e-Parking System in Corporate Parking Systems by Implementing Optical Character Recognition

Juan Farell Haryanto[a], Justin Mudita Kristian[a], William Sunjaya[a]

[a] *Computer Science Department, School of Computer Science,*

Bina Nusantara University, Jakarta, Indonesia

Abstract

The COVID-19 pandemic has made the society to do mandatory health and safety protocols. The drivers are also required to take part in these protocols to minimize the virus spread. The automated parking systems can help to minimize the direct human contact in the processes of parking transactions. Optical Character Recognition can be used to create an automated parking system in corporations and offices. This is useful for offices as this is where the virus has a high rate of infections among the employees. Tesseract and OpenCV are used to implement these Optical Character Recognition systems. By using Tesseract and OpenCV, the algorithm is able to achieve the rate of accuracy of 73.75%, precision of 82.54%, recall of 83.87%, and the F1 score of 83.2%. The usage of Optical Character Recognition is able to increase the general security and improve the safety protocol for touchless systems. The implementations of these can also achieve efficiency and to decrease the complicated usage of physical parking receipts in the parking systems.

1. Introduction

During the midst of the COVID-19 pandemic, the obligation of social distancing and lockdown has been highly implemented. With social distancing between people, everyone living in a society needs to keep their distance from each other which also includes keeping any form of contact with devices that share contact with other people. As in the current state, the virus can spread from the shared human devices. One of these devices that most people very often interact with is the parking devices. Parking devices can be found anywhere and are surely used by drivers and riders. From figure 1.1, in 2018 the amount of passenger cars alone in Indonesia reached to the number of 14 million[1]. The numbers will surely increase every year and it will also increase the amount of contact between drivers and riders who share this. The issue is also frequent in the scope of corporations' parking systems. The issue of the ineffective parking system in general is also characterized by the physical parking ticket. Which is frequently lost or easily discarded. Hence, it is important to be able to automatically read the driving license plate to reduce the use of physical parking tickets, and also reduce the risk of spreading COVID-19 at the same time.

The solution to the issue is by making a program that is a web-based app. The technology that will be implemented is Optical Character Recognition. Optical Character Recognition is an innovation that empowers the change of scanned paper reports into editable and accessible content information (Laique et al., 2021). There are several Optical Character Recognition employment opportunities in library settings (Hahn, 2014). From prior research, it is done by taking a photo of the license plate and scanning it using Optical Character Recognition from the photo to generate the characters in the license plate (Vaishnav & Mandot, 2019). This is just a basic update of a system that has been widely used. In late years, Optical Character Recognition has attracted broad consideration in the field of computer vision, medical care applications, smart transport systems, historical manuscript restoration and image to speech translation, and so on (Das & Mohanty, 2021). It has a significant role in the financial area where an optical reader is utilized to extract important data printed on any credit card, bank passbook, or scanned forms. Afterwards, it has been effectively applied in most character recognition situations and shows solid tolerance to internal failure (Li et al., 2020). As a significant course in design acknowledgment, the use of Optical Character Recognition has more prominent prospects. This paper aims to implement the entirety of the improved Optical Character Recognition scanning techniques exclusively in corporate's parking system. However this time, the system will be able to categorize whether this vehicle is one of their employees or not, based on their driving license plate. If it is an employee's vehicle (registered in database), then the vehicle will be allowed to enter the parking area. However, if the vehicle was not registered in the corporate's database, then the vehicle will not be allowed to enter the corporate building. Afterwards, the database also will receive the exact time when the vehicle entered and left the building. By this scenario, physical parking tickets, such as paper-based or card, won't be needed anymore when the users park their vehicle.

Parking is one of the central elements for any corporate employees, since there are a lot of employees who drive their own cars or ride their own bikes to arrive at each of their own corporations. For instance, there are some corporations that are quite far from the bus station or train station, and it makes some employees think that by driving their own car or riding their own bikes. While they bring their own vehicle, of course they also need a parking lot which is also safe and comfortable. To tackle these problems, an automated program that is going to recognize and process a vehicle's number plate will be proposed in this paper.

