**A Major-Project Report**

**on**

# AUTOMATIC NUMBER PLATE RECOGNITION WITH DEEP LEARNING

***Submitted in the partial fulfillment of the requirements for the Industry Oriented Major Project of***

**BACHELOR OF TECHNOLOGY**

in

**INFORMATION TECHNOLOGY**

Submitted by


| | |
|---|---|
| G CHARITHA | 18071A12D9 |
| K SAI DEEKSHITH | 18071A12E8 |
| M VARSHITHA | 18071A12F3 |
| M ASHOK VARMA | 18071A12F4 |


MAJOR PROJECT PHASE-1

TAMASŌMA JYOTIRGAMAYA


**DEPARTMENT OF INFORMATION TECHNOLOGY**

**VNR Vignana Jyothi Institute of Engineering & Technology**

(Autonomous Institute, Accredited by NAAC with 'A++' grade and NBA)

Bachupally, Nizampet (S.O.) Hyderabad- 500 090,


**January 2022**

**A Major-Project Report**

**on**

# AUTOMATIC NUMBER PLATE RECOGNITION WITH DEEP LEARNING

***Submitted in the partial fulfillment of the requirements for the Industry Oriented Major Project of***

**BACHELOR OF TECHNOLOGY**

In

**INFORMATION TECHNOLOGY**

Submitted by


| | |
|---|---|
| G CHARITHA | 18071A12D9 |
| K SAI DEEKSHITH | 18071A12E8 |
| M VARSHITHA | 18071A12F3 |
| M ASHOK VARMA | 18071A12F4 |


MAJOR PROJECT PHASE-1

**Under the esteemed guidance of**


**PROJECT GUIDE**

V. Manoj Kumar

Assistant Professor,

Dept. of Information Technology,

VNRVJIET

**DEPARTMENT OF INFORMATION TECHNOLOGY**

**VNR Vignana Jyothi Institute of Engineering & Technology**

(Autonomous Institute, Accredited by NAAC with ‘A++’ grade and NBA)

Bachupally, Nizampet (S.O.) Hyderabad- 500 090,


**January 2022**

**VNR Vignana Jyothi Institute of Engineering & Technology**
Autonomous Institute, Accredited by NAAC with 'A++' grade and
NBA) Bachupally, Nizampet (S.O.) Hyderabad- 500 090

**Department of Information Technology**

Date: January 2022

## CERTIFICATE

This is to certify that the major-project work entitled **"AUTOMATIC NUMBER PLATE RECOGNITION WITH DEEP LEARNING"** is being submitted by **Ms. Charitha Gundlapalli (18071A12D9), Mr. Komuravelli Sai Deekshith (18071A12E8), Ms. Varshitha Muddasani (18071A12F3), Mr. Mudunuri Ashok Varma (18071A12F4)** in partial fulfilment for the award of Degree of **BACHELOR OF TECHNOLOGY** in **INFORMATION TECHNOLOGY** to the Jawaharlal Nehru Technological University, Hyderabad during the academic year 2021-22 is a record of bona-fide work carried out by them under our guidance and supervision.

The results embodied in this report have not been submitted by the students to any other University or Institution for the award of any degree or diploma.

| **Project Guide** | **Head of Department** |
| --- | --- |
| **V. MANOJ KUMAR** | **Dr. D. SRINIVASA RAO** |
| **Assistant Professor** | **Head of Department,** |
| **Dept. of IT,** | **Dept. of IT,** |
| **VNRVJIET,** | **VNRVJIET,** |
| **Hyderabad.** | **Hyderabad.** |

# VNR Vignana Jyothi Institute of Engineering & Technology

Autonomous Institute, Accredited by NAAC with 'A++' grade and
NBA) Bachupally, Nizampet (S.O.) Hyderabad- 500090.

## Department of Information Technology

Date: January 2022

## DECLARATION

I hereby declare that the project entitled "Automatic number plate recognition with deep learning" submitted to VNR Vignana Jyothi Institute of Engineering and Technology in partial fulfillment of requirement of the award of Bachelor of Technology in Information Technology is a bona-fide report of work carried out by us under our guidance and supervision of Department of Information Technology, VNR Vignana Jyothi Institute of Engineering and Technology.
To the best of my knowledge, this has not been submitted in any form to any University or Institution for the award of any degree or diploma.

