2024 2nd International Conference on Intelligent Data Communication Technologies and Internet of Things (IDCIoT) | 979-8-3503-2753-3/24/$31.00 ©2024 IEEE | DOI: 10.1109/IDCIOT59759.2024.10467269

Number Plate Detection And Recognition using OpenCV

Akshat Parikh
KDPIT, CSPIT
CHARUSAT, INDIA
20it078@charusat.edu.in

Harita Parikh
KDPIT, CSPIT
CHARUSAT, INDIA
20it079@charusat.edu.in

Khush Patel
KDPIT, CSPIT
CHARUSAT, INDIA
20it098@charusat.edu.in

Dr. Nirav bhatt
Department of Artifical Intelligence
and Machine Learning, CSPIT
CHARUSAT,INDIA
niravbhatt.it@charusat.ac.in

Dr. Purvi Prajapati
KDPIT, CSPIT
CHARUSAT, INDIA
purviprajapati.it@charusat.ac.in

***Abstract*— Number Plate Recognition using image processing and Optical Character Recognition (OCR) techniques aims to create an efficient automatic system for authorized vehicle identification. The proposed method involves a comprehensive iterative waterfall model for software development, emphasizing flexibility and risk management. Challenges include handling diverse fonts, languages, and image variations, such as clear, blurred, and skewed plates. The system employs OpenCV for image manipulation, Tesseract OCR for text extraction, and Pandas for database matching. The objectives encompass accurate license plate detection, character recognition, and extraction of vehicle information from a dataset. The model demonstrates an 83% accuracy rate and offers potential applications in traffic management, surveillance, and toll collection. Future enhancements include extending the system to detect international plates, various languages, real-time applications, and integration into traffic infrastructure for enhanced functionality.**

Keywords*— Number plate recognition, OpenCV, Optical Character Recognition, Image processing, Character Segmentation

I. INTRODUCTION

Number plate recognition is an image processing technology that uses a number (license) plate to identify the vehicle. The objective is to design an efficient automatic authorized vehicle identification system by using the vehicle number plate. The main objective of this project is to detect the number plate on vehicles and to extract the information of the owner from it. Our project mainly uses the techniques of image processing and Optical character recognition technique. This technique detects the number plate on the vehicle, using the image processing algorithms, it filters the image of the number plate and enhances the character on it, the OCR algorithm recognizes the character on it and gives us the vehicle registration number with this information we can use our dataset and fetch the information of the owner. The developed system first detects the vehicle and then captures the vehicle image. The vehicle number plate region is then converted into grayscale. The number plate is then extracted. After that, the characters of the number plate are extracted and matched with database entries to extract vehicle information. The system is implemented using Python,and its performance is tested on real images.