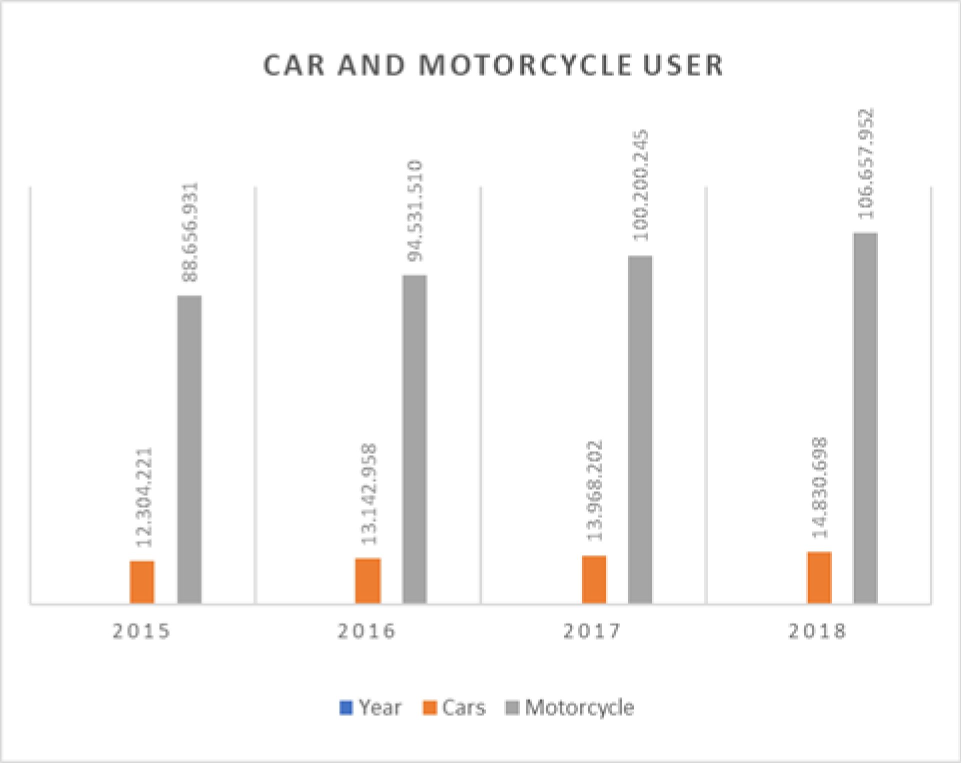


Figure 1. Comparison of Cars and Motorcycle users.

2. Literature Review

2.1 Overview

Optical Character Recognition, or also known as OCR, is used to convert scanned or handwritten text/images into editable characters[2]. Optical Character Recognition has many forms of implementations to be able to analyze datasets. The various implementations such as optical scanning, location segmentation, preprocessing, and others can provide accurate analysis[3]. The usage of Optical Character Recognition also has many collaborations with other algorithms to provide a much-improved accuracy. One such research implements it with deep neural networks, which is very useful to read writings with a heavy noise on the surface of the writing[4]. Other than that, is by using the K-Nearest Neighbor approach to the Optical Character Recognition program, which will calculate the algorithm accuracy[5]. Additionally, another study aims to summarize research that has been conducted on character recognition of handwritten documents and to provide research directions[6]. There is also another study which already implements Optical Character Recognition as the solution to simplify humans' work. On the other hand, another study that relates to analysing financial bills[7], implements optical character recognition to automate and improve the existing banking processes, which can be achieved through automatic cheque processing[8]. From a pool of sample cheques that were used for testing, an accuracy of 95.71% was achieved. Other uses of information processing is by using Optical Character Recognition for automating vehicle licenses[9] and for marketing purposes such as filtering promotion offers[10].

