IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

A Review Paper on Automatic Number Plate Recognition.

[1]B J Praveena, [2]Nampalli Shravan Kumar, [3]Sudaveni Nishitha, [4]T Akhil Kumar

[1]Assistant Professor,[2]Student,[3]Student,[4]Student

[1,2,3,4]Department of Computer Science and Engineering,

[1,2,3,4]Matrusri Engineering College, Hyderabad, India.

Abstract:

In Our day-to-day life usage of vehicles are increasing rapidly. So the vehicle-related crimes, thefts, and violation of traffic rule increasing. So ANRP(Automatic Number Plate Recognition) is introduced which solves all the existing problems. As there are plenty of ANRP approaches as well as solutions, even there exist some unsolved problems like the characters on the number plate are unrecognizable, lighting also plays as a parameter and ANRP recognize only horizontal number plates but what about some angles number plates. In this paper, we approached some other techniques to overcome some existing problems, and the number recognized by the system will cross-compare with the database to retrieve the owner's information. This work presents the usage of CV and Keras to localize the plate and extraction of text from it.

Keywords :

Automatic Number Plate Recognition (ANPR), Keras OCR (optical character recognition) ,CV(computer vision), Artificial Neural Network (ANN),Processing, Localizing , Character Segmentation and text extraction.

I. Introduction

In present situation it is almost impossible to identify vehicles information manually, history of the vehicle, traffic challan status ect. So ANRP came into picture to overcome all this problems. Even those there are some areas which required manual interruption. Some are discussed and solved in this paper. This Automatic number plate detection system is extremely useful in many ways. Our system will take a image as input which contains only a car , gives the text of its number plate, and it compare the number with the database to retrieve the owners information , this is the main objective of the system. This helps cops to identify the stolen vehicles, If any traffic violation by the car cops can fine them just by taking a picture of the car, finding the authorised car to enter into a private place ect.

II. LITERATURE REVIEW

1. Number plate recognition in India

In this paper[1], Number plate recognition in India, a large quantity of problems come up, reason being, a difference in of font sizes, different colors and double line number plates etc. This leads to high level of inaccuracy in the final result. In this research all these problems on real Indian road conditions are taken care of. ANN is used for character recognition and SVM is to detect plate contour.Various algorithms to remove noise and enhance plate recognition and the usage of neural network for best results with easing lots of camera constraints.

2. Different conditions to capture

In this paper[2], The third person proposed system concerns unclear number plates with differences in weather and lighting environments, high speed vehicle and different traffic situations, which add to the difficulties of extracting relevant details of vehicle number plate. By addressing the various issues with hardware platform along with real time and ingenious algorithms. The dataset thus received includes images as of various paths from different road, street and highway, daytime and nighttime, inclement weather environment and different number plate clarities. The proposed application for various parts of the system are robustly receptive to variations in light, size, clarity of the number plates. Since our system is not language dependent, an industrious reliable ANPR for high speed application is proposed. The aforementioned techniques and algorithms along with our dataset help in compiling a dedicated set of solutions to problems and challenges involved in the formation of number plate recognition system in various intelligent transportation system applications.

III. Technology Stack

3.1 PYTHON: Python is an interpreted, object-oriented, high-level programming language with dynamic semantics. ... Python's simple, easy to learn syntax emphasizes readability and therefore reduces the cost of program maintenance. Python supports modules and packages, which encourages program modularity and code reuse.[3]

3.2 Colaboratory: Colaboratory, or “Colab” for short, is a product from Google Research. Colab allows anybody to write and execute arbitrary python code through the browser, and is especially well suited to machine learning, data analysis and education.[4]

3.3 OPEN-CV: OpenCV (Open Source Computer Vision Library) is an open source computer vision and machine learning software library. ... The library has more than 2500 optimized algorithms, which includes a comprehensive set of both classic and state-of-the-art computer vision and machine learning algorithms[5].