II. LITERATURE STUDY

First Optical character recognition is a technique of identifying different characters from a recorded picture or printed/handwritten characters, also helps to recognize numerical texts, symbols and ASCII values, etc. It generally includes 3 main categories: preprocessing, feature extraction, and recognition. The study is carried out on different language characters such as English, Chinese, Farsi, Kannada, Arabic, and so on. The paper covers English scripter character recognition which has samples that are divided in the ratio of 80:20 of training to testing and uses the support vector system for recognition part. Second, comes the Arabic scripter character recognition using a neural network. Here it conducted a test on their dataset (43000 data entries) and found an accuracy of 99% lastly it included Devanagari scripter character recognition that uses k-nearest neighbour for recognition. The multilingual environment led to the problem for OCR of dividing and extracting the character correctly [1] OCR is an automatic identification technique that enables machines to read information from various sources, including handwritten text and printed characters. The paper covers stages in text recognition, including preprocessing, feature extraction, recognition, and post-processing, as well as segmentation and morphological processing. It also touches on recent research in OCR, such as multilingual character recognition and its applications in fields like electricity billing and fast scanning of robotic eyes. It is to be noticed that the segmentation-free approach using DNN is also possible in OCR. The research aims to improve OCR accuracy and efficiency, bridging the gap between humans and machines in text recognition [2] The tesseract OCR is widely used to extract text from an image or document that does not have a text layer, and output it to a new text file that can be searched, such as a PDF, CSV, or other popular formats. Mainly it has a collection of features like the line finding, baseline finding, chopping joined characters or broken characters, and also the static character classifier, etc which can be considered to be the key strengths of Tesseract. However, the underlying flaw related to it is the use of polygonal approximation as input rather than raw outlines [3] The model captures the image from the camera, and converts the RGB image to grayscale using MATLAB. The objective of the research paper was to capture the number plate and then generate the e-challan from the database. Here, Optical Character Recognition (OCR) is used and the RGB image is converted to a grey scale format using MATLAB. The major drawbacks of this technique are that the system could only detect white colour number plates and the image should be clear [4] There are various techniques for license plate detection, one such technique is to convert the captured image into a thresholding image and then apply the recognition part by normal methods such as character detection, character classifier, etc and then match the character feature with reference features using the statistical nearest neighbour classifier. This process uses a template for storing and comparing. Having a pipeline structure where all components play a statistically independent role and having the accuracy of 95% in feature extraction and 80% in feature matching leads to higher accuracy performance but due to the threshold image, the image is not well segmented as it would be in tesseract OCR [5] Automatic License Plate Recognition using OCR is used to identify vehicle owner, where android cameras are used to capture the image. ML-Kit is an additional library that extracts number plates using an Android camera. Accuracy for well-maintained and clear number plates is 78.57% while for unclear number plates, it is 57.41%. The major problem was recognition failures due to the resemblance of characters to each other, such as 2 and Z, 8 and B, and 0 and O [6] The proposed system uses the OCR mechanism to segment the characters of the number plate and extracts the numbers and alphabets from it. The ANPR system then compares the data with the universal database provided by MySQL. It uses the conventional ANPR method for image acquisition and morphological operations. The system uses ANN and KNN as its major algorithms but the morphological operations are done using conventional technologies like Normal grayscale and standardized edge detection. Furthermore, the methodology for real-time capturing is also not feasible. The model shows drastic variations of the accuracy under different circumstances like clear plates, blurred plates, and skewed plates, which average with a weighted score of approximately 65 [7] ANPR system here claims to be helpful in the management of parking, to speed up the process of toll collection, to locate stolen vehicles by using the license plate number and to maintain traffic rules and regulations. A custom YOLOv5 model is used only for number plate detection and then the number plate is segmented. The You Only Look Once (YOLO) family of computer vision models includes the model known as YOLOv5. YOLOv5 is frequently employed for object detection. The idea of "spatial pyramid pooling" (SPP), a kind of pooling layer used to lower the spatial resolution of the feature maps, is also introduced in YOLO v5. Since SPP enables the model to view objects at various scales, it is employed to enhance the detection performance for small objects. The system later uses OCR to recognize the segmented characters on the number plate. In this model, morphological operations are not used to enhance the image for better detection of number plates. The whole process involves four steps the acquisition of image, the license plate detection using YOLOv5 after that, the number plate is segmented using plate segmentation and lastly, the characters are recognised using optical character recognition (OCR). The model also focuses on the detection of number plates with diversified fonts and languages. They further wish to work on number plates with different colours and different shapes and sizes across different regions of India. The proposed model offers a precision of 0.629 recall of 0.943 and an approximate accuracy of 94% when performed across a dataset with 500 images [8] In the inquiry conducted by the Vision Based Panel, four factors will be examined. The first element that will be examined is the picture's aspect ratio. The size of the discovered blob is compared to what is required for the potential license plate during this key phase. In the second stage, the image's alignment or rotation is evaluated. The third element is the Euclidean distance. To evaluate the 2-pixel resolution camera's capacity to establish the precise range at which a license plate can be recognized and detected; this particular measurement is necessary. The binary threshold value is calculated in the fourth stage using the Global Threshold and Adaptive Threshold [9] The ALPR (Automatic license plate recognition/reader) helps to extract the vehicle license plate. Traditionally, the recognition takes place with the help of machine learning techniques such as morphological attributes and now it is done through character segmentation and OCR. The traditional recognition methods such as the LPR system had poor reconstruction rate and so many efforts were made by installing HD cameras for better image and to improve the recognition rate. The application intended to recognise number plates on the following characteristics Nigerian plate numbers rectangle plates third single plate and arrangement of letters/numbers having an accuracy of 80 % success rate [10] The number plate detection can also use YOLOV4 which is a one-stage object detection model. It utilizes the CNN model to recognize. It is supported by a single conventional neural arrangement. Here the CNN separates a picture into areas and after that predicts a few boundary boxes and probabilities. Now the database will be prepared to permit plays and make boundaries around them. And now the dark scaling method is used to recognize the number plate. The accuracy for the character recognition is 100% but for the license plate detection is 88% [11] The paper utilizes IoT technology to capture images for vehicle surveillance. The system employs image processing techniques, including pre-processing for optimal Optical Character Recognition (OCR) results. The technology stack includes Raspberry Pi, OpenCV, and GPS for image capture, and the developed system is accessible through a responsive web interface. The system detects irregular vehicles and notifies a network of users via email. It combines hardware and software to enhance property security in remote areas with limited surveillance investments. The results and discussions section presents the validation of a prototype for identifying irregular vehicles. The chosen algorithm, Open ALPR, achieved an 81.48% accuracy in recognizing license plates in various conditions. Camera positioning, lighting, and temperature were found to impact recognition. Communication methods (3G, Wi-Fi, cable) showed Wi-Fi as the most feasible due to faster speeds. The system has potential applications in public safety but faces limitations, including nighttime recognition and cost considerations. Future work could explore advanced methods like YOLO for license plate position detection [12]