Optical Character Recognition has also branched to reading numerous language character analysis other than Latin characters, such as Arabic for Quranic scanning to check the similarity and the originality of it[11] and Hindi language writings[12]. Publicly available datasets also include stimuli that are aligned well with each other since all the languages which have been used as samples have achieved accuracy above 86%. Furthermore, there is another study which utilizes Optical Character Recognition related to document segmentation and language translation and it has shown that any document whose image is available can be read and translated by means of some python scripting and which will ultimately help anyone to understand it in his or her known language[13]. These can vary despite the type of writing that is used as the samples. Optical Character Recognition can identify letters regardless of if they are handwritten, typed, printed, or molded.

2.2 Image Preprocessing

In terms of pre-processing, it will be useful to produce the data that are easy for the Optical Character Recognition systems to operate in the most accurate way[3]. Optical Character Recognition is able to extract distinct features from the input image for classifying its contents as characters. Several conditions may affect optical character recognition execution like the resolution of the picture, the quality of the picture, or the color of the background[14]. Image pre-processing can be helpful by having methods such as noise reductions, which is achieved by de-noising the image themselves by using wavelet transform[15], normalization which is used to obtain standardized data. Normalization can be conducted manually by reviewing samples like colonoscopy procedure reports to ensure the collected information on variables was accurate[16]. The last process is compression which is used to reduce the storage information and therefore providing which much speeder processing times. Another research conducts a similar technique that uses five steps such as image alignment, extraction of region of interests, noise reduction, threshold management, and segmentation[17]. Although it is a big challenge for researchers to develop Optical Character Recognition capability[18].

Optical Character Recognition is organized in two subsections, with the first one introduces the character recognition system for cosmetic-related advertising images by implementing the region of interest (ROI) for identifying the text box, and the second one describes the text detection and recognition system for natural scenes by implementing Raspberry Pi[19], which is connected to a camera, and a sensor[20]. The recognition results reveal that the proposed method was highly effective at extracting text from images and had a 93% accuracy rate. In order to improve the Optical Character Recognition's performance, four different image preprocessing methods are implemented, namely image resizing, image sharpening, image blurring, and separating the text from the background. Generally, there are many image formats such as GIF, JPEG, PNG, etc.[21]. This type of format will be used as the input of the Optical Character Recognition's system. The output of printed text can be extracted using Tesseract[22] and all the results are compared based on various parameters. Hardware that could be used for the research also varies. The proposed preprocessing techniques improve Optical Character Recognition execution by 33.3% (Tesseract 3.5) and 22.6% (Tesseract 4.0) on the accessible sample pictures[23].

2.3 Pattern Analysis

With enough data samples provided from preprocessing, the samples are now provided for further analysis including classification. The classification of both high quality and low-quality data are needed to be put into attention. As images such as faces or license plates tend to have a lower accuracy rate in the Optical Character Recognition process in the lower lighting areas[24]. For license plates, Automatic Number Plate Recognition (ANPR) uses Optical Character Recognition to classify the alphanumeric characters from an image[25]. For face recognition, it could be more challenging. As a lower quality image will make the facial features of a person to be difficult to be determined. The other way to make result predictions to be more accurate is by separating the part of the dataset to be used for testing the accuracy of the predictions[26]. This is helpful to prevent false positives when running the tests due to the part of having a set of images that are not a part of the sample images known to the model.

Regarding the Automatic Number Plate Recognition process, firstly, the video camera is started, reads and resizes the frames. Afterwards, filters are used to find the image contours and a bilateral filter is applied for removing noises[27]. Other methodologies are able to be involved, such as Binary Image Processing, Adaptive Thresholding, Contrast Extension, Median Filtering, Character Segmentation, Feature Extraction, and finally Optical Character Recognition by use of template matching[28]. The algorithm accuracy is 83%, the filters Gaussian Blur precision is 80%, and Filter 2D achieves 72% accuracy.