3.4 KERAS OCR: This is a slightly polished and packaged version of the Keras CRNN implementation and the published CRAFT text detection model. It provides a high level API for training a text detection and OCR pipeline.[6]

IV. Proposed System [9]

System Architecture:

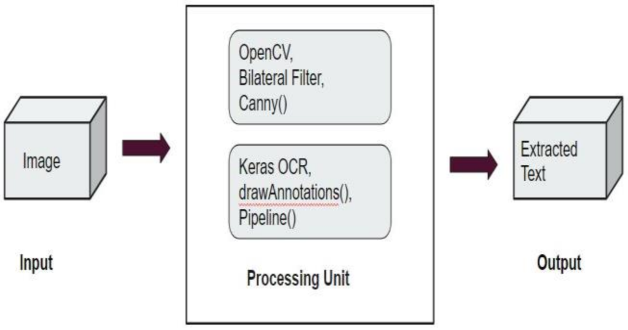


Fig.1 : skeleton Architecture

Pictorial Architecture:

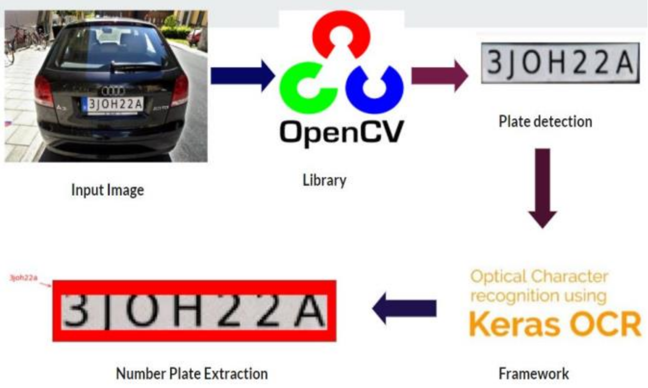


Fig.2 : Pictorial Architecture

1. INPUT:
   The administrator will pass a image of car to the system which is already processed from existing systems to extract car from a picture.
2. Processing Unit:
   The system takes input and process it with certain methods from openCv and OCR.
3. OUTPUT:
   From the unit it send the extracted text as output to the administrator.

V. Flow and Implementation:

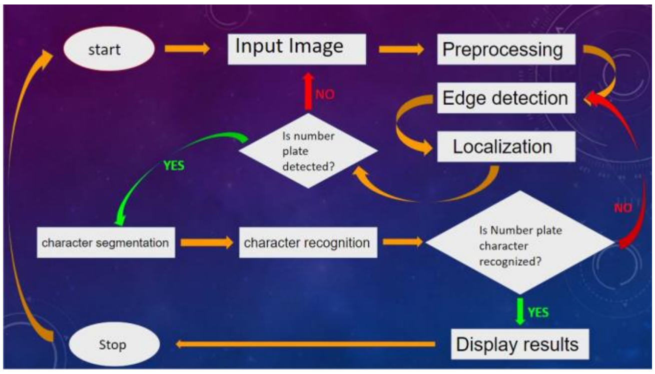


Fig:3 flow chart.

In the above flow we start with giving a image as input to Preprocessing, here we process the image from noice then it moves to Edge detection where it detects all the edges in the input by comparing the intensity of colour change. Then it localize the picture and extract the number plate. From there the number plate is given as input the character segmentation which extracts text and displays the same.

Image Acquisition

The initial step is to receiving input image. These Caught images are in RGB format so it can be further process for the Number Plate Extraction.

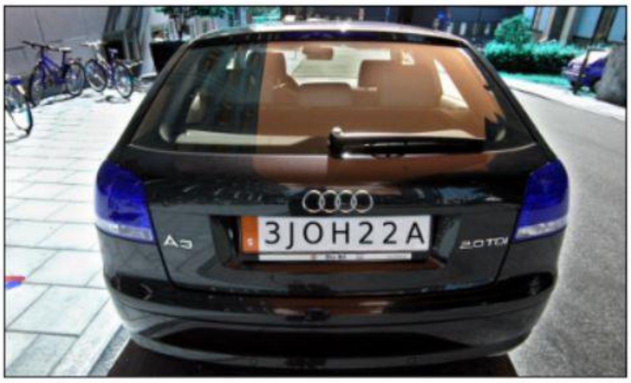


Fig 4: Input image

Processing the Image

before the main image processing, pre-processing of the captured image ought to be taken out which include converting RGB to gray , clamor evacuation, and border enhancement for brightness.

