RESEARCH ARTICLE OPEN ACCESS

An Automatic Number Plate Recognition System for Sri Lankan Vehicle Number Plates

M. R. P.Nalaka Bandara[*, **], Xu Cheng[*]

[*]School of Computer & Software, Nanjing University of Information Science & Technology, 210044, Nanjing, China

[**]Criminal Investigation Department, Sri Lanka Police, Colombo 01, Sri Lanka

Email:nalakabandara.trs@gmail.com (MRPNB), xcheng@nuist.edu.cn (XC)

--************************----------------------------------

Abstract:

This study aimed at designing an automatic number plate recognition system (ANPR) using OpenCV, Python, and Tesseract to extract the vehicle registration number in Sri Lankan vehicle number plates for the possible after use in vehicle registration compliance checking by the Sri Lanka Police (SLP). The key steps followed in the ANPR system were the image capture, image preprocessing by resizing, gray scaling, and blurring, number plate localization by edge detection and contouring, character segmentation and finally character recognition. At the end, the accuracy of the system was tested using 10 sample images of Sri Lankan vehicles, which showed an accuracy percentage of 90% in number plate detection.

Keywords **—ANPR, Computer Vision, Image Processing, OpenCV, Python**

--************************----------------------------------

I. INTRODUCTION

Automatic number plate recognition (ANPR) is an image processing technology for capturing images of vehicles to extract information about their number plates (license plates). The image of the vehicle number plate is translated into a machine-readable format, such as text string, which can then be processed and indexed into a database to find information about the vehicle for meaningful applications by governments or companies [1].

In Sri Lanka for any vehicle to travel on the road, the vehicle should be legally imported and must be cleared from Sri Lanka customs. At any time, the driver should be able to provide up-to-date reports issued by the institutions including the department of motor traffic (DMT), an authorized insurance company, vehicle emission testing agency, and provincial revenue license department. Currently Sri Lanka Police (SLP) department's traffic branches all over the country conducts on-road inspections of vehicles by randomly stopping vehicles to check these reports manually which has created issues related to practical difficulties to check all vehicles, requirement of a large number of officers, possible corruption, and induction of traffic jams. Therefore, an ANPR system for Sri Lanka is timely needed.

In this regard, this study aims at designing an ANPR system to extract the Sri Lankan vehicle number plate that can be used in the future to check vehicle registration compliances by cross checking with the databases of the above-mentioned key institutions.

A typical ANPR system is comprised of several steps such as image capture, image pre-processing, number plate localization, character segmentation, and character recognition [2, 3].In image capture, an image of the vehicle is captured either using a conventional analog camera and a scanner, digital camera, or using a video camera or CCTV and a frame grabber to select a frame [4].The next step is image pre-processing in which several techniques such as resizing, gray scaling, and blurring are employed to prepare the captured image for the ANPR process [5]. These techniques reduce the background noise and enhance the contrastof the image. Then the image is used for number plate localization during which the location of the number plate is identified, and the unwanted regions of the image are removed [3]. Techniques such as edge detection and contouring help in this step. For example, edge detection helps to detect the edges in the image in which the number plate is also a part of [2].The contouring step is useful in shape analysis and object detection and recognition. A contour can be simply defined as a link of equal intensity points along a boundary [6]. From this step rectangular contours can be separated out as the number plates are rectangular in shape [2]. In the character segmentation step, individual characters (letters) in the number plate are separated and extracted [2, 3]. Finally, the characters in the number plate are recognized in the character recognition step using optical character recognition (OCR) algorithms.

Among several software, the ANPR systems can be easily and effectively developed using OpenCV, Python, and Tesseract.OpenCV is an open-source computer vision and machine learning library which is used for real-time computer vision. This library is used to detect and recognize images and objects, track moving objects, blend the images, and to detect motion [7]. OpenCV provides options to deal with several image processing techniques useful in ANPR such as image resizing, color conversion, smoothening and blurring, edge detection, contouring, etc. [7, 8].Python is the most employed language in image processing owing to its simple syntax and the availability of a number of libraries and modules to work with image processing such as the python image library (PIL), Matplotlib, and NumPy [8, 9].The PIL provides useful image handling options such as resizing, cropping, rotating, color conversion, etc. The Matplotlib module is an important graphics library when working with mathematics and plotting graphs, lines, and curves on images. The NumPy package contains a number of useful concepts such as array objects (for representing vectors, matrices, images and much more) and linear algebra functions useful in image processing [9]. Tesseract is a widely used open-source OCR engine which helps extracting the text from images [8].

