Comparison Using OCR for Car Plate Recognition

Erlangga Rizal Mahendra
Computer Science
Binus University
Jakarta, Indonesia
erlangga.mahendra@binus.ac.id

Michelle Angela Guntjoro
Computer Science
Binus University
Jakarta, Indonesia
michelle.guntjoro@binus.ac.id

Oliver Rian Setiono
Computer Science
Binus University
Jakarta, Indonesia
oliver.rian@binus.ac.id

Abstract

Optical Character Recognition (OCR) is a method that is currently being developed. However, there are several variations that affect the accuracy of this plate recognition technology, such as type, color, and font size. At present Plate recognition is used as an example of the development of an electronic system by the Indonesian National Police (POLRI) by initiating the ETLE or Electronic Traffic Law Enforcement program. To find out which method is the best with several variations that affect it. Then a step is taken, namely by comparing the three OCR methods from PytesseractOCR, EasyOCR and KerasOCR. We evaluate and compare these three methods with 27 Indonesian car license plates. We built a model to recognize Car Plate Recognition with KerasOCR which is the best method in this experiment to find out the best method, which is implemented to recognize Car Plates.

Keywords: LisencePlate Recognition, EasyOCR, TesseractOCR, KerasOCR, OCR, Car Plate Recognition,

I. INTRODUCTION

Vehicle improvement began in the late 1900s [1] where the invention of the first vehicle that could lead humans to their destination more quickly. Including in Indonesia itself, quoted from the Central Bureau of Statistics every year the increase in cars, both private and public cars, has experienced an increase in the growth in the number of vehicles. Between 2015 and 2020, the number of cars increased from 16,646,387 to 21,114,412 [2]. This growth is in line with population growth and daily mobilization needs. With this population growth, the number of vehicles used on the streets has also increased.

The increase in the number of vehicles, especially cars on this highway, often causes traffic violations to be committed, for example speeding beyond the standard speed and not using a safety belt. Therefore, with the developing of the current technological revolution, a vehicle plate recognition system has begun to be developed which is captured from the camera, but this recognition system cannot be said to be easy, there are several things that cause errors in license plate recognition, such as there are many variations on license plates. for example type, colour, and font size cause problems during the process of identifying license plates [3].

So, we need a system that can recognize license plates correctly. This can be seen by the example of the development of an electronic fine system by the Indonesian National Police (POLRI) by initiating the ETLE or Electronic Traffic Law Enforcement program as a breakthrough in the context of electronic traffic law enforcement. [4]. The development of this system aims to give violators a penalty in the form of a ticket and eliminate the manual ticket system. The ETLE system utilizes an image recognition system obtained from street cameras to obtain the license plate numbers of traffic violators.

With the widespread use of the camera system which is currently being developed, so as not to miss or make mistakes in recognizing license plates, an appropriate method is needed to read license plates to identify license plates. Identification of license plates is very important and interesting to discuss. In addition to the traffic system, this recognition system is also used for various other purposes such as in parking lots, borders, ports and airports. [1]Therefore, to find which plate recognition system is the best in terms of effectiveness and efficiency, we compared three methods, namely Easy OCR, Keras OCR and Tesseract OCR. Where the process is carried out is by using an image preprocessing technique that takes the vehicle plate number [5] and outputs it in the form of a series of digital characters. This license plate identification system utilizes image processing technology to identify vehicles from license plate images [6] . The dataset that we are trying for this experiment is the car dataset. Where our group takes itself by shooting and does not use datasets from sources. With experiments using Easy OCR[7], KerasOCR[8], and Tesseract OCR[9], it can give the best results regarding which method should be used to overcome this problem.

