', '1' and 'l'. The other numbers were chosen randomly from the possible European combinations. All tests were carried out on three different vehicles, differing in shape and colour. It seems not to have had any significant impact on the results. The tests were performed in different weather and lighting conditions. The regular illumination of the licence plate affects the chance of detecting the plate. All tests included 899 detections of the licence plates. Table 1 shows the results.

The background colour of the cells in Table 1 corresponds to the background colour of each licence plate. The first three records are the actual plates. The overall success ratio is around 66%. Of 899 attempts, 591 ended up with correct recognition. The background's colour seems to have no significant impact on the result. As might have been expected, numbers with combinations of similar letters and digits were almost always recognised incorrectly. Randomly chosen numbers are usually identified correctly. However, to

increase the value of these statistics, records with a small number of tries should be removed. Table 1 includes only records with 40 or more attempts.

Table 1. Results of plate detection with at least 40 tries.

Licence Plate	Correct Recognition	Incorrect Recognition	Tries	Success Rate [%]
RBI11156	25	15	40	62.50
RBI04840	11	56	67	16.42
WE984KV	79	24	103	76.70
GD241AD	36	13	49	73.47
CB448ES	49	14	63	77.78
KR776JZ	83	23	106	78.30
ZS225GY	86	4	90	95.56
DW375UV	47	13	60	78.33
NOLKT40	40	21	61	65.57
KRK71U2	42	21	63	66.67
	498	204	702	70.94

Table 1 contains two real and eight false plates, of which one is green and one is yellow. The coloured ones show a success ratio around average, which in this case is almost 71%. RBI04840 has the lowest success rate in the table above. It was the real plate with the little fatigued numbers. A similar case was found with RBIH096 in Table 2, although fewer trials existed. The correct recognisability coefficients of the remaining licence plates are identical and oscillate around the average.

Table 2. Results of all plate detections.

Licence Plate	Correct Recognition	Incorrect Recognition	Tries	Success Rate [%]
RBI11156	25	15	40	62.50
RBIH096	4	16	20	20.00
RBI04840	11	56	67	16.42
GWE72Y9	33	5	38	86.84
WE984KV	79	24	103	76.70
RBI11R7	17	4	21	80.95
PO070OP	0	25	25	0.00
DW111IB	0	20	20	0.00
PZL31I7	2	12	14	14.29
GD241AD	36	13	49	73.47
CB448ES	49	14	63	77.78
KR776JZ	83	23	106	78.30
ZS225GY	86	4	90	95.56
DW375UV	47	13	60	78.33
WWARH99	7	6	13	53.85
GKA12I2	9	1	10	90.00
NOLKT40	40	21	61	65.57
KRK71U2	42	21	63	66.67
TOSCB70	16	13	29	55.17
BI446FH	5	2	7	71.43
	591	308	899	65.74

In most cases, the first recognition after the vehicle arrives is incorrect, but the second and subsequent ones are successful. Even lightning on the plate has a significant effect on the results. When the licence plate is illuminated only partially, the system usually does not detect the plate at all. However, the most problematic issue is the appearance of similar letters and digits in contiguous positions.

Both Tables 1 and 2 show the average success ratio above design assumptions. On this basis, it can be confidently stated that the developed system works as intended. However, it must be noted that some combinations of plate numbers have a 0% chance of correct recognition. This is still a minority, and there are several ways to deal with this, which will be discussed later in this article.

In all tests, there was no situation where the barrier closed when something was under the barrier arm. This confirms that the ultrasonic distance sensor works as intended and that its detection range was correctly set.

The maximum time for each iteration is 9 s. However, it still takes 4 s to detect the licence plate, check the database, and turn on the signal to open the barrier. To check whether the database size affects the record search time, four different databases were prepared. All of them had one table, but they had an additional number of records. The first had 24 records, the second had 1000 records, the third had 10,000 records, and the last had over 100,000 records. No differences in processing time were detected in any of the created databases. In each case, the maximum time for each iteration was exactly 9 s, as set in the script code.

6. Comparative Analysis, Recognition Problems and Future Development

6.1. Comparative Analysis of the Similar Devices Available on the Market

The idea of using OCR technology in road transport is not new. Devices specialised in licence plate recognition (LPR) have been on the market for a long time and are becoming more popular each year. Most modern systems are much more powerful and accurate than the solution presented in this paper. Many of them can detect multiple objects at once and read the licence plates of vehicles moving at speeds up to 200 km/h. Examples of such systems are the LPR/ANPR cameras for vehicle access control delivered by the Supervision company, solutions from Neural Labs (e.g., Neural Edge Urban), or the products and services from the company Milesight. Many models of surveillance cameras developed by various companies feature LPR.