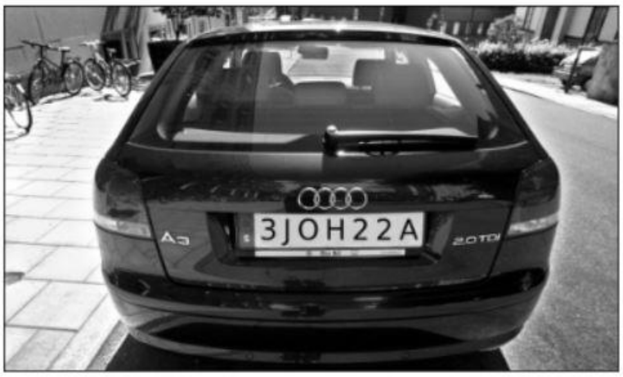


Fig 5: Gray image

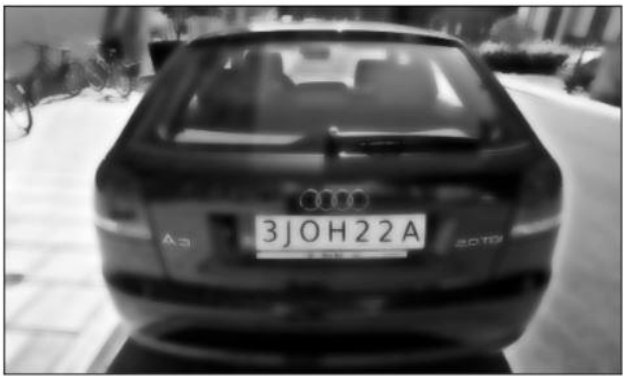


Fig 6: Blurred gray image

Edge detection:[7]

Here we pass the above fig as input to the Canny method which is available in openCv. It detects the all the edges as shown below.

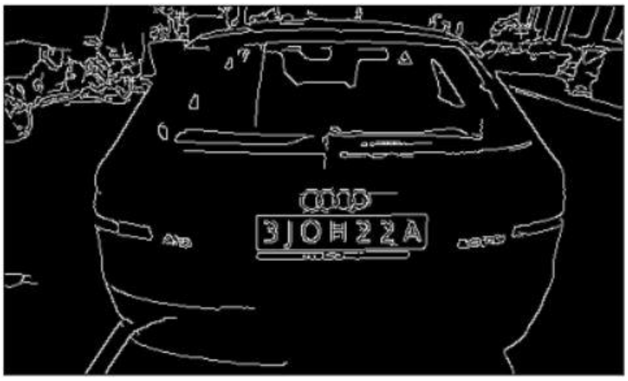
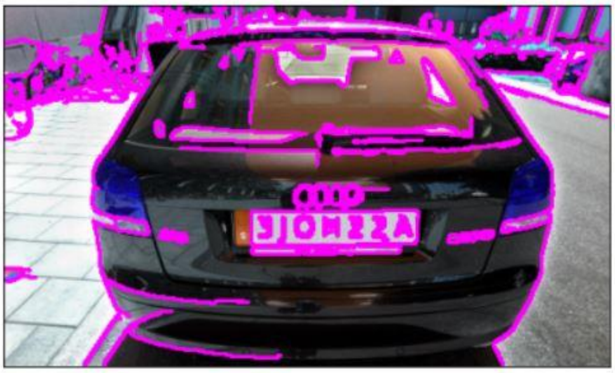


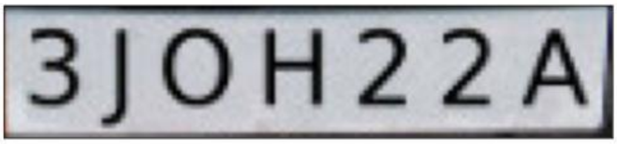
Fig 7: Edge detection

Extraction of Number plate:

Next step is draw contours to detect rectangular edges. We pass the above fig as input to this, so it produce the output as shown below.

Fig 8: Detecting plate

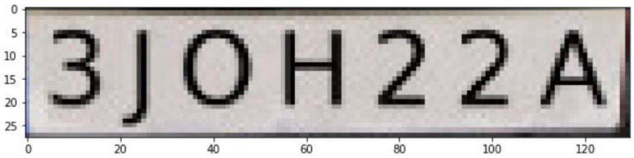




Extraction of Characters :

So, finally we got required number plate , now extraction as text as follows,

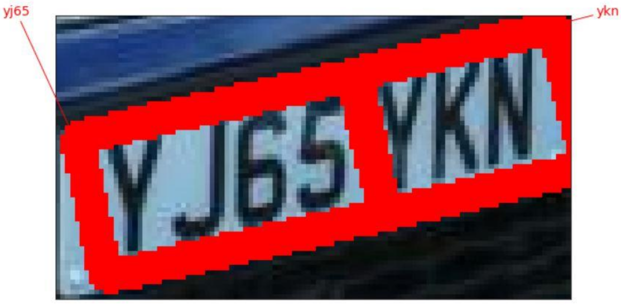
First we pass the above fig as input to the Keras ocr pipeline[8]. It will recognise the text are by blurring the rest.