II. LITERATURE REVIEW

2.1. An Evaluation of Various Pre-trained Optical Character Recognition Models for Complex License Plates

This research discusses the analysis of the three methods of OCR to detect License Plate Numbers. OCR works by converting images that have characters into digital text, namely by using Pytesseract, Easy OCR, and Keras OCR. These three models were used as Pre-trained OCR models to evaluate their performance in recognizing characters on car plates from a collection of datasets. The results obtained are that Easy OCR and Pytesseract experience limitations in recognizing two texts and confusion in reading a character. This limitation is overcome by the KerasOCR method. but KerasOCR also cannot recognize license plates that are blurry and the characters are very close together. This paper explains that the accuracy of the pre-trained model with KerasOCR is 40.53%, where the accuracy rate is 4% higher than Pytesseract and 7.5% higher than EasyOCR[10]

2.2. Recognition Design of Lisence Plate and Car Type Using Tesseract OCR and EmguCV.

This paper discusses the use of Tesseract OCR and EmguCV to detect plate numbers and the types of cars detected. The advantage of the Tesseract OCR method is that the program can recognize license plates from four-wheeled vehicles with an accuracy of 80.223% with a speed of less than 500 milliseconds. All edges can be recognized as a square shape so that the license plate can be recognized properly. But the weakness of the Tesseract OCR method in this paper is that it cannot detect modified or damaged plates, for example

faded plates, cannot detect all plate sizes, only certain length and width sizes can be detected perfectly, the detection of the type of car must take the process long training, and the introduction of the type of car must be on the picture in the appropriate position.[11]

2.3.Automatic License Plate Recognition with YOLOv5 and Easy-OCR method

In this paper, in detecting the license plate, we use the Easy-OCR and YOLOv5 methods. In this paper it is explained that Easy-OCR cannot be said to be similar to the human eye. But the human eye can see the original image source easily. Whereas EasyOCR itself has to separate the characters from the background and with Easy OCR the more sophisticated the original image quality is. In an experiment conducted by the author of this paper, it was revealed that EasyOCR can provide an accuracy of around 95%.[12]

2.4.Leveraging Transfer Learning and GAN Models for OCR from Engineering Documents

In this paper. The author is explained that KerasOCR and Easy OCR are difficult to handle mixed documents such as textual styles and fonts. For TesseractOCR to work well on quality and well-structured documents. Word recognition accuracy increased significantly from 10.71% obtained with the KerasOCR pretrained model to 57.14%. So it can be said that the KerasOCR model is a good method for training texts than the EasyOCR and TesseractOCR models which have lower accuracy.[13]

III. METHODOLOGY

In this work, we build a system to detect car plates by comparing 3 methods of OCR.

A. Dataset

Before comparing the three OCR methods. dataset preparation is done first. The dataset that we use is not too much, because this dataset will grow according to the photos that will be uploaded by the user

1. Dataset preparation
   The dataset that we use is a collection of datasets using images of Indonesian license plates.
2. Data collection
   We took pictures of the vehicle with the Oppo A92 cellphone camera with a resolution of 64 mega pixels which took 27 license plate photos with different cars.

B. Image preprocessing

The preprocessing stage aims to process the input image. This is done to remove inconsistencies, redundancies, and noise in the image[14] so that the quality of the image can be increased to enter the next processing stage[15].

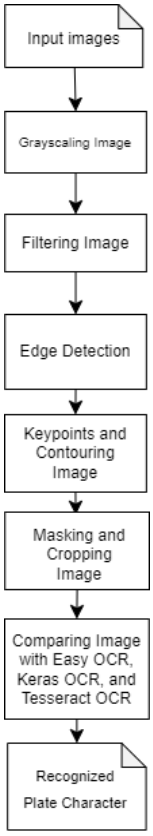

Figure 1. Steps to recognize the car plate character

1. Grayscaling Image
   Grayscaling is used to change the RGB image on car picture which taken from Figure 2 into 2D image which has 8bit with range 0 up to 255 like image in the Figure 3. This is used so that the machine can read the image better. After grayscaling.