The aim of this study was to design an ANPR system using OpenCV, Python, and Tesseract to extract the Sri Lankan vehicle number plates for the possible after use in vehicle registration compliance checking by the SLP.

II. METHODOLOGY

A. Number Plate Extraction

1. Image Capture

In this project, high resolution images were captured using a digital camera. A sample image to describe the further steps is shown in Fig. 1.

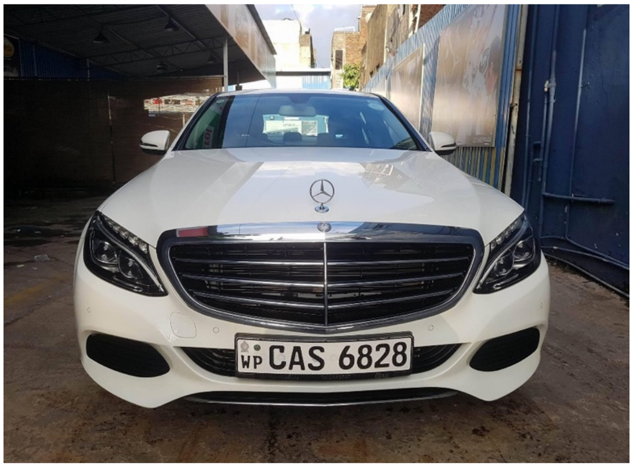


Fig. 1 Sample image

The image was first loaded to OpenCV (Fig. 2) using the following code.

```
testImage = cv2.imread('D:/sample
image/9.jpg',cv2.IMREAD_COLOR)
cv2.imshow("testImage",img)
cv2.waitKey(0)
cv2.destroyAllWindows()
```

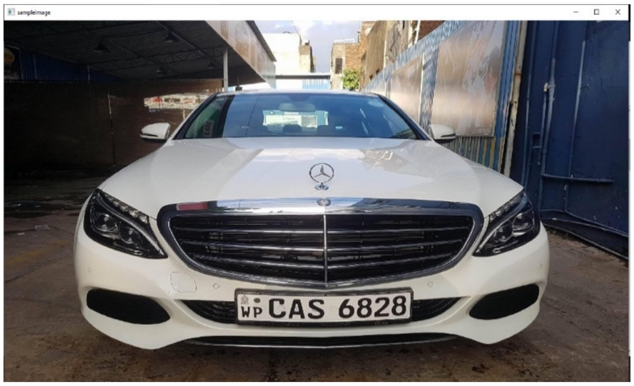


Fig. 2The input image

2. Image Pre-processing

Then as the first step in image pre-processing, the image was resized using the following code. The resized image is shown in Fig. 3.

```
testImg= cv2.resize(img, (600,400) )
cv2.imshow("resizedImage", testImg)
cv2.waitKey(0)
cv2.destroyAllWindows()
```

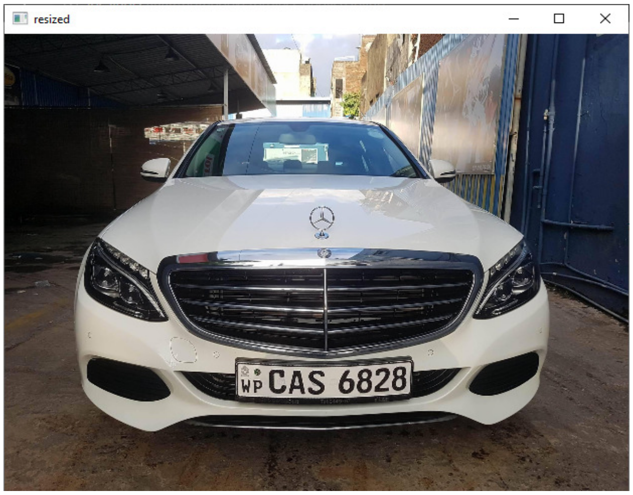


Fig. 3The resized image

Secondly, the image was grayscaled using the following code. The gray scaled image is shown in Fig. 4.

```
grayImg = cv2.cvtColor(img,
cv2.COLOR_BGR2GRAY)
cv2.imshow("grayImage",gray)
cv2.waitKey(0)
cv2.destroyAllWindows()
```