Next, it extracts the text ,

Finally we got the required text and we can convert it into uppercase , so the final output will be **"3JOH22A".**

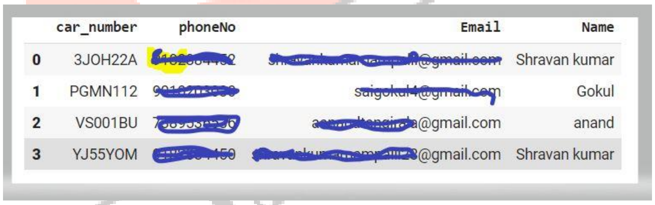
Example for angled number plate detection



Figures 9,10,11,12 showing number plate.

VI. Future Work And Conclusion

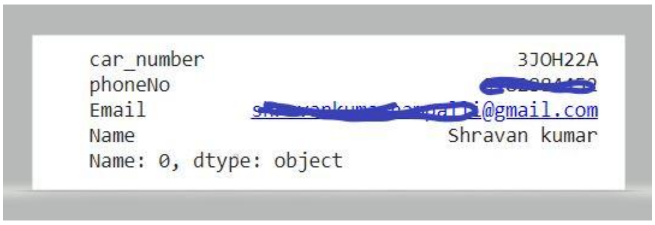
Our future work is as follows , we compare the extracted number text to our existing database to retrieve the owner information , some work is shown below.



	car_number	phoneNo	Email	Name
0	3JOH22A	[redacted]	[redacted]@gmail.com	Shravan kumar
1	PGMN112	[redacted]	[redacted]	Gokul
2	VS001BU	[redacted]	[redacted]@gmail.com	anand
3	YJ55YOM	[redacted]	[redacted]@gmail.com	Shravan kumar

Fig 13: Sample data set

So from above set we compare the car number to get owner information as shown ,



```
car_number                    3JOH22A
phoneNo                    [redacted]
Email         [redacted]i@gmail.com
Name                    Shravan kumar
Name: 0, dtype: object
```

Fig 14: retrieving owner information

We can use this information to find stolen vehicles, original owner and we can add data to the dataset if we required respectively.

In this paper, the automatic number plate recognition system using vehicle license plate is introduced. The system utilizes image processing techniques for recognizing the vehicle from the database stored in the computer by user. In the current system it process only horizontal number plates , in our proposed system we can process angled number plates too. First we select the image, remove noise and find the interested area of image, then the license plate location is extracted using edge detection then segmentation of each characters individually. At-last the template matching method is used with the use of keras for recognition of each characters in the number plate.

REFERENCES

[1] Singh, A. K., & Roy, S. (2015). "ANPR Indian system using surveillance cameras",2015 Eighth International Conference on ContemporaryComputing(IC3).

[2] Rahim Panahi and Iman Gholampour "Accurate Detection and Recognition of Dirty Vehicle Plate Numbers for High-Speed Applications",IEEE Transactions on intelligent transportation systems, vol. 18, no. 4, april2017.

[3]https://www.google.com/search?q=about+python+language&rlz=1C1CHBF_enIN856IN856&oq=about+python+language&aqs=chrome.0.69i59j0l3j0i22i30l6.6401j0j7&sourceid=chrome&ie=UTF-8

[4]https://www.google.com/search?q=about+google+colab&rlz=1C1CHBF_enIN856IN856&oq=about+google+colab&aqs=chrome.0.69i59j0i22i30l5j0i390.5060j0j4&sourceid=chrome&ie=UTF-8

[5] https://www.google.com/search?q=about+opencv&rlz=1C1CHBF_enIN856IN856&oq=about+opencv&aqs=chrome.0.69i59j0i20i263j0j0i22i30l2.3062j0j7&sourceid=chrome&ie=UTF-8

[6] https://www.google.com/search?q=about+keras+ocr&rlz=1C1CHBF_enIN856IN856&oq=about+keras+ocr&aqs=chrome.0.69i59j0j0i433j0l2j69i60l2j69i61.2808j0j7&sourceid=chrome&ie=UTF-8

[7] https://docs.opencv.org/master/da/d22/tutorial_py_canny.html

[8] https://keras-ocr.readthedocs.io/en/latest/examples/using_pretrained_models.html