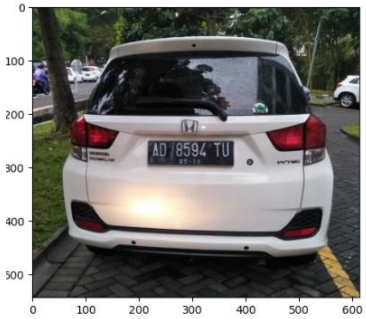

Figure 2. Original image (RGB)

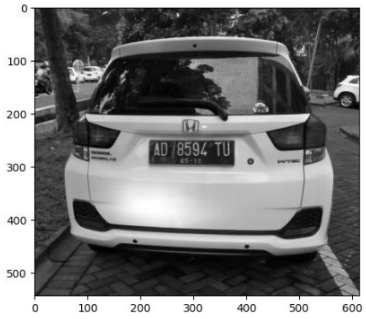

Figure 3. Gray scale image

2. Filtering Image
   Images captured directly (captured with a camera) may have a lot of noise, so the image will be filtered using bilateral filter for reducing the noise inside the picture, so that the image does not lack any information from the image[16].

3. Edge Detection
   The next step after filtering is edge detection, the edge detection method used in this research is Canny Edge Detection. This is used to detect the edge of object in the image. This method has several stages to finally be able to find edge[17] as in Figure 4. The steps taken include noise reduction/smoothing to eliminate noise and prevent spurious edges, gradient calculation to calculate horizontal edges, non-maximum suppression to eliminate edges. twins, edge linking returns to connect the broken lines due to the non-maxima suppression process.

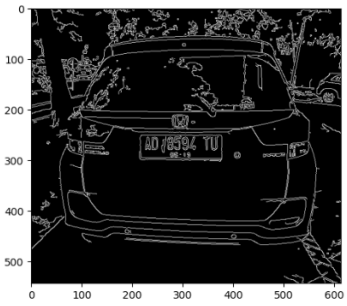

Figure 4. Edge detection result with Canny method

4. Find Keypoints and Contour Images
   In order to find the curve joining all of the continuous point in the image which has same colour or intensity we use a method called findContours in Opencv library. After the contour of the image is found, the location of the car plate inside the image can be found.

C. Masking and Cropping Image
   After the pre-processing steps, we continue the process by doing masking and cropping on the image, so that the method can be more focused on the car plate number that will be detected. After doing the masking, the OCR methods will be used.

D. OCR method
   1. Easy OCR
      Easy-OCR[7] is a method for detecting text using Character-

Region Awareness for Text detection (CRAFT) algorithm. Easy-OCR uses the model of Convolutional Recurrent Neural Network (CRNN). Which includes three components: feature extraction (Resnet) and Visual Geometry Groups (VGG), sequence labelling (LSTM) and decoding (Connectionist Temporal Classification (CTC)). Easy OCR method is the easiest compared to the OCR, Tesseract OCR, and Keras OCR methods, also for some aspects the Easy-OCR method can provide more accurate results than the other two methods.

2. Keras OCR
   Keras OCR[8] uses a hardware combination such as an optical scanner and software capable of image processing. Where Keras OCR will identify and digitize the numeric and alphabetic characters present in the image

3. Tesseract OCR
   Tesseract OCR[9] is an OCR machine which was developed in HP (Hewlett-Packard) between 1984 and 1994.[18] Which in some cases handling a high-resolution images can work very well if compared to the other OCR methods. But the Tesseract OCR method is very dependent to the preprocessing phase which carried out [19]. In the Tesseract OCR Method, there are many variables of image that can affect the result such as font and attribute of the text like underline/bold/italic, the distance between word in the text, and other variables.

IV. RESULT AND DISCUSSION

In this Research, we have few methods for the machine can read and recognize a car plate number well. The methods we used in this research are Easy OCR, Tesseract OCR, and Keras OCR. In this project, these three OCR methods used in this project came up with different result using the same image.