Signature of the Student:

| | | | |
|---|---|---|---|
| G CHARITHA | K SAI DEEKSHITH | M VARSHITHA | M ASHOK VARMA |
| (18071A12D9) | (18071A12E8) | (18071A12F3) | (18071A12F4) |

Place:
Date:

## ACKNOWLEDGEMENT

We express our deep sense of gratitude to our beloved **President, Sri D. Suresh Babu garu, VNR Vignana Jyothi Institute of Engineering &Technology** for the valuable support, and encouragement for permitting us to carry out this project work.

With immense pleasure, we record our deep sense of gratitude to our beloved **Principal, Dr. C. D. Naidu** for permitting us to carry out this project.

We express our deep sense of gratitude to our beloved HOD, **Dr. D. Srinivas Rao, Associate Professor and Head, Department of Information Technology, VNR Vignana Jyothi Institute of Engineering & Technology, Hyderabad-90** for the valuable guidance and suggestions, keen interest and through encouragement extended throughout period of project work.

We take immense pleasure to express our deep sense of gratitude to our beloved Guide **V. Manoj Kumar, Assistant Professor in Information Technology, VNR Vignana Jyothi Institute of Engineering & Technology, Hyderabad,** for his valuable suggestions and rare insights, for constant source of encouragement and inspiration throughout my project work.

We express our thanks to all those who contributed for the successful completion of our project work.

1. G CHARITHA ______________________
2. K SAI DEEKSHITH ______________________
3. M VARSHITHA ______________________
4. M ASHOK VARMA ______________________

**A Major-Project Report**

**on**

# AUTOMATIC NUMBER PLATE RECOGNITION WITH DEEP LEARNING

***Submitted in the partial fulfillment of the requirements for the Industry Oriented Major Project of***

**BACHELOR OF TECHNOLOGY**

in

**INFORMATION TECHNOLOGY**


Submitted by

G. CHARITHA 18071A12D9


MAJOR PROJECT PHASE-1

**Under the esteemed guidance of**


**PROJECT GUIDE**

V. Manoj Kumar

Assistant Professor

Dept. of Information Technology,

VNRVJIET

**DEPARTMENT OF INFORMATION TECHNOLOGY**

**VNR Vignana Jyothi Institute of Engineering & Technology**

(Autonomous Institute, Accredited by NAAC with 'A++' grade and NBA)

Bachupally, Nizampet (S.O.) Hyderabad- 500 090,


**January 2022**

**A Major-Project Report**
**on**

# AUTOMATIC NUMBER PLATE RECOGNITION WITH DEEP LEARNING

***Submitted in the partial fulfillment of the requirements for the Industry Oriented Major Project of***

**BACHELOR OF TECHNOLOGY**
in
**INFORMATION TECHNOLOGY**

Submitted by

K. SAI DEEKSHITH 18071A12E8

MAJOR PROJECT PHASE-1

**Under the esteemed guidance of**

**PROJECT GUIDE**

V. Manoj Kumar
Assistant Professor
Dept. of Information Technology,
VNRVJIET

**DEPARTMENT OF INFORMATION TECHNOLOGY**
**VNR Vignana Jyothi Institute of Engineering & Technology**
(Autonomous Institute, Accredited by NAAC with ‘A++’ grade and NBA)
Bachupally, Nizampet (S.O.) Hyderabad- 500 090,

**January 2022**

A Major-Project Report

on

# AUTOMATIC NUMBER PLATE RECOGNITION WITH DEEP LEARNING

***Submitted in the partial fulfillment of the requirements for the Industry Oriented Major Project of***

**BACHELOR OF TECHNOLOGY**

in

**INFORMATION TECHNOLOGY**

Submitted by


M. VARSHITHA 18071A12F3


MAJOR PROJECT PHASE-1

**Under the esteemed guidance of**


**PROJECT GUIDE**

V. Manoj Kumar

Assistant Professor

Dept. of Information Technology,

VNRVJIET

**DEPARTMENT OF INFORMATION TECHNOLOGY**

**VNR Vignana Jyothi Institute of Engineering & Technology**

(Autonomous Institute, Accredited by NAAC with ‘A++’ grade and NBA)

