



**DEPARTMENT OF COMPUTER SCIENCE ENGINEERING  
SCHOOL OF ENGINEERING AND TECHNOLOGY,  
SHARDA UNIVERSITY, GREATER NOIDA**

## **LICENSE PLATE DETECTION USING OPEN CV**

**A project submitted in partial fulfillment of the requirements for the degree of Bachelor  
of Technology in Computer Science and Engineering**

**By**

**Himanshu Goyal - 2019001426  
Anushka Chaubey - 2019004401  
Jagriti Pal - 2019528733  
Himanshu Tripathi - 2019622995**

**Supervised By**

**Dr. Jagdeep Singh , Assistant Professor**

**Term - 2001**

# ACKNOWLEDGEMENT

We express our deep gratitude to **Prof.(Dr.) Nitin Rakesh ,Head,Department of Computer Science and Engineering .**

We are thankful to **Prof.(Dr.) Jagdeep Singh ,Project Mentor** for their valuable guidance.

We would like to express our gratitude towards my parents & member of **SHARDA UNIVERSITY** for their kind co-operation and encouragement which help us in working of this project.

We would like to express our gratitude towards my parents & member of **SHARDA UNIVERSITY** for their kind co-operation and encouragement which help us in working of this project.

We would like to express our special gratitude and thanks to faculties for giving us such attention and time.

Our thanks and appreciations also go to our colleague in developing the project and people who have willingly helped me out with their abilities.

# UNDERTAKING

This is to declare that the project entitled "**License Plate Detection Using Open CV**" is an original work done by undersigned, in partial fulfillment of the requirements for the degree "**Bachelor of Technology in Computer Science and Engineering**" at **Department of Computer Science Engineering, School of Engineering and Technology, Sharda University, Greater Noida, Uttar Pradesh.**

**Student 1 :- Himanshu Goyal**

**Student 2 :- Anushka Chaubey**

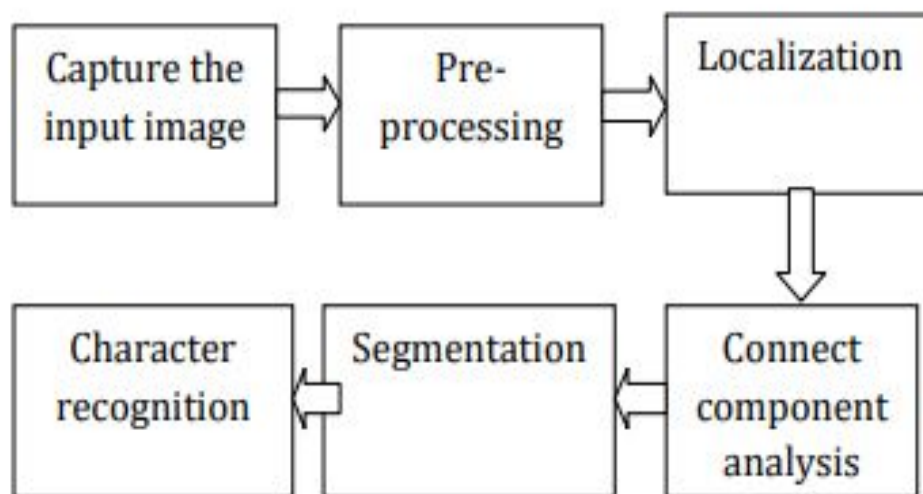
**Student 3 :- Jagriti Pal**

**Student 4 :- Himanshu Tripathi**

## ABSTRACT

License Plate Recognition was a computer system that recognizes any digital image automatically on the number plate. This system includes various operations such as taking pictures, localizing the number pad, truncating characters and OCR from alphanumeric characters. The main idea of this system is to design and develop effective image processing techniques and algorithms to localize the license plate in the captured image, to divide the characters from that number plate and to identify each character of the segment by using the KNN Algorithm and Open Computer Vision Library. This has been implemented in python programming language. Many applications can be implemented by using this system, such as security, highway speed detection, violation of light, identification of handwritten text, discovery of stolen cars, automatic fee collection systems

## Block Diagram



## >CAPTURE THE INPUT IMAGE:

The car's number pad is taken from a high resolution camera. The resolution of the number plate recognition system depends on the captured image. The image captured in RGB format must be converted to a gray image .