Comparison of the model with Tesseract, Keras, and Easy OCR can be seen in how to read license plates. Where in Keras OCR itself it will read the surroundings and display the text read by Keras, in Tesseract OCR it will be read one by one and will display the number plate and print the text, and the last is Easy OCR where in the reading process it is lighter than Keras and tesseract and reads faster and has good accuracy.

To demonstrate the use of car plate recognition, we use Streamlit. However, we encountered a problem with one of our models where we were unable to load Keras OCR on Streamlit. This is because Keras OCR is working too optimally so that it exceeds the RAM and processor capacity limits of each member of our group.

The results of the three OCR experiments that we implemented can be seen in Figure 5. It shows that the Keras OCR we are running can only read the license plate number of the vehicle, while in Figure 6 Easy OCR can read the entire license plate properly, but not with Tesseract OCR for the last letter on the plate that cannot be read.

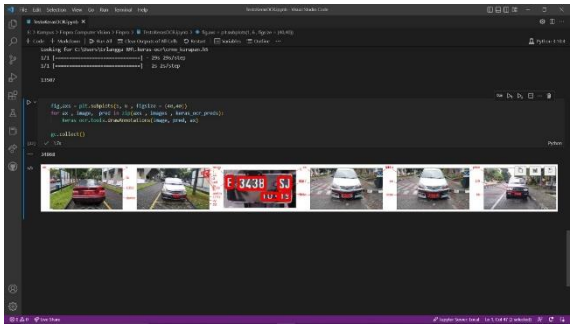

Figure 5. KerasOCR Recognition Result

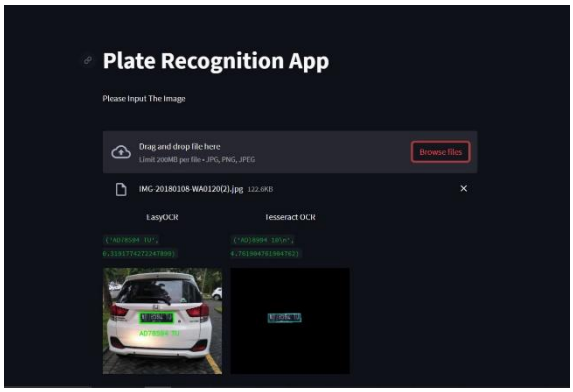

Figure 6. EasyOCR and Tesseract OCR recognition result

V. CONCLUSSION

From the experimental results by comparing using Easy OCR, Tesseract OCR, and Keras OCR to detect car plates, it can be concluded that each method shows different results and accuracy using the same image. All of the methods show can read the car plate number. These three methods can read license plates properly but on Tesseract and Keras OCR they cannot fully read the plate dataset. Our suggestion is that in the future maybe we can improve the method from Tesseract OCR or Keras OCR so that these two models can actually find and recognize car number plates according to the plates uploaded.

The Keras OCR method has an accuracy of 0.8 while the Easy OCR method has an accuracy of 0.3 and tesseract OCR method of 0.04. Using the results taken from this research, we can see that the highest method is held by Keras OCR.

References

[1] A. Songa, R. Bolineni, H. Reddy, S. Korrapolu, and V. J. Geddada, "Vehicle Number Plate Recognition System Using TESSERACT-OCR," *Int. J. Res. Appl. Sci. Eng. Technol.*, vol. 10, no. 4, pp. 323–327, 2022, doi: 10.22214/ijraset.2022.41198.

[2] B. P. Statistik, "Perkembangan Jumlah Kendaraan Bermotor Menurut Jenis (Unit), 2018-2020." https://www.bps.go.id/indicator/17/57/1/perkembangan-jumlah-kendaraan-bermotor-menurut-jenis.html (accessed Jan. 05, 2023).

[3] R. Ramachandiran, R. Ramapraba, and K. Suresh Joseph, "Multi depot vehicle routing problem using ODV EV based genetic algorithm," *2019 IEEE Int. Conf. Syst. Comput. Autom. Networking, ICSCAN 2019*, pp. 1–6, 2019, doi: 10.1109/ICSCAN.2019.8878869.