3. Methodology

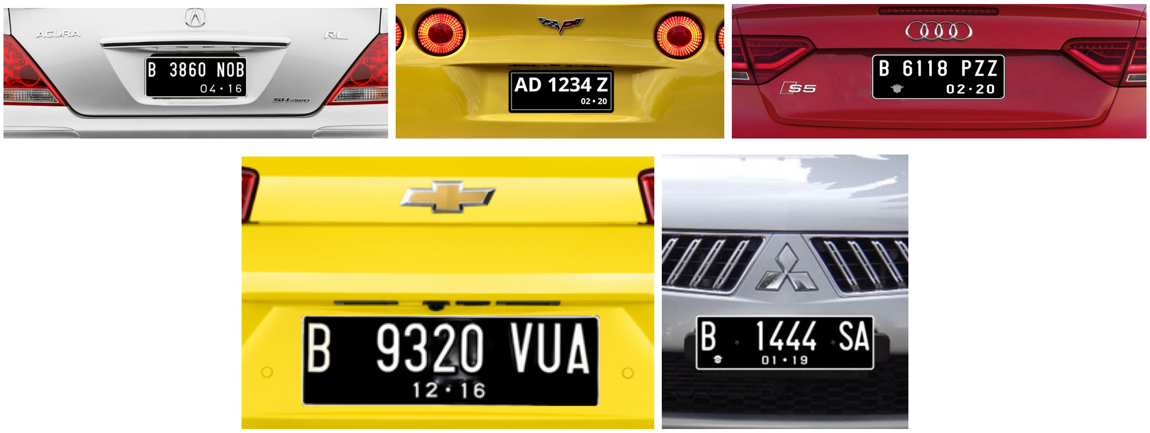


Figure 2. Examples of dataset.

The samples of data that were used in this research are in the form. Meanwhile, to achieve our goal which is to recognize every vehicles' license plate that enters a corporation area or complex, then it will check in the corporation's database (including staff and student) whether it is recorded in the database or not. This is because of using the text that is contained on the license plate as the main reason to inform whether an image that the camera captured is valid to enter the building or not. In addition, Tesseract is used as one of the best implementations of Optical Character Recognitions to detect a character or text, and of course it is open source. Tesseract is used as the open source for the Optical Character Recognition because Tesseract is easy to implement, has a good amount of extensions, and flexibility. Tesseract also has a community that is active. Other than Tesseract, OpenCV is used for image preprocessing such as reading the file and processing it. Just like Tesseract, OpenCV is also easy to implement.