## >PRE-PROCESSING:

Pre-processing is a set of algorithms applied to the image to improve the quality by which the gray image is converted to a binary image. Before converting to a binary image, the image is smoothed to reduce noise. Pre-processing can be done by the threshold algorithm.

There is a different kind of threshold like

- \*Global threshold
- \*Adaptive mean threshold
- \*Adaptive Gaussian threshold

## >NUMBER PLATE LOCALIZATION:

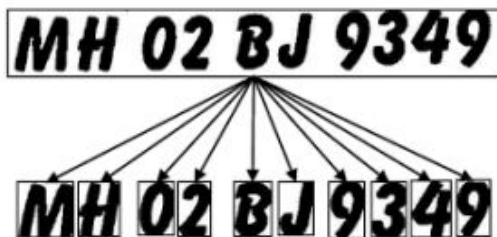
The license plate is extracted using either a shape analysis or a color analysis method. In the General License Panel has in form of a rectangular shape. Thus, algorithms look for geometrical shapes of a rectangular proportion. In India, most license plates are white or yellow, and therefore can also use color analysis. Before you find the rectangle in an image, the image must be in a binary image or the edges of the image should be detected. Then you should find and connect to the relevant rectangular corners. Finally, the areas connected to the box are connected and all rectangular areas of interest are extracted

## >CONNECT COMPONENT ANALYSIS:

To remove the unwanted image space, the algorithm of the component connected to the binary filter is applied first. The parsing of the connected component is done to determine the characters in the image. The basic proposal is to pass through the image and find a connected pixel. Each component (dots) is distinguished and extracted.

## > SEGMENTATION:

Once the license plate has been extracted, each character must be fragmented. For component division, the component label is used to see the computer in order to discover the connected areas in binary digital images. The label of connected components works by scanning a pixel-in-pixel image from top to down to find connected pixels and connected pixel cards.



## > CHARACTER RECOGNITION:

To identify characters, the segmented characters in the license panel must match the templates that are already created. The recognition process returns the license number in ASCII format and saves it in a text document. In this recognition is a two-track process. In the first pass, an attempt was made to identify each word in turn. Each satisfactory word is passed to the adaptive workbook as training data. The adaptive workbook gets an opportunity to learn the text more accurately

## ALGORITHM PROCEDURE

- 1: Begin
- 2: Input: Original Image
- 3: Output: Characters
- 4: Method: K-Nearest Neighbors
- 5: LP: License Plate
- 6: Convert RGB image to Grayscale
- 7: Filter Morphological Transformation
- 8: Transforms Grayscale image to binary image
- 9: Filter Gaussian for Blurs image
- 10: Finding all contours in image
- 11: Search & recognize all possible character in image
- 12: Crop part of image with highest candidate LP
- 13: Crop the LP from original image
- 14: Apply steps from 6 to 11 again on crop image
- 15: Print the characters in LP
- 16: End



## Steps With Images



1. Input Image



2. RGB to Grayscale image



3. Morphological Operations



4. Gray to Binary



5. Finding all Contours and Blobs in the Image



6. Output of Contour Image



7. Cropped the Image



8. RGB to Gray [Apply the Steps from 6 to 8]



## 9. Morphological Operations



## 10. Finding all Contours and Blobs



## 11. Output of the Contours



## 12. Character Recognizing



## 13. Final Output

# Conclusion

Future extension of this work is to develop character recognition using template matching algorithm. Detecting number plate characters during night times work efficient but it gets inefficient in case of sunny time.

Thank

You !