International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 05 Issue: 06 | June - 2021 ISSN: 2582-3930

Automated Detection and Recognition of License Plate

Varalatchoumy M[1], Kushal B S[2], M Vignesh Sundar[3], Yuvaraj V[4], Arunima Barman[5]

[1]Associate Professor, Dept of CSE, Cambridge Institute of Technology
[2,3,4,5] UG Students, Dept of CSE, Cambridge Institute of Technology

---

Abstract - For a long time, two-, three-, and four-wheeler negligence has been a threat to society. Various methods have been proposed and implemented to detect and extract the license plate number; however, due to poor video quality and non-uniform illumination, license plate detection becomes a very difficult task. Recent advances in the field of Computer Science like Machine learning, Image Processing, Edge detection and OCR provide an innovative solution to this conundrum. The input to this model is the image which is obtained from camera which is strategically placed. The output is license plate number obtained in the text format, which is further used in swift identification of the registered vehicle's owner. This project License plate detection and recognition module aims to aid the general public in deciphering the vital details of the registered vehicle owner and ultimately help apprehend the wrong doers.

Key Words: License plate detection, Character Segmentation, Optical character recognition.

1. INTRODUCTION

Travel by road is one of the cheapest and most convenient forms of transportation. Apart from the regular taxies and auto rickshaws, majority of the vehicle owners be it 2-wheeler, 3-wheeler or 4- wheeler make up majority of the road population and due to their negligence have always caused a nuisance and have forever been a pain to the society. Their disregard and negligence towards the protocols governing the roads have led to so many fatalities till date. Though it is known that the majority of accidents are caused on the road by these vehicle owners, still nothing has been done about it.

According to the Indian National Status Report on Road Safety 2018, road safety remains a major problem, a public health concern, and a leading cause of death and injury around the world, killing more than 1.35 million people, with ninety percent of these fatalities occurring in developing countries and India accounting for eleven percent of all fatalities [1].

According to the 2019 Report on Road Accidents in India, more than 1.5 million people died in accidents in India in 2019. Ultimately, no significant headway on this front, despite the Indian government's continuing efforts to reduce mortality rates

. The Ministry of Road Transport and Highways has undertaken a number of projects, including those connected to vehicular and road engineering, as well as academic initiatives aimed at raising road safety awareness. I believe that the complete implementation of the MVA 2019 legislation throughout the country will mark a turning point in the country's road-related legislation [1].

This project is a Graphical User Interface (GUI) desktop application which is developed using Python Application Programming language. this project is mainly divided into four different modules. The first module will take an image as input and convert that image into a NumPy array i.e., converting a digital image into number representation in an array. Then this NumPy array is fed into second module which is converting the color image to greyscale image because processing a greyscale image is easy and faster compared to color image.

This greyscale image is now fed into the third module i.e., Edge Detection module where it will find the required region of interest. In this project the region of interest is the license plate region, after identifying this region it is fed into the last module i.e., Optical Character Recognition (OCR) module which converts this region of interest to text format. Further this text can be used as a key to search for the vehicle owner details.

2. RELATED WORK

Researchers from all over the world have previously devised a number of ways for identifying license plate numbers and extracting the characters. These methods, however, do not perform well on Indian number plates, owing to the wide range of letter shapes and sizes. Hence, the main aim of this paper is to briefly introduce the previous approaches to detect Indian license plates and then to present a deep learning-based approach for the same.

Previously, classic image processing techniques were used to detect license plates. In [2], To identify the license plate region, the authors apply histogram equalization to enhance the image, followed by morphological opening and closing with edge detection using the Sobel operator. Another example of a similar approach can be seen when the authors in [3] Following a virtually identical set of processes as shown formerly for detecting license plates, character segmentation is performed. The photos are preprocessed using a median and wiener filter in the segmentation section. Finally, each character is divided separately using the connected components methodology to identify it.

Convolutional neural networks (CNNs) have been prominent in tackling object detection issues as a result of recent breakthroughs in deep learning models. As a result, many researchers have suggested license plate detection methods based on CNN. In [4], to extract the characters from license plates, the authors presented a deep learning-based technique.

The image is preprocessed appropriately for robust plate localization in the early stages. After that, the license plate is located by guessing the x and y coordinates of the plate's limits. The picture of the plate is then recorded as the python class object of each character using these coordinates. Following that, the characters from the plate were split, and any undesirable symbols or characters were eliminated.