Bachupally, Nizampet (S.O.) Hyderabad- 500 090,


**January 2022**

**A Major-Project Report**
**on**

# AUTOMATIC NUMBER PLATE RECOGNITION WITH DEEP LEARNING

***Submitted in the partial fulfillment of the requirements for the Industry Oriented Major Project of***

**BACHELOR OF TECHNOLOGY**
In
**INFORMATION TECHNOLOGY**

Submitted by

M. ASHOK VARMA 18071A12F4

MAJOR PROJECT PHASE-1

**Under the esteemed guidance of**

**PROJECT GUIDE**

V. Manoj Kumar
Assistant Professor
Dept. of Information Technology,
VNRVJIET

**DEPARTMENT OF INFORMATION TECHNOLOGY**
**VNR Vignana Jyothi Institute of Engineering & Technology**
(Autonomous Institute, Accredited by NAAC with 'A++' grade and NBA)
Bachupally, Nizampet (S.O.) Hyderabad- 500 090,

**January 2022**

# INDEX

# LIST OF FIGURES

# ABSTRACT

In this day of rapidly evolving technology, individuals have a strong need for a safe way of life and travel. The number of automobiles on the road has grown during the last decade. With the tremendous development in the vehicular industry every day, tracking individual vehicles has become a very difficult undertaking. In the realm of image processing, it is a very prominent and active study topic. Different methods, techniques and algorithms have been developed to detect and recognize number plates. Nevertheless, due to the number plates characteristics that vary from one country to another in terms of numbering system, colors, language of characters, fonts, and size. Our main purpose here is to make this process simpler by trying to implement the recognition process by using deep learning algorithms so that the number plates can be recognized from the video captured by the cctv cameras on the roads or any pictures captured.

As previously said, with the fast growth in vehicle numbers, vehicle identification and traffic management are becoming increasingly difficult in every country. Furthermore, it is quite difficult for a traffic cop to recognize someone who is driving too quickly, breaking traffic regulations, and determining if the person is the owner or a thief from moving cars. As a result, they are unable to apprehend and prosecute such individuals. As a result, an Automatic License Plate Recognition system is required. Despite the fact that various LPR systems are available, the work remains difficult owing to unstructured license plate forms, wording on plates, vehicle speed, and lightning impacts.

Automatic Number Plate Recognition is an issue that has received a lot of attention and has a lot of successful solutions. Due of the differences in number plate characteristics throughout the world, these solutions are generally tailored for a specific environment. These attributes are employed in number plate recognition algorithms; thus, a universal solution would be difficult to achieve because the image analysis techniques used to develop these algorithms cannot guarantee 100% accuracy.

Our emphasis is to reduce the long computational time of existing models and to increase the efficiency of them so that the procedure to identify the number plates would be done easily. The method, which we have used to solve this issue is implemented in Python and uses the OpenCV library, locates the plate by combining canny edge recognition with mathematical morphology. The discovered characters on the plate were then identified using the Tesseract OCR engine.

# INTRODUCTION

The dramatic growth in vehicular traffic on the roads has created a great need for traffic monitoring and management technologies. Manual tracking of fast-moving automobiles on the road is almost impossible under this situation. There will be a waste of time and effort. Even if it's controlled manually, it'll reveal great challenges and faults. Using machine learning techniques, there are existing systems for tracking automobiles and license plates. However, due to their complexity for processing in the background in real time, these methods fail completely. As a result, there is an immediate need to design an automatic system that will assist in vehicle monitoring by efficiently tracing their number plates.

The scientific community is conducting research in areas related to intelligent transportation that have a substantial influence on people's lives. Vehicles are constantly rising in number nowadays, making it difficult to keep track of them all. Vehicle identification is required in a variety of situations, such as traffic, gate admission, and so on, to ensure security. The number plates on each vehicle serve as a unique identifier. Manually inputting vehicle license numbers will be a time-consuming task. Automatic number-plate recognition is a system that reads car registration plates and creates vehicle position data using optical character recognition on photographs.

The pictures taken by the cameras, as well as the text from the license plate, may be stored using automatic number-plate recognition. ANPR system must adjust for plate variances from location to location.