III. Proposed Method

A. Flowchart

The software development approach used for the project: Iterative Waterfall Model This includes the stages like:

- Feasibility Study
- Requirement Analysis

- Design
- Coding
- Testing
- Maintenance

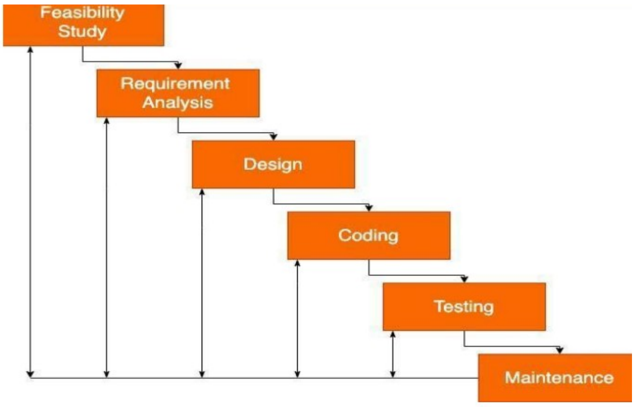


Fig 3.1.1 Iterative Waterfall Model

The approach is selected because it gives the liberty to get back to a stage if it needs some redo. The entire approach was trial and error and learn as you need. By choosing the iterative waterfall, the liberty of flexibility and risk management can be done easily.

This software, when fully functional, will consume a substantial amount of the server bandwidth and memory. Thus, an immense amount of reliable hardware is required for this service to function flawlessly.

- Maintainability

Over time the data collected by this software will grow exponentially. Thus, it should be possible to maintain this data in case of any change in the software architecture and the development process.

- Security

The software can store a massive amount of user data which will tend to grow over time. Since this software is supposed to be used by developers to track their web traffic, any leak in data provided by the users, can not only harm this platform but potentially harm their web application as well. Thus, the user data should be stored securely and an adequate amount of encryption should be administered to achieve security we can use several techniques like Access control, Secure Communication Protocols, network security, and so on.

- Performance and Reliability

Web developers and administrators will be relying on the software to make strategies and change their websites based on the results provided by the application. Thus, it is very important for the maximum availability and reliability of the web server and to achieve so various things like ensuring scalability, Optimized Data Storage, and so on. Apart from that parallel processing can be added to reduce the load.

B. Algorithm/ Model Steps

STEP 1:

The imread library is employed to import images into an application, extracting pixel information for further processing, and then matplotlib is utilized to plot axes around the image, providing a visual reference for pixel identification. This facilitates a systematic approach for the computer to interpret and manipulate individual pixel values, enhancing the overall efficiency of image analysis, processing, and visualization within the application.

STEP 2:

The cvtColor function in OpenCV is employed to convert the original image into a greyscale image, reducing complexity for computational analysis. By transforming the image to greyscale, complicated color information is discarded, simplifying subsequent processing tasks. It also uses the primary pigments for better recognition of the number plate.

STEP 3:

The OpenCV bilateral filter function is used to smoothen out the image to reduce the excess noise and remove blurry areas. By averaging technique, the filter smoothens the image while preserving important edges. This pre-processing step is particularly beneficial in refining image clarity, ensuring that subsequent computer processing tasks are not hindered by unwanted noise or unclear areas, thus improving overall image analysis accuracy.