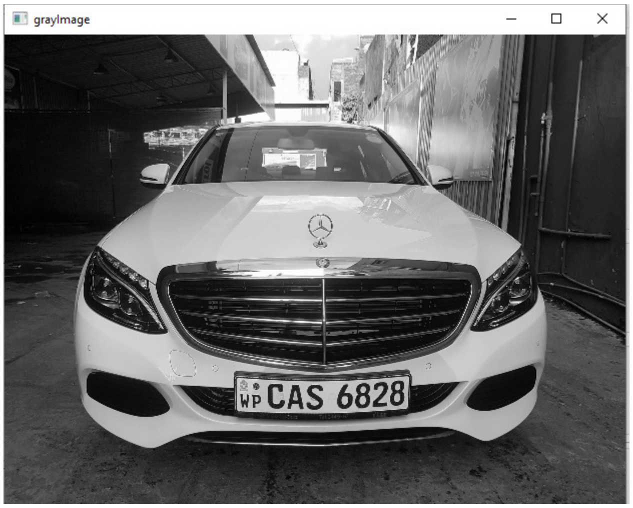


Fig. 4The gray scaled image

Then the image was further pre-processed by applying a bilateral filter (blurring). By blurring the image, its unimportant details are removed. However, we should be careful to avoid blurring the useful information. So, the sigma color and sigma space were given a value of 15.

```
grayImg = cv2.bilateralFilter(gray, 13, 15,
15)
cv2.imshow("blurImage",grayImg)
cv2.waitKey(0)
cv2.destroyAllWindows()
```

The blurred image is shown in Fig. 5. As clearly seen in fig. 5, the background details (e.g., the building) has been blurred after this step.

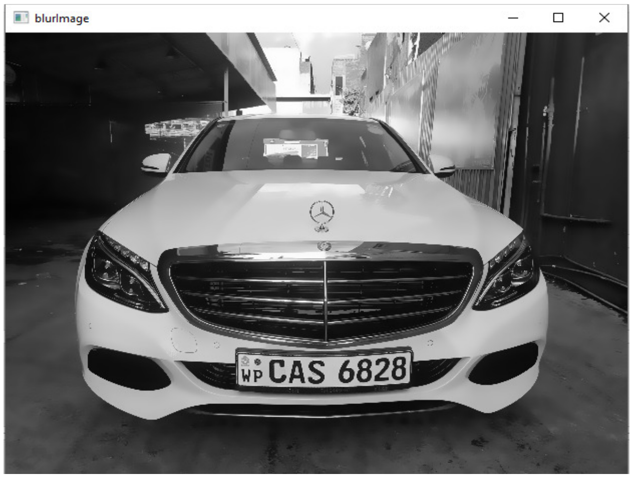


Fig. 5The blurred image

3. Number Plate Localization

As the initial step in localizing the number plate, an edge detection step was carried out using OpenCV's smart edge technique (Canny edge detection) using the following code.

```
edgedImg = cv2.Canny(gray, 30, 200)
cv2.imshow("edgedImage",edgedImg)
cv2.waitKey(0)
cv2.destroyAllWindows()
```

The values indicated as 30 and 200 in the above syntax are the minimum and maximum threshold values. By setting those values, the only edges that is shown in the output (Fig. 6) are those whose intensity gradient is greater than the lowest threshold value but less than the maximum threshold value.

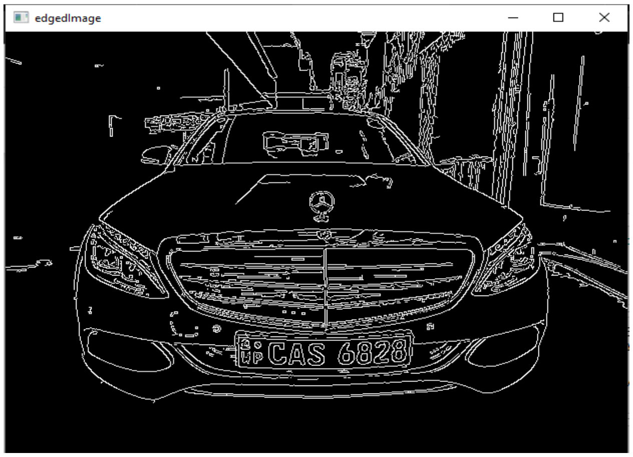


Fig. 6The image after edge detection

As the next step in number plate localization, contours were identified on the image.The use of a value of -1 as the 3rd parameter in the following syntax enable us to draw all the contours in the image.

```
cv2.drawContours(image,contours,-1,(0,255,0),3)
cv2.imshow('Contours',image)
cv2.waitKey(0)
cv2.destroyAllWindows()
```