The problem with the solutions mentioned above is their high overall price. The cost of a single LPR camera is usually around USD 500 and rises depending on the model's capabilities, accuracy, and overall quality. For example, the Amcrest LPR camera available on Amazon costs above USD 700. With more powerful and complex solutions, the licence could cost even more (e.g., a licence for the Neural Edge Urban solution for one camera costs EUR 1029, and the camera itself is not included). Some companies do not even show their prices because they take every transaction individually and calculate the cost.

Compared to the devices mentioned above, the system presented in this paper is much cheaper. As previously mentioned, its overall cost is less than USD 200 and could be even lower, while the price of the single LPR camera available on the market is usually more than twice that (without the cost of the computer). The system presented in this paper is certainly not as powerful as the devices mentioned above. Still, it fulfils its purpose and proves that it is possible to maintain a simple access control platform on a microcomputer.

6.2. Recognition Problems

All the main and additional goals of the described project have been fulfilled. It is possible to create an autonomous vehicle access control system using microcomputers and OCR technology. Despite the lower computing power compared to classic desktop computers, devices such as Raspberry Pi can efficiently operate on large databases, process images, and quickly analyse the data. However, it is not yet a perfect system. Although the average accuracy exceeds the minimum requirements, there are still some problems. The first is the lightning conditions—the entire surface of the licence plate must be illuminated evenly. This condition has to be implemented in the design process during the infrastructure implementation of the described system.

Another, and perhaps the biggest, problem is the misrecognition of registration numbers caused by pairs of similar letters and digits in adjacent positions. Some rare but

possible combinations of licence plate numbers may have an almost 0% chance of being correctly recognised by the system and consequently will not be allowed to enter a restricted area, even if they should be allowed to. This problem can be solved in two ways. The first is related to the idea of artificial intelligence and machine learning. The OCR engine used in the described project—Tesseract—is equipped with training tools. It brings the possibility of improving its accuracy, especially when working with specific fonts and the visual format of the image. Training consists of recognising text and comparing the results with the “ground truth”—pairs of text images and files specifying each character and its exact position in the picture. The training program runs through a predetermined number of iterations and modifies the text recognition process with each iteration to lower the error rate and thus achieve better accuracy. The results should be more proportional to the number of samples and iterations, but the training process would take much longer.

However, the training process does not guarantee 100% accuracy. Optical character recognition and image processing are complex processes, especially when the input images are so different. There is always the possibility that some unforeseen errors will occur. The second and easiest way to solve this problem is to add an intercom to the system that enables communication between the driver and the person responsible for access control. This would ensure that the inspection process would be automatic most of the time. Still, in the event of a problem, the relevant person can manually check if the vehicle can enter the premises. This solution is currently used in all similar systems and will be used until the technology can maintain 100% accuracy and prevent errors.

6.3. Possible Future Development

The OCR engine used in the presented project—Tesseract—is constantly improving. Version 5 was released in 2021, and the last patch was released in July 2023. Considering the constant growth of the capabilities and improvement of the accuracy of this engine for over 20 years, it is safe to assume that in the future, its overall quality will be better. However, it is already possible to improve the accuracy of the described system. Tesseract is considered to be one of the best free, open-source OCR engines, but it is not the only one. Perhaps another piece of software could improve the overall accuracy of such a system. It also seems highly probable that the commercial, paid software would be better. Future research may find an optimal solution for creating a similar system with the advantages of current high-end devices and without significant cost growth. If that happens, LPR technology may become more accessible even for small companies and entrepreneurs, which should help them save certain resources that could be invested in further improvements and growth of the company.

7. Conclusions

Considering the results from the previous chapter, it should be concluded that it is possible to develop an OCR vehicle access control system based on microcomputers and open-source software. Such a system maintains the necessary conditions to be helpful in both personal and commercial projects. However, the accuracy of the system described could be improved, and further research should be conducted to explore the potential for improvements without increasing overall costs. Shortly, it may lead to a reduction in power consumption and the costs of installing and maintaining similar autonomous systems.

Author Contributions: Conceptualisation, T.N. and P.W.; methodology, P.W.; software, P.W.; validation, T.N.; formal analysis, T.N.; resources, P.W.; writing—original draft preparation, P.W.; writing—review and editing, T.N.; supervision, T.N.; project administration, T.N.; funding acquisition, T.N. All authors have read and agreed to the published version of the manuscript.