[4] D. A. Wicaksono, "PENEGAKAN HUKUM LALU LINTAS JALAN SECARA ELEKTRONIK SEBAGAI WUJUD PEMBANGUNAN HUKUM DALAM ERA DIGITAL," *J. Rechts Vinding*, vol. 9, no. 2, pp. 311–329, 2020.

[5] D. Islam, T. Mahmud, and T. Chowdhury, "An efficient automated vehicle license plate recognition system under image processing," *Indones. J. Electr. Eng. Comput. Sci.*, vol. 29, no. 2, pp. 1055–1062, 2023, doi: 10.11591/ijeecs.v29.i2.pp1055-1062.

[6] R. V. Singh and N. Randhawa, "Automobile Number Plate Recognition And Extraction Using Optical Character Recognition," *Int. J. Sci. Technol. Res.*, vol. 3, no. 10,

pp. 37–39, 2014, [Online]. Available: www.ijstr.org

[7] "EasyOCR." https://github.com/JaidedAI/EasyOCR. (accessed Jan. 11, 2023).

[8] "KerasOCR." https://github.com/faustomorales/keras-ocr (accessed Jan. 12, 2023).

[9] "TesseractOCR." https://tesseract-ocr.github.io/ (accessed Jan. 11, 2023).

[10] H. Idrose, N. A. B, H. A. K. B, and R. Shahid, *An Evaluation of Various Pre-trained Optical Character Recognition Models for Complex Lisence Plate*, vol. 1. Atlantis Press International BV. doi: 10.2991/978-94-6463-082-4.

[11] A. Herusutopo, R. Zuhrudin, W. Wijaya, and Y. Musiko, "RECOGNITION DESIGN OF LICENSE PLATE AND CAR TYPE USING TESSERACT OCR AND EmguCV," *CommIT (Communication Inf. Technol. J.*, vol. 6, no. 2, p. 76, 2012, doi: 10.21512/commit.v6i2.573.

[12] R. Sushma, M. Rithika Devi, N. Maheshwaram, and D. R. Sreedhar Bhukya, "Automatic License Plate Recognition with YOLOv5 and Easy-OCR method," vol. 9, no. 1, pp. 1243–1247, 2022.

[13] W. Khallouli, R. Pamie-George, S. Kovacic, A. Sousa-Poza, M. Canan, and J. Li, "Leveraging Transfer Learning and GAN Models for OCR from Engineering Documents," *2022 IEEE World AI IoT Congr. AIIoT 2022*, no. June, pp. 15–21, 2022, doi: 10.1109/AIIoT54504.2022.9817319.

[14] V. Agarwal, "Research on Data Preprocessing and Categorization Technique for Smartphone Review Analysis," *Int. J. Comput. Appl.*, vol. 131, no. 4, pp. 30–36, 2015, doi: 10.5120/ijca2015907309.

[15] X. Zhang, F. Xu, and Y. Su, "Research on the license plate recognition based on MATLAB," *Procedia Eng.*, vol. 15, pp. 1330–1334, 2011, doi: 10.1016/j.proeng.2011.08.246.

[16] W. A. Ulbeh, A. Moustafa, and Z. A. Alqadi, "Gray image reconstruction," *Eur. J. Sci. Res.*, vol. 27, no. 2, pp. 167–173, 2009.

[17] "Canny Edge Detection." https://docs.opencv.org/4.x/da/d22/tutorial_py_canny.html (accessed Jan. 12, 2023).

[18] A. Aprilino, I. Husni, and A. Amin, "Sistem Deteksi Plat Nomor Otomatis," vol. 16, no. 1, pp. 54–59, 2022.

[19] Y. Andreas, K. Gunadi, and A. N. Purbowo, "Implementasi Tesseract OCR untuk Pembuatan Aplikasi Pengenalan Nota pada Android," *J. Infra*, vol. 8, no. 1, pp. 2–7, 2020.