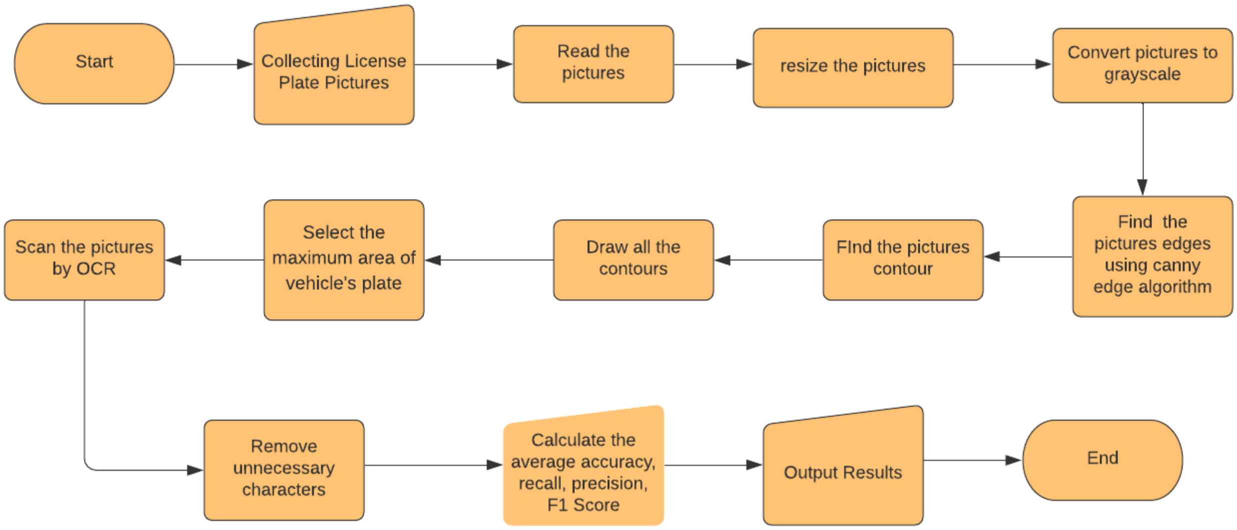

Start
Collecting License Plate Pictures
Read the pictures
resize the pictures
Convert pictures to grayscale
Find the pictures edges using canny edge algorithm
FInd the pictures contour
Draw all the contours
Select the maximum area of vehicle's plate
Scan the pictures by OCR
Remove unnecessary characters
Calculate the average accuracy, recall, precision, F1 Score
Output Results
End

Figure 3. Flowchart

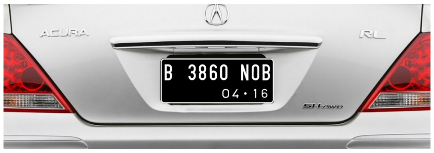
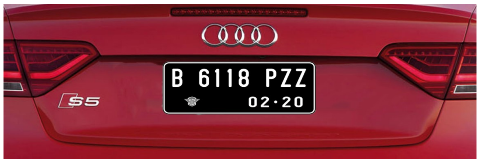
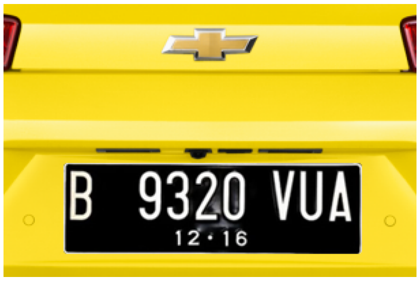
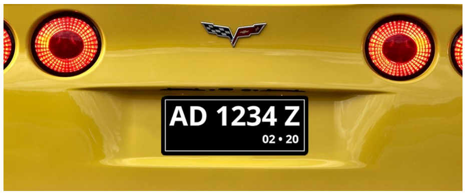
3.1 Dataset

The sample of the data that have been gathered for this research are gathered from various resources. Around 5 images found that were used as data samples due to the limited resources and time due to the COVID-19 pandemic. From the sample that have been gathered, it's categorized into two types, which are employee or not employee. Only half of them are employee's vehicles, while the rest of them are not registered in the data asset.

3.2 Preprocessing

After gathering some sample data, it is important to preprocess the image before all of them are able to be utilized. Pytesseract is used, which is an Optical Character Recognition library in Python programming language, to extract the text or character from the vehicle's plate image. Tesseract is an interesting library which is contained in Python language, since it can detect several texts. In addition, Tesseract is not only able to recognize roman characters, however it is also able to detect Chinese and Japanese since the third version, and Arabic and Hebrew since the fourth version[29].

Several libraries that are available to create the Optical Character Recognition system such as pytesseract, and OpenCV. The process starts by reading the image file, while afterwards it will be resized and converted to a grayscale image. This process is essential because the Canny Edge Algorithm works on grayscale images. The next process is finding edges of the image and the image's contours. In order to crop the image to just the vehicle's plate part, the maximum area which is the top 25 contours of the image are taken. Lastly, it will go through the image preprocessing process which eliminates unnecessary characters such as the car's logo, the license plate's expiry date, and car accessories such as bumper stickers, back window attachments, and the output is printable characters of the vehicle's plate.

Raw images	Results from Optical Character Recognition	Results after Preprocessing
	B 3860 NOB 04-16 ♀	B 3860 NOB
	B 6118 PZZ . 02-20 ♀	B 6118 PZZ
	B 9320 VUA J ♀	B 9320 VUA
	AD 1234Z 02+20 ♀	AD 1234 Z