Funding: This study was funded by the Gdynia Maritime University, under research project WN/2023/PZ/07.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Algarni, S.; Eassa, F.; Almarhabi, K.; Almalaise, A.; Albassam, E.; Alsubhi, K.; Yamin, M. Blockchain-Based Secured Access Control in an IoT System. *Appl. Sci.* **2021**, *11*, 1772. [CrossRef]
2. Daoudagh, S.; Marchetti, E.; Savarino, V.; Bernabe, J.B.; García-Rodríguez, J.; Moreno, R.T.; Martínez, J.A.; Skarmeta, A.F. Data Protection by Design in the Context of Smart Cities: A Consent and Access Control Proposal. *Sensors* **2021**, *21*, 7154. [CrossRef] [PubMed]
3. Xu, R.; Jin, W.; Hong, Y.; Kim, D.-H. Intelligent Optimization Mechanism Based on an Objective Function for Efficient Home Appliances Control in an Embedded Edge Platform. *Electronics* **2021**, *10*, 1460. [CrossRef]
4. Chekired, F.; Taabli, O.; Khellili, Z.M.; Tilmatine, A.; de Almeida, A.T.; Canale, L. Near-Zero-Energy Building Management Based on Arduino Microcontroller—On-Site Lighting Management Application. *Energies* **2022**, *15*, 9064. [CrossRef]
5. Eirale, A.; Martini, M.; Tagliavini, L.; Gandini, D.; Chiaberge, M.; Quaglia, G. Marvin: An Innovative Omni-Directional Robotic Assistant for Domestic Environments. *Sensors* **2022**, *22*, 5261. [CrossRef] [PubMed]
6. Dobrovolskis, A.; Kazanavičius, E.; Kižauskienė, L. Building XAI-Based Agents for IoT Systems. *Appl. Sci.* **2023**, *13*, 4040. [CrossRef]
7. Shakerighadi, B.; Anvari-Moghaddam, A.; Vasquez, J.C.; Guerrero, J.M. Internet of Things for Modern Energy Systems: State-of-the-Art, Challenges, and Open Issues. *Energies* **2018**, *11*, 1252. [CrossRef]
8. Benomar, M.; Cao, S.; Vishwanath, M.; Vo, K.; Cao, H. Investigation of EEG-Based Biometric Identification Using State-of-the-Art Neural Architectures on a Real-Time Raspberry Pi-Based System. *Sensors* **2022**, *22*, 9547. [CrossRef] [PubMed]
9. Neumann, T. The Single-Board Computer As a Tool to Measure the Weather Parameters in the Marine Areas. *TransNav Int. J. Mar. Navig. Saf. Sea Transp.* **2020**, *14*, 901–906. [CrossRef]
10. Yanchin, I.; Petrov, O. Towards Autonomous Shipping: Benefits and Challenges in the Field of Information Technology and Telecommunication. *TransNav Int. J. Mar. Navig. Saf. Sea Transp.* **2020**, *14*, 611–619. [CrossRef]
11. Wan, S.; Zhao, K.; Lu, Z.; Li, J.; Lu, T.; Wang, H. A Modularized IoT Monitoring System with Edge-Computing for Aquaponics. *Sensors* **2022**, *22*, 9260. [CrossRef] [PubMed]
12. Singh, I.; Singh, B. Access Management of IoT Devices Using Access Control Mechanism and Decentralized Authentication: A Review. *Meas. Sens.* **2023**, *25*, 100591. [CrossRef]
13. Huang, W.; Xie, X.; Wang, Z.; Feng, J.; Han, G.; Zhang, W. ZT-Access: A Combining Zero Trust Access Control with Attribute-Based Encryption Scheme against Compromised Devices in Power IoT Environments. *Ad Hoc Netw.* **2023**, *145*, 103161. [CrossRef]
14. Neumann, T. Enhancing Safety and Reduction of Maritime Travel Time with In-Vehicle Telematics. In *Communications in Computer and Information Science*; Springer: Berlin/Heidelberg, Germany, 2018; Volume 897.
15. LPR Cameras for Vehicle Access Control. Survision. Available online: <https://survisiongroup.com/post-lpr-cameras-for-vehicle-access-control> (accessed on 13 August 2023).
16. License Neural Edge Urban—Urban LPR and Lists Management. Available online: <https://www.neurallabs.net/en/neural-store/license-neural-edge-urban> (accessed on 13 August 2023).
17. The Best License Plate Recognition Security Camera. Nelly's Security. Available online: <https://www.nellyssecurity.com/blog/articles/video-surveillance/best-license-plate-recognition-camera-nsc-lpr832-bt1> (accessed on 13 August 2023).

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.