The following is how we define our problem statement:

***Creating an automatic number plate recognition system to identify the number plate of a vehicle from a picture of the vehicle, thereby saving time and effort.***

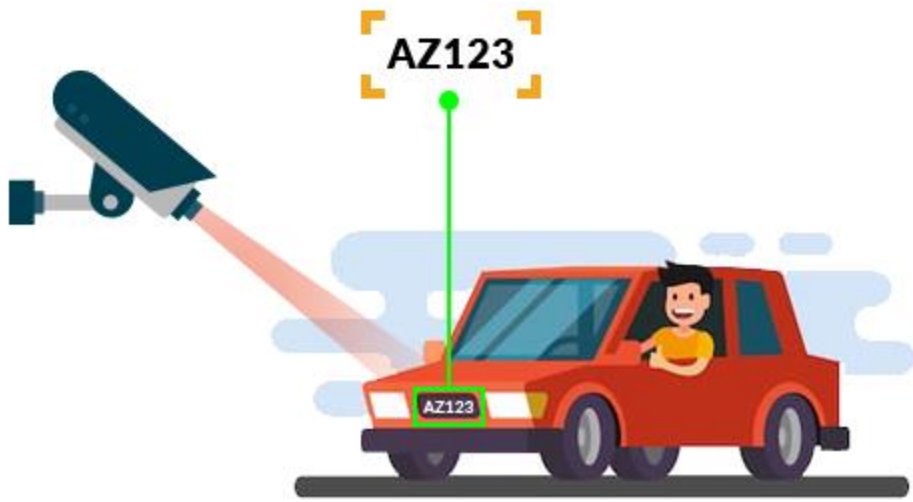


Fig 1.1: Real life ANPR example

ANPR is utilized by police agencies all around the world to verify if a vehicle is registered or licensed, among other things. It may also be used for electronic toll collection on pay-per-use roads and as a technique of cataloguing traffic movements by highways authorities. This technology is used to recognize and record license plate numbers, and it has benefits for law enforcement, security, and vehicle access. Secure and gated entries, traffic enforcement, police enforcement, and toll gates are all using license plate technology.

Our main objective through this project is to make the procedure of getting details about a number plate simpler, easier, and quicker.

There are a variety of existing models for identifying a vehicle's number plate. However, most number plate identification systems combine numerous steps, resulting in lengthy computing (and hence execution) times (this may be reduced by applying less and simpler algorithms). In certain cases, the individuals involved must manually study the videos or photos in order to capture the number plate.

We decided to address these problems and give a proper solution which can help to clear the above-mentioned problems. Thus, we have decided to create an automatic number plate recognition using Canny edge detection algorithm. This is not only an easy procedure to get the number of license plate but also an effective and better way to implement in fields of law enforcement, speed detection, traffic control and etc.

## LITERATURE SURVEY

- “An Embedded Automatic License Plate Recognition System Using Deep Learning” VIII Brazilian Symposium on Computing Systems Engineering (SBESC), 2018, pp. 38-45, doi: 10.1109/SBESC.2018.00015.

  Authors: D. M. F. Izidio, A. P. A. Ferreira and E. N. S. Barros

  This paper proposes an embedded method for detecting and recognizing Brazilian license plates using convolutional neural networks. The system was built using a Raspberry Pi 3 with a Pi NoIR v2 camera module to capture photos of moving automobiles.

- “Deep Learning System for Automatic License Plate Detection and Recognition” 2017 14th IAPR International Conference on Document Analysis and Recognition (ICDAR)

  Authors: Zied Selmia, Mohamed Ben Halimab , Umapada Pal c and M.Adel Alimi

  The authors of this work implemented the first Convolution Neural Network (CNN) model for the classification of plates / non-plates.

- "Automatic license plate recognition," in IEEE Transactions on Intelligent Transportation Systems, vol. 5, no. 1, pp. 42-53, March 2004, doi: 10.1109/TITS.2004.825086.

  Authors: Shyang-Lih Chang, Li-Shien Chen, Yun-Chung Chung and Sei-Wan Chen,

  Authors implemented a model consisting of two main modules: a license plate locating module and a license number identification module; former using fuzzy disciplines and