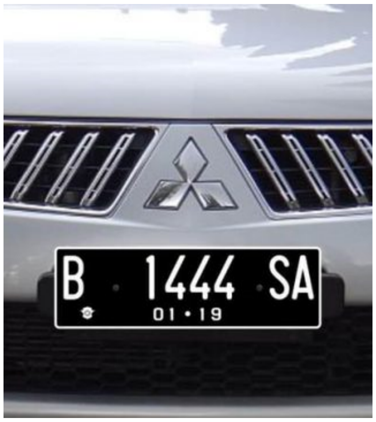
	B 1444 SA ♀	B 1444 SA

Table 1. Result of Image Preprocessing

3.3 Classification

From the result that has been gathered from previous stages, the system will check the vehicle's plate number, whether it's registered as an employee or not in the database since the system is connected with the Corporation Data Assets. It will be classified as an employee's vehicle if the vehicle's license plate is registered. In addition, the vehicle's will be allowed to enter the corporation's building. Meanwhile, if the vehicle's license plate is unable to be recognized by the system, the vehicle will not be allowed to enter the building since it's classified non-employee to prevent something bad from happening.

4. Result and Discussion

Real Text	**Results from Optical Character Recognition**	**True Positive**	**False Positive**	**True Negative**	**False Negative**
B 3860 NOB 04-16	B 3860 NOB 04-16 ♀	13	0	1	0
B 6118 PZZ 02-20	B 6118 PZZ . 02-20 ♀	13	0	2	2
B 9320 VUA 12-16	B 9320 VUA	8	5	2	4

	J ♀				
AD 1234 Z 02-20	AD 1234Z 02+20 ♀	11	1	1	1
B 1444 SA 01-19	B 1444 SA ♀	7	5	1	3
D 6568 LM 03 - 24	D 6568 LM ♀	7	5	1	0
A 2157 WTH 02 - 25	A 2157 WTH ♀	8	5	1	3
F 9485 NC 03 - 24	F 9458 NC ♀	7	5	1	0
B 514 EXP 02 - 25	B 514 EXP 02 25 ♀	11	1	1	0

B 6746 ACD 03 - 24	B 6746 ACD ♀	8	5	1	0
B 189 EEX 12 - 24	B 189 EEX ♀	7	5	1	2
A 357 NBJ 03 - 25	A 357 NBJ 03.25, ♀	11	1	2	1

Table 2. Dataset Testing

True Positive is an outcome where the model recognizes the relevant text[30]. On the other hand, True Negative is an outcome where the model recognizes the irrelevant text. Meanwhile, False Positive is an outcome where the model does not recognize the relevant text. Additionally, False Negative is an outcome where the model does not recognize the irrelevant text.

Algorithm	Accuracy	Precision	Recall	F1 Score
Tesseract OCR	70%	74.5%	87.4%	80.43%

Table 3. Result of the Test

From the table 3 above, it shows the accuracy, precision, recall, and F1 Score of the Optical Character Recognition by implementing Tesseract after performing several processes using the dataset that has been collected before. The result will be precise if the image has great lighting conditions and a perfect framing placement. This will also be better if the image quality is also Full HD. The Optical Character Recognition result will also work better if the license plate's surface itself does not gleam. The Optical Character Recognition is further improved if the scanned license plate has a template of a white line surrounding the license plate. In addition, as seen on dataset testing above, half of the license plates' expiry date are unable to be recognized by the algorithm, hence it also affects the algorithm's accuracy. However, it is understandable that the results of accuracy, precision, recall, and F1 Score above might not show the full potential of the techniques used since the dataset used for this paper is still lacking. This was caused by the limited time and resources available during the pandemic to collect more dataset at the time of the study. Although the test result above might not show superior results, it is crucial to implement optical character recognition as soon as possible since everyone is obliged to do anything without direct contact or touchless during these pandemic, including the parking system.

5. Conclusion

In conclusion, Optical Character Recognition is able to resolve our issues in implementing the automatic parking system for corporations in Indonesia. By using Tesseract and OpenCV, the algorithm is able to achieve the rate of accuracy of 70%, precision of 74.5%, recall of 87.4%, and the F1 score of 80.43%. The usage of Optical Character Recognition is able to increase the general security and improve the safety protocol for touchless systems. The available systems are still able to be improved in the future to be better, and the Optical Character Recognition technology has a potential to be useful in many more aspects in the future.