STEP 4:

The OpenCV Canny function is utilized to point up the contours of an image, highlighting only the edges. This process enhances the visibility of structural features, emphasizing the edges where significant changes in intensity occur. By focusing on these pronounced boundaries, the algorithm improves the identification of rectangular shapes, specifically assisting in the recognition of number plates within the image.

STEP 5:

Utilizing OpenCV's "Find Contours" and "Draw Contours" functions is important in recognizing rectangle boxes within an image. The "Find Contours" function identifies closed objects, while "Draw Contours" outlines these objects. By detecting contours and drawing them on the image, the algorithm identifies rectangular shapes which is the Number plate.

STEP 6:

Now the masking technique is employed to precisely extract the number plate from a greyscale image. This process involves isolating the relevant area using a mask and a cropped number plate is then displayed, offering a focused and enhanced view.

STEP 7:

Now Tesseract OCR (Optical Character Recognition) image-to-string function is pivotal in extracting numerical information from images, such as number plates. By converting image text into a string, Tesseract enables seamless decoding of numeric data in a readable form that a computer can understand and use in applications requiring automated recognition of numbers from visual inputs.

STEP 8:

Now to recall details of the car from the testicle database Excel, the program reads the relevant sheet, searching for matching records. Upon finding a match, comprehensive information, including car details and owner-related data, is extracted and printed as the final output. This systematic approach ensures an accurate and organized retrieval of pertinent details, streamlining the process of accessing and presenting information stored in the Excel database.

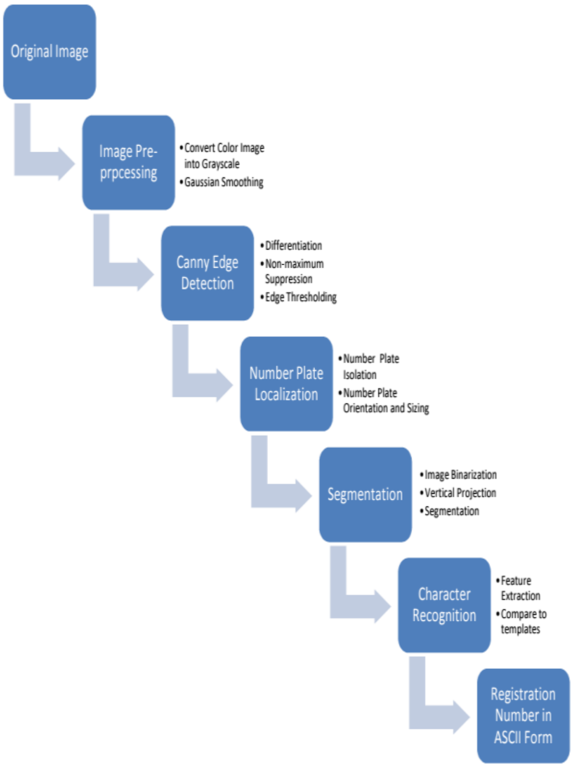


Fig 3.2.1 System Flow Diagram

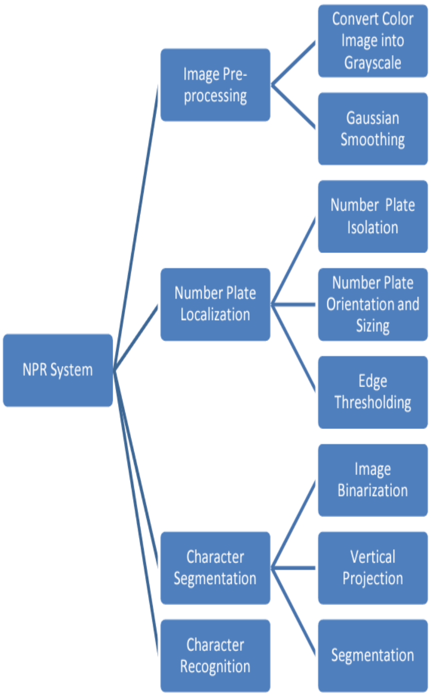


Fig 3.2.2 Product Function Diagram

OPTICAL CHARACTER RECOGNITION

OCR (optical character recognition) is a technique that uses digital images of real-world documents, like scanned paper documents, to identify printed or handwritten text characters. OCR's fundamental procedure entails reading a document's text and converting the symbols into a code that can be used for data processing. The major steps involved in OCR are as follows:

1. Line Finding
2. Baseline Finding
3. Fixed Pitch Detection and Chopping
4. Proportional Word Finding
5. Word Recognition
6. Chopping Joined Characters
7. Associating Broken Characters
8. Character Classification

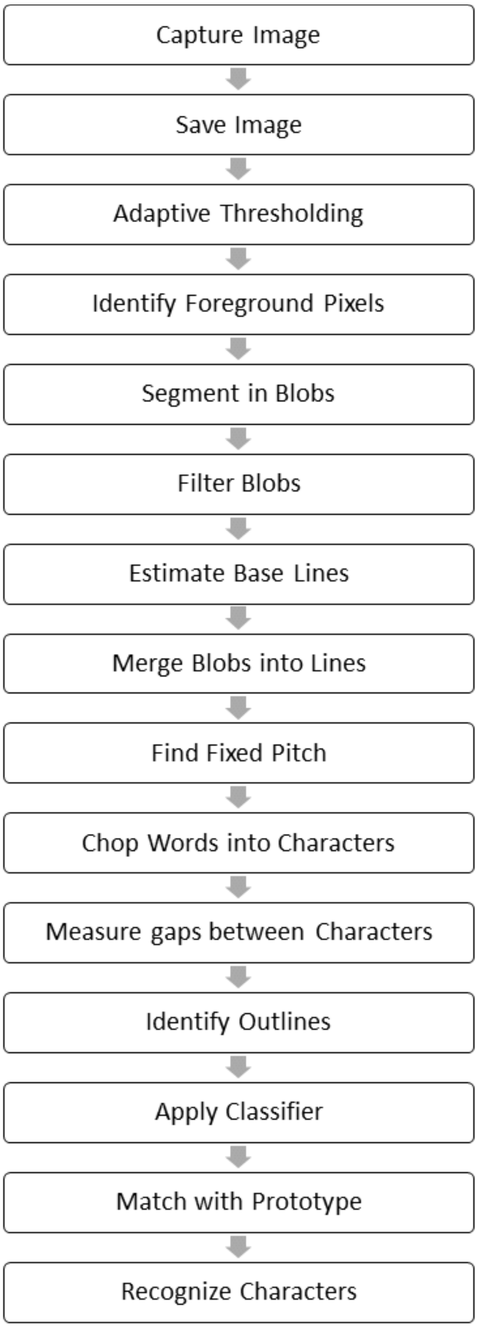


Fig 3.2.3 OCR Flow Diagram

C. Implementation Environment

During the complete implementation, we worked on Jupyter Notebook and Anaconda as our main software to develop the project. The project language was Python as it has several libraries that make the whole task easier. Jupyiter Notebook was used because it runs the code block by block and shows output at the end of each block so it becomes easier to understand what the machine is doing instead of only seeing the final output.

For calling the data we have used an Excel sheet where the project searches for the numberplate in Excel and then once the match is found the application will show every detail that is available in the Excel sheet

The following Python libraries were used to complete the following number plate detection project

- OpenCV – It is used to read and write images, process images (filter, transform), perform feature detection
- Imutils - A function to make basic image processing functions such as resizing, skeletonization, displaying Matplotlib images, sorting contours, detecting edges, etc
- Numpy – It is used to work with arrays.
- Matplotlib – It is used to plot the images.
- Pandas – To access the Database i.e. excel or CSV sheets
- Pytesseract – It is used to read and recognize text in images
- Termcolor - The Termcolor module is a Python module for ANSII Color formatting for output in the terminal.
- Xlrd - xlrd is a library for reading data and formatting information from Excel files in the historical.

IV. RESULTS AND DISCUSSION

Firstly, the original image is captured and read by OpenCV.