3. SYSTEM ARCHITECTURE

The Fig-1 depicts the system architecture of the proposed system. As input, a picture of a vehicle with a license plate that has to be recognized and identified can be converted into a NumPy array that may be used for further processing. The NumPy array is then converted to a greyscale image in the OCR module for faster character detection.

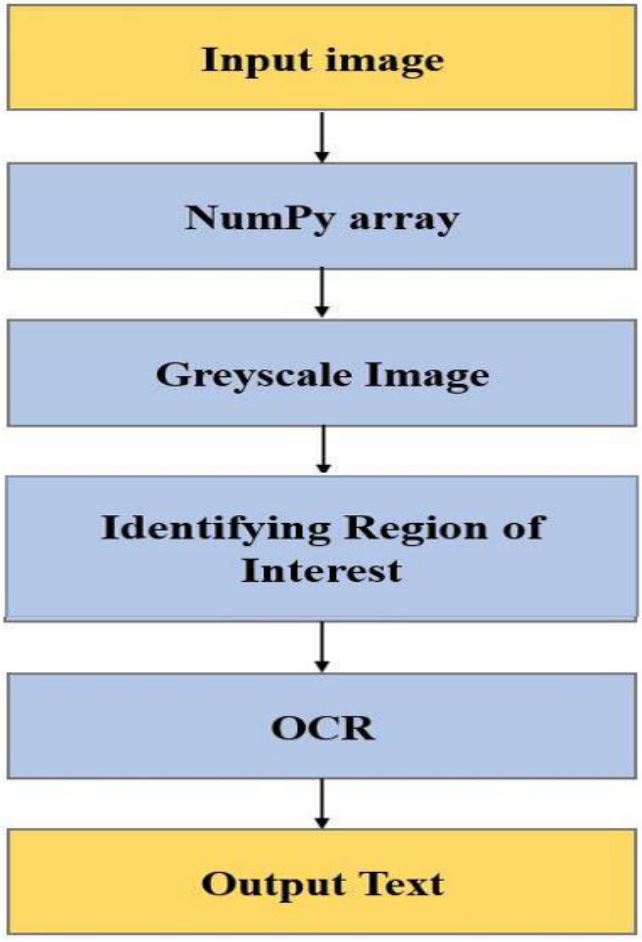

Fig-1: System Architecture

After converting the image to greyscale, there need to start identifying the region of interest, which is the license plate region. Techniques like the Sobel operator, Canny algorithm, and Prewitt operator, among others, can assist with this. The region of interest will be utilized as input and passed to the OCR module once it has been discovered using the techniques outlined above.

The OCR module extracts the needed text from the region of interest and outputs it after obtaining the region of interest. This text can be used to identify and recognize the owner of that specific vehicle.

4. METHODOLOGIES

As illustrated in Fig-2 and Fig-3, the input picture is grayscale transformed once it is acquired, and several approaches are available to help with this process, one of which is the use of a NumPy array. Because there are other direct ways in Python to convert an image into grayscale image, such as cvtColor(), one may bypass the step of turning the input image into a NumPy array.

```
img = cv2.imread('./images/2.jpg')
display_cv(img)
```

Fig-2: Input Image

```
img = cv2.imread('./images/2.1.jpg')
grayimg = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)
display_cv(grayimg)
```

Fig-3: Input Image subjected to Grayscale Transformation

Post Grayscale transformation, various edge detection techniques are applied on the gray scaled image in order to obtain the desired region of interest. The region of interest in this project is only the license plate and not the entire vehicle.

Fig-4 and Fig-5 illustrates the concepts of bounding box and object detection to obtain the region of interest. In Python for object detection, the detectMultiScale() method is

used. This method belongs to cv2 package and comes under CascadeClassifier class.

The parameters passed to this method input image, scale factor and minNeighbors. Scale factor is defined as the Parameter specifying how much the image size is reduced at each image scale, the scale factor value used in this project is 1.3. minNeighbors is defined as Parameter specifying how many neighbors each candidate rectangle should have to retain it. The minNeighbors value used in this project is 7. After providing these parameters, this method returns the co-ordinates of the required region of interest.

In conclusion, the detectMultiScale () method returns

```
display_cv(output_img)
```

Fig-4: The Detected Bounding Box

```
cropped = img[y:y+h, x:x+w, :]
display_cv(cropped)
```

Fig-5: The Estimated Region of Interest.

The image consisting of the region of interest is fed to the OCR (Optical Character Recognition) module. The OCR module is used to convert the image that is imputed to it into text. The various OCR modules like Google Drive OCR or Google Cloud Vision, Tesseract etc. that exists, in this project we have opted to use the inbuilt OCR module available in Python i.e., EasyOCR, it is a Python package that allows computer vision developers to effortlessly perform Optical Character Recognition.