References

[1] Badan Pusat Statistik (BPS). "Perkembangan Jumlah Kendaraan Bermotor Menurut Jenis, 1949-2018". Retrieved from https://www.bps.go.id/linkTableDinamis/view/id/1133 on March 4th, 2021.

[2] B Hamad, K., & Kaya, M. (2016). A Detailed Analysis of Optical Character Recognition Technology. International Journal of Applied Mathematics, Electronics and Computers, 4(Special Issue-1), 244. https://doi.org/10.18100/ijamec.270374

[3] Chaudhuri, A., Mandaviya, K., Badelia, P., & Ghosh, S. K. (2016). Optical Character Recognition Systems. Studies in Fuzziness and Soft Computing, 9–41. doi:10.1007/978-3-319-50252-6_2

[4] Wei, T. C., Sheikh, U. U., & Rahman, A. A.-H. A. (2018). Improved optical character recognition with deep neural network. 2018 IEEE 14th International Colloquium on Signal Processing & Its Applications (CSPA). doi:10.1109/cspa.2018.8368720

[5] Ong, V., & Suhartono, D. (2016). Using K-Nearest Neighbor in Optical Character Recognition. ComTech: Computer, Mathematics and Engineering Applications, 7(1), 53–65. https://doi.org/10.21512/comtech.v7i1.2223

[6] Memon, J., Sami, M., Khan, R. A., & Uddin, M. (2020). Handwritten Optical Character Recognition (OCR): A Comprehensive Systematic Literature Review (SLR). IEEE Access, 8, 142642–142668. https://doi.org/10.1109/access.2020.3012542

[7] Li, H., Huang, C., & Gu, L. (2020). Image pattern recognition in identification of financial bills risk management. Neural Computing and Applications, 33(3), 867–876. https://doi.org/10.1007/s00521-020-05261-3

[8] Srivastava, S., Priyadarshini, J., Gopal, S., Gupta, S., & Dayal, H. S. (2018). Optical Character Recognition on Bank Cheques Using 2D Convolution Neural Network. Advances in Intelligent Systems and Computing, 589–596. https://doi.org/10.1007/978-981-13-1822-1_55

[9] Mohammed, B. N., & Ahmad, H. B. (2021). Advanced car-parking security platform using Arduino along with automatic license and number recognition. Academic Journal of Nawroz University, 10(1), 1. https://doi.org/10.25007/ajnu.v10n1a996

[10] Hubert, Phoenix, P., Sudaryono, R., & Suhartono, D. (2021). Classifying Promotion Images Using Optical Character Recognition and Naïve Bayes Classifier. Procedia Computer Science, 179, 498–506. https://doi.org/10.1016/j.procs.2021.01.033

[11] Alotaibi, F., Abdullah, M. T., Abdullah, R. B. H., Rahmat, R. W. B. O. K., Hashem, I. A. T., & Sangaiah, A. K. (2018). Optical Character Recognition for Quranic Image Similarity Matching. IEEE Access, 6, 554–562. https://doi.org/10.1109/access.2017.2771621

[12] Bairagi, P. P., & Dulal, G. (2018). Optical character recognition for Hindi. International Research Journal of Engineering and Technology, 5(5), 3968-3973.

[13] Thakare, S., Kamble, A., Thengne, V., & Kamble, U. (2018). Document segmentation and language translation using tesseract-ocr. 2018 IEEE 13th International Conference on Industrial and Information Systems (ICIIS). doi:10.1109/iciinfs.2018.8721372

[14] Hazra, T. K., Singh, D. P., & Daga, N. (2017). Optical character recognition USING KNN on custom image dataset. 2017 8th Annual Industrial Automation and Electromechanical Engineering Conference (IEMECON). doi:10.1109/iemecon.2017.8079572

[15] Veni, S., Sabeenian, R., Shanthi, T., & Anand, R. (2020). Real time noisy dataset implementation of optical character identification using CNN. International Journal of Intelligent Enterprise, 7(1/3), 67–80. https://doi.org/10.1504/ijie.2020.10026346