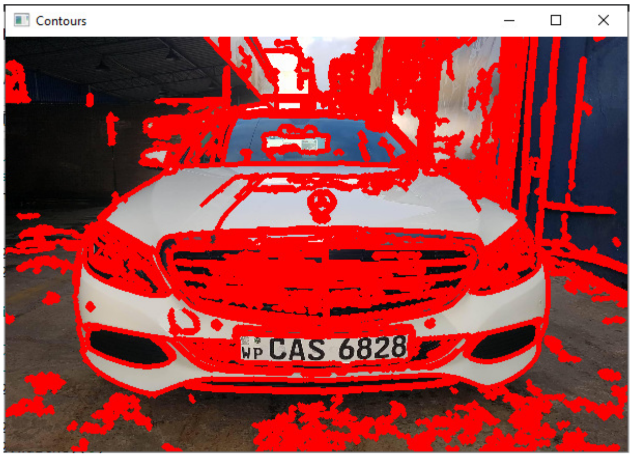


Fig. 7Contours detected on the image

The contours were sorted from large to smaller and the first 10 contours were selected as the candidate contours. The image with the identified contours is shown in Fig. 7.

From the top 10 candidate contours, the rectangle contour with four sides and a closed figure was identified as the number plate. In order to confirm the number plate is correctly identified, a rectangular box was created around the correct contour and the image was masked only to show the number plate (Fig. 8).

```
mask = np.zeros(gray.shape,np.uint8)
new_image =
cv2.drawContours(mask,[screenCnt],0,255,-
1,)
new_image =
cv2.bitwise_and(img,img,mask=mask)
```

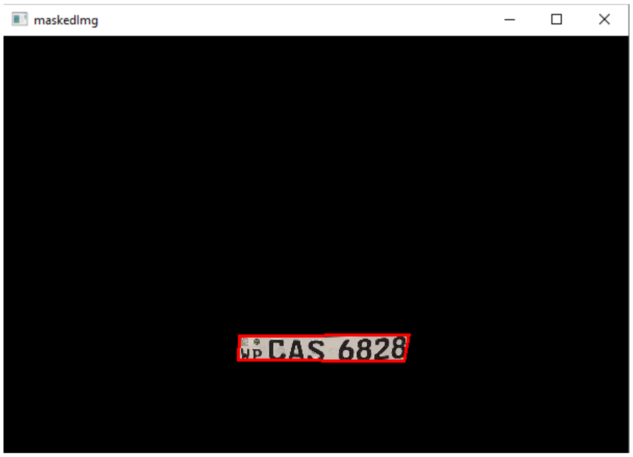


Fig. 8The masked image with the number plate

4. Character Segmentation

Then the number plate was cropped and stored as a different image (Fig. 9) before identifying the characters on it.

```
(x, y) = np.where(mask == 255)
(topx, topy) = (np.min(x), np.min(y))
(bottomx, bottomy) = (np.max(x), np.max(y))
Cropped = gray[topx:bottomx+1,
topy:bottomy+1]
```

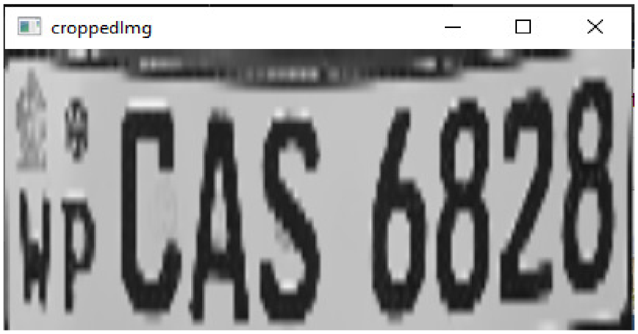


Fig. 9The cropped image of the number plate

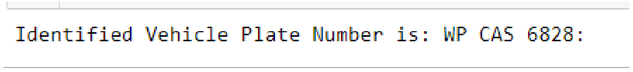
5. Character Recognition

Finally, the characters were recognized from the cropped image of the number plate using the Py-Tesseract package using the following code.

```
toText =
pytesseract.image_to_string(Cropped,
config='--psm 11')
```

```
print("Detected license plate Number is:",text)
```

The output result is shown in Fig. 10.



```
Identified Vehicle Plate Number is: WP CAS 6828:
```

Fig. 10The identified vehicle number

B. Accuracy Testing

Then the accuracy of the system was tested using 10 sample vehicle images.

III. RESULTS

The results showed that the developed method can successfully detect Sri Lankan vehicle number plates and extract the number plate in text format.

Table 1 summarizes the results of the accuracy test where we obtained a 90% accuracy in the number plate detection. Out of the 10 images of Sri Lankan vehicles only one image (No. 03) was not recognized due to an unknown reason. Apart from that, in another image (No. 04) which belonged to a vehicle that has a number plate that followed the previous guidelines of the DMT, an additional letter "P" was detected which indicated the fuel type (petrol).