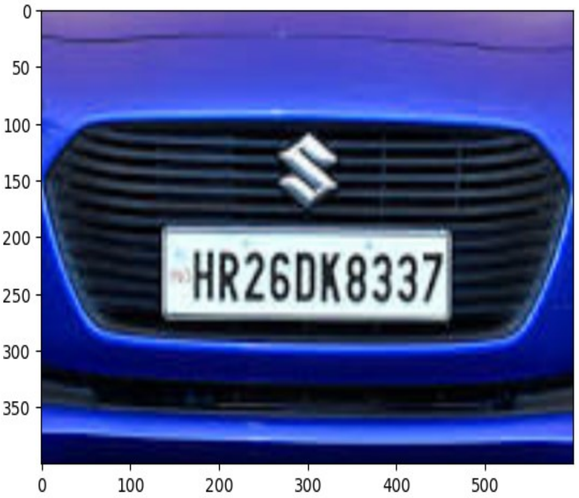


Fig 4.1 Original Image

In the second step, we will convert the image from RGB to a greyscale image for further ease of character recognition. The greyscale image helps to reduce the complexities and to eradicate and simplify the algorithm. Alternatively, it uses equal amounts of cyan, yellow, and magenta which are the primary pigments. Each pixel is a representation of the luminous intensity of the image.

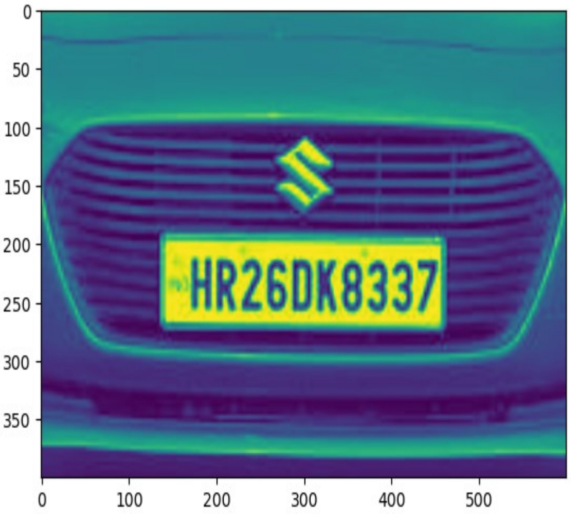


Fig 4.2 Grey Scale Image

In the third step, the image is smoothened. A process called image smoothing is used to remove noise, sharpness, and clutter from the image to produce a much more blended and smooth effect. Average smoothing calculates the average of all the pixels in the kernel area and replaces the central element with the average.

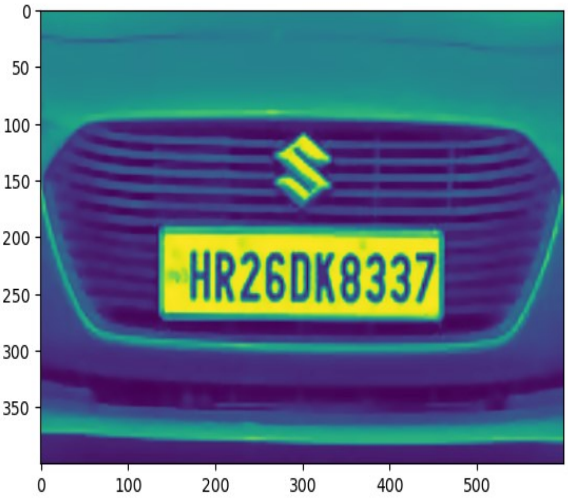


Fig 4.3 Smoothened Image

In the fourth step, we use the edge detection algorithm on the image. Edge detection is a method of image processing that locates the edges of things in pictures. It operates by looking for changes in luminance. In fields like image processing, computer vision, and machine vision, edge detection is used for picture segmentation and data extraction. With the use of the Canny edge detection method, the amount of data that needs to be processed can be drastically reduced while still extracting useful structural information from various vision objects. It is frequently used in different computer vision systems.

The Canny edge detection algorithm is composed of 5 steps:

i. Noise reduction
ii. Gradient calculation

iii. Non-maximum suppression
iv. Double threshold
v. Edge Tracking by Hysteresis

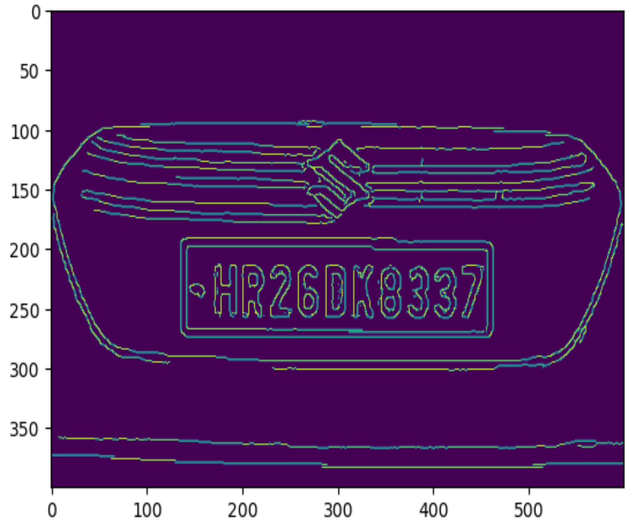


Fig 4.4-Edged Image

In the fifth step, we contour the image. By locating the structural outlines of objects in a picture, image contouring allows us to determine the shape of the object. In general, the accuracy of a contour tracing algorithm can be assessed based on the following four factors: (1) processing time; (2) data size required to save the traced contour information; and (3) accuracy of the ability to accurately restore and enlarge the original contour using the saved data.

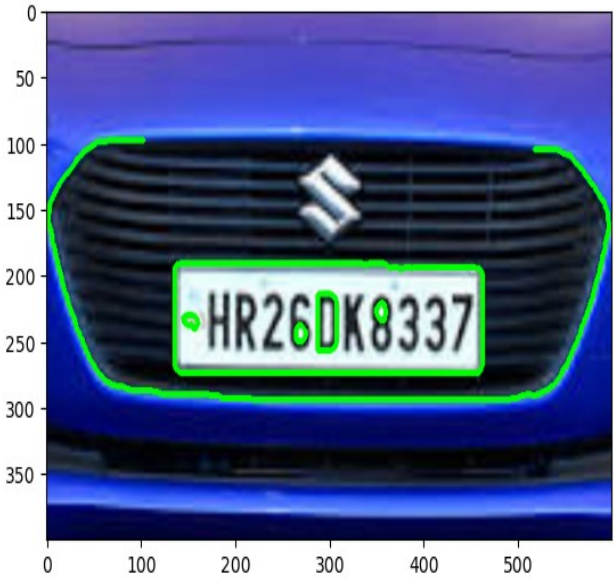


Fig 4.5 Contour Image

Furthermore, the image is cropped for accuracy.

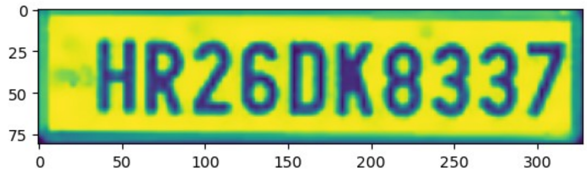
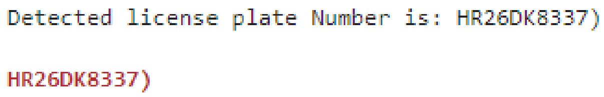


Fig 4.6 Cropped Image

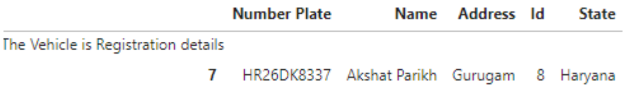


```
Detected license plate Number is: HR26DK8337)

HR26DK8337)
```

Fig 4.7 Number Plate

After the registration number is fetched from the number plate, we can look into the available data using machine learning algorithms and get the owner data from it.



```
                      Number Plate      Name    Address  Id   State
The Vehicle is Registration details
                   7  HR26DK8337  Akshat Parikh  Gurugam  8  Haryana
```

Fig 4.8 Final Output

V. Conclusion And Future Scope

A. Conclusion

This model uses OpenCV operations to manipulate the image and then uses Tesseract OCR to recognize text from an image furthermore, it uses Pandas library to match the data. In the proposed system, firstly the image is read using OpenCV then it is converted to a greyscale to reduce complexities from the image. Following this step, the image is smoothened to remove noise from it then, using the canny edge detection algorithm the main edge in the image is detected which includes the edges of the license plate. Contours are used to detect the appropriate edges for the license plate and hence the cropped license plate is recognized. After performing such morphological operations on the image, the number is extracted from the image using Optical Character Recognition. Hence, the final output is obtained. This model helps to detect number plates in both day and night vision and accurately provides the result. The accuracy of our model across various parameters is 83%.