[16] Laique, S. N., Hayat, U., Sarvepalli, S., Vaughn, B., Ibrahim, M., McMichael, J., Qaiser, K. N., Burke, C., Bhatt, A., Rhodes, C., & Rizk, M. K. (2021). Application of optical character recognition with natural language processing for large-scale quality metric data extraction in colonoscopy reports. Gastrointestinal Endoscopy, 93(3), 750–757. https://doi.org/10.1016/j.gie.2020.08.038

[17] Rizvi, M., Raza, H., Tahzeeb, S., & Jaffry, S. (2019). Optical Character Recognition Based Intelligent Database Management System for Examination Process Control. 2019 16th International Bhurban Conference on Applied Sciences and Technology (IBCAST), 500–507. https://doi.org/10.1109/ibcast.2019.8667127

[18] Islam, N., Islam, Z., & Noor, N. (2017). A survey on optical character recognition system. arXiv preprint arXiv:1710.05703.

[19] Su, Y., Peng, H., Huang, K., & Yang, C. (2019). Image processing technology for text recognition. 2019 International Conference on Technologies and Applications of Artificial Intelligence (TAAI). doi:10.1109/taai48200.2019.8959877

[20] Menon, D. R., Keerthika, P., Madonna, N., Nandhini, S., & Jayanthi, A. N. AUTOMATIC NUMBER PLATE RECOGNITION SYSTEM USING RASPBERRY PI AND PYTHON.

[21] Sahu, N., & Sonkusare, M. (2017). A study on optical character recognition techniques. The International Journal of Computational Science, Information Technology and Control Engineering, 4, 1-14.

[22] Park, J., Lee, E., Kim, Y., Kang, I., Koo, H. I., & Cho, N. I. (2020). Multi-Lingual Optical Character Recognition System Using the Reinforcement Learning of Character Segmenter. IEEE Access, 8, 174437–174448. https://doi.org/10.1109/access.2020.3025769

[23] Brisinello, M., Grbic, R., Pul, M., & Andelic, T. (2017). Improving optical character recognition performance for low quality images. 2017 International Symposium ELMAR. doi:10.23919/elmar.2017.8124460

[24] Zaheen, M. Y., Mohi-u-din, Z., Siddique, A. A., & Qadri, M. T. (2020). Exhaustive Security System Based on Face Recognition Incorporated with Number Plate Identification using Optical Character Recognition. January 2020, 39(1), 145–152. https://doi.org/10.22581/muet1982.2001.14

[25] Vaishnav, A., & Mandot, M. (2019). Template Matching for Automatic Number Plate Recognition System with Optical Character Recognition. Information and Communication Technology for Sustainable Development, 683–694. https://doi.org/10.1007/978-981- 13-7166-0_69

[26] Bazerque, A., Moraes, D., & Souza, M. (2020). Using Object Detection Algorithm and Optical Character Recognition to Read Data from alphanumeric tags in text.

[27] Dalarmelina, N. D. V., Teixeira, M. A., & Meneguette, R. I. (2019). A Real-Time Automatic Plate Recognition System Based on Optical Character Recognition and Wireless Sensor Networks for ITS. Sensors, 20(1), 55. https://doi.org/10.3390/s20010055

[28] Kashyap, A., Suresh, B., Patil, A., Sharma, S., & Jaiswal, A. (2018). Automatic number plate recognition. 2018 International Conference on Advances in Computing, Communication Control and Networking (ICACCCN). doi:10.1109/icacccn.2018.8748287

[29] Rosebrock, A. (2020, September 24). Tesseract OCR for Non-English Languages. PyImageSearch. https://www.pyimagesearch.com/2020/08/03/tesseract-ocr-for-non-english-languages/

[30] *A simple guide to building a confusion matrix*. (2020, November 11). Oracle. https://blogs.oracle.com/ai-and-datascience/post/a-simple-guide-to-building-a-confusion-matrix