TABLE I
RESULTS OF THE ACCURACY TEST

No.	Vehicle number according to the registration	Detected number by the ANPR of the study
01	WP QZ-1815	WP QZ-1815
02	WP CAA 1454	WP CAA 1454
03	WP CBB 5921	Not Detected
04	NW KL-6036	NW KL-P6036
05	WP CAC 7305	WP CAC 7305
06	NW CBJ 3263	NW CBJ 3263
07	NC CAY 7358	NC CAY 7358
08	WP CAI 8335	WP CAI 8335
09	WP CAS 6828	WP CAS 6828
10	WP CAP 9686	WP CAP 9686

IV. DISCUSSION

Traffic control and identification of vehicle ownership has become a major problem all over the world in recent days because of the growing number of vehicles in every country. In that regard, the development of ANPR systems followed by compliance checking with necessary databases is an effective option to address this issue. In this project an ANPR system was developed using OpenCV, Python, and Tesseract to extract Sri Lankan vehicle number plates with the possibility of after use in compliance checking with relevant databases.

Our results showed a high accuracy in number plate detection. However, in Sri Lanka the vehicle number plates do not follow a uniform standard, and this may cause problems when the output is used to cross check with databases for compliance checking. For example, during a particular time duration in Sri Lanka, the DMT has decided to add the fuel type (petrol vehicle as “P” and diesel vehicles as “D”) of the vehicle into the vehicle’s number plate. As a result of that when such a number is detected by this ANPR system, that character which is used to denote fuel type also detected. To avoid such confusions, the output can be further processed using algorithms to store the number plate in the correct formatin accordance with the registration databases in the future.

V. CONCLUSIONS & RECOMMENDATIONS

The present study designed an ANPR system using OpenCV, Python, and Tesseract to detect Sri Lankan vehicle number plates. The key steps followed in this system were image capture, image pre-processing, number plate localization, character segmentation and character recognition. The system showed a 90% accuracy in the detection of number plates. As the next step, it is recommended to develop a system that checks vehicle registration compliance by cross checking the extracted vehicle numbers with the relevant authorities.

REFERENCES

[1] Tang, J., Wan, L., Schooling, J., Zhao, P., Chen, J., & Wei, S. (2022). Automatic number plate recognition (ANPR) in smart cities: A systematic review on technological advancements and application cases. Cities 129: 103833.

[2] Kumari, S., Gupta, L., & Gupta, P. (2017). Automatic license plate recognition using OpenCV and neural network. International Journal of Computer Science Trends and Technology (IJCST) 5(3): 114–118.

[3] Khinchi, M., & Agarwal, C. (2019). A review on automatic number plate recognition technology and methods. In International Conference on Intelligent Sustainable Systems (ICISS 2019): 363–366. IEEE.

[4] Sarfraz, M., Ahmed, M. J., & Ghazi, S. A. (2003). Saudi Arabian license plate recognition system. In Proceedings of the 2003 International Conference on Geometric Modeling and Graphics (GMAG’03): 36–41. IEEE Computer Society.

[5] Agbemenu, A. S., Yankey, J., & Addo, E. O. (2018). An automatic number plate recognition system using opencv and tesseract ocr engine. International Journal of Computer Applications, 180(43): 1–5.

[6] Varma P, R.K., Ganta, S., Krishna B, H., &Svsrk, P. (2020). A novel method for Indian vehicle registration number plate detection and recognition using image processing techniques. Procedia Computer Science 167: 2623-2633.

[7] Sharma, V., Jain, M., Jain, T., & Mishra, R. (2022). License plate detection and recognition using OpenCV–Python. In Recent Innovations in Computing: Proceedings of ICRIC 2021(1): 251–261. Springer Singapore.

[8] Sharif, A.S.M., Bhatia, R., Kuma, R., & Jha, S. (2021). Vehicle number plate detection using python and opencv. In 2021 International Conference on Advance Computing and Innovative Technologies in Engineering (ICACITE): 525–529. IEEE

[9] Solem, J. E. (2012). Programming Computer Vision with Python: Tools and algorithms for analyzing images. O'Reilly Media Inc., USA

[10] Sanjana, S., Sanjana, S., Shriya, V. R., Vaishnavi, G., & Ashwini, K. (2021). A review on various methodologies used for vehicle classification, helmet detection and number plate recognition. Evolutionary Intelligence 14(2): 979–987.