B. Future Enhancements

- Can be used in detecting different countries' number plates.
- Can be used to detect different languages.
- Can be used for the detection of number plates in real time.
- Can be used in traffic signal cameras to create challans.
- Can be used in toll plazas for automated toll deduction.
- Can be used in police patrol vehicles to ensure people's safety while patrolling.
- Can be used to manage journey time analysis.
- Can be used for security and surveillance.

BIBLIOGRAPHY

- https://pyimagesearch.com/2020/09/21/opencv-automatic-license-number-plate-recognition-anpr-with-python/
- https://www.section.io/engineering-education/licenseplate-detection-and-recognition-using-opencv-and-pytesseract/
- https://pypi.org/project/opencv-python/
- https://docs.opencv.org/4.x/d6/d00/tutorial_py_root.html

REFERENCES

[1] Muna Ahmed Awel, Ali Imam Abidi, "REVIEW ON OPTICAL CHARACTER RECOGNITION", Computer Science and Engineering, Sharda University, Greater Noida, India

[2] K.Karthick, K.B.Ravindrakumar, R.Francis, S.Ilankannan, "Steps Involved in Text Recognition and Recent Research in OCR; A Study".

[3] Ray Smith, Google Inc, "An Overview of the Tesseract OCR Engine", Ninth International Conference on Document Analysis and Recognition (ICDAR 2007).

[4] Abhay Singh, Anand Kumar Gupta, Anmol Singh, Anuj Gupta, Sherish Johri, "VEHICLE NUMBER PLATE DETECTION USING IMAGE PROCESSING", IMS Engineering College, Ghaziabad, India.

[5] R A. Lotufo, A. D. Morgan, A. S. Johnson, "AUTOMATIC NUMBER-PLATE RECOGNITION", Faculty of Engineering University of Bristol Bristol BS8 1TR

[6] U Salimah, V Maharani, R Nursyanti, "AUTOMATIC LICENSE PLATE RECOGNITION USING OPTICAL CHARACTER", Faculty of Technology and Informatics, Indonesian University of Informatics and Business.

[7] Monika Arora, Anubha Jain, Shubham Rustagi, Tushar Yadav, "AUTOMATIC NUMBER PLATE RECOGNITION SYSTEM USING OPTICAL CHARACTER RECOGNITION", Department of I.T., Bhagwan Parshuram Institute of Technology, Delhi, India.

[8] Gayatri Hajare, Utkarsh Kharche, Pritam Mahajan, Apurva Shinde, "AUTOMATIC NUMBER PLATE RECOGNITION SYSTEM FOR INDIAN NUMBER PLATES USING MACHINE LEARNING TECHNIQUES", Department of Computer Engineering, Ramrao Adik Institute of Technology D.Y Patil Deemed to be University, Nerul.

[9] P.Swathi, N,Shree Haritha, G.Venkata Sai Krishna, A.Veena Jasmine, V.Vamsi Krishna, Y.Sai Dinesh Reddy, "VEHICLE LICENSE PLATE DETECTION OPENCV AND TESSERACT OCR", Department of Computer Science and Engineering, Annamacharya Institute of Technology and Sciences, Rajampet, Andhra Pradesh, India.

[10] Matthew B. Olajide, Najeem O. Adelaku, David S. Kuponiyi , Zaid O. Jagun, Charity S. Odeyemi, "DESIGN OF AN AUTOMATIC LICENSE PLATE READER", Olabisi Onabanjo University, Federal College of Education Iwo, Gateway (ICT) Polytechnic Saapade, Federal University of Technology Akure, Nigeria.

[11] G. Poorani, B.R. Krithick Krishnna, T. Raahul, P. Praveen Kumar, "NUMBER PLATE DETECTION USING YOLOV4 AND TESSERACT OCR", 1Department of Computer Science Engineering, Sri Krishna College of Technology, Coimbatore, Tamil Nadu, India.

[12] Luiz Alfonso Glasenapp, Aurélio Faustino Hoppe, Miguel Alexandre Wisintainer, Andreza Sartori, Stefano Frizzo Stefenon, "OCR APPLIED FOR IDENTIFICATION OF VEHICLES WITH IRREGULAR DOCUMENTATION USING IOT", Regional University of Blumenau-Brazil, Digital Industry Center-Italy, University of Udine-Italy.