5. EXPERIMENTAL RESULTS

Fig-6 represents the text obtained from the OCR module can be further used to conduct a detailed search in the backend database, following which, if a match is obtained, the following vehicle owner's details like the name, address and email is retrieved.

```
C:\Users\KUSHAL BASAPATHI>python gui.py
Using CPU. Note: This module is much faster wi
C:\Users\KUSHAL BASAPATHI\AppData\Local\Progra
ng: Named tensors and all their associated API
m for anything important until they are releas
```

Fig-6: Predicted Text of the License Plate

```
predicted text is: KL55R2473
```

To check the robustness of the proposed approach, A test was conducted on the proposed model using 98 testing images and the corresponding quantitative metrics were calculated. Out of 98 test images, the model performs successfully on 82 number of images Based on this information, the quantitative metrics (i.e., accuracy and precision) have been calculated and the same is mentioned in Table-1.

Table -1: The quantitative measurements of the proposed approach obtained from the testing phase.

Parameter	Value
Accuracy	0.9125
Precession	0.9376

6. CONCLUSION

In this article, a Python-based model is proposed for identifying and recognizing license plate numbers from vehicles, which may be used in various law enforcement and surveillance systems. The proposed technique consists of three modules: detection, extraction, and identification of license plates.

In the suggested technique, making use Python libraries such as NumPy and pillow in the detection module, which was used to recognize license plates of vehicles registered in India. After that, the plates are sent via the suggested extraction module, which crops the identified license plate region from the entire image. Finally, only the license plate region is fed into an optical character recognition (OCR) module, where the text format of the license plate is obtained as output. Following that, the obtained text can be utilized to obtain pertinent information on the vehicle owner, who can then be prosecuted if he or she breaks any traffic rules.

REFERENCES

1. https://morth.nic.in/sites/default/files/RA_Uploading.pdf.
2. Kaur, Sarabjit, and Sukhvir Kaur, "An Efficient Method of Number Plate Extraction from Indian Vehicles Image", International Journal of Computer Applications 88, no. 4 (2014).
3. Sharma, Chetan & Kaur, Amandeep, "Indian vehicle licence plate extraction and segmentation", International Journal of Computer Science and Communication 2. no. 2(2011). 593-599.
4. Hashmi, Saquib & Kumar, Kaushtubh & Khandelwal, Siddhant & Lochan, Dravit & Mittal, Sangeeta, "Real Time License Plate Recognition from Video Streams using Deep Learning", International Journal of Information Retrieval Research. 9. 65-87. 2019, 10.4018/IJIRR.2019010105.
5. Vishal Jain, Zitha Sasindran, Anoop Rajagopal, Soma Biswas, Harish S Bharadwaj, and K R Ramakrishnan, "Deep automatic license plate recognition system", In Proceedings of the Tenth Indian Conference on Computer Vision, Graphics and Image Processing (ICVGIP '16). ACM, New York, NY, USA, Article 6, 8 pages, 2016. DOI: https://doi.org/10.1145/3009977.3010052
6. Tom M. Mitchell, Machine Learning, India Edition 2013, McGraw Hill Education.
7. Ethem Alpaydın, Introduction to machine learning, second edition, MIT press.
8. Rafael C G., Woods R E. and Eddins S L, Digital Image Processing, Prentice Hall, 3rd edition, 2008.
9. http://mospi.nic.in/sites/default/files/statisticalyearbookindia2015/Table-20.1 1.xlsx
10. Trevor Hastie, Robert Tibshirani, Jerome Friedman, h The Elements of Statistical Learning, 2nd edition, springer series in statistics.
11. Milan Sonka," Image Processing, analysis and Machine Vision", Thomson Press India Ltd, Fourth Edition.
12. Howard, Andrew G., Menglong Zhu, Bo Chen, Dmitry Kalenichenko, Weijun Wang, Tobias Weyand, Marco Andreetto and Hartwig Adam. "MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications", ArXiv abs/1704.04861 (2017).
13. Mayasari, N. & Siahaan, A.P.U., "Vehicle Plate Recognition using Template Matching, "International Journal for Innovative Research in Multidisciplinary Field, vol. 4, No. 10, pp. 259-263, Oct. 2018.
14. W. Zhou, H. Li, Y. Lu, and Q. Tian, "Principal visual Word discovery for automatic license plate detection," IEEE Trans. Image Process., vol. 21, no. 9, p. 4269-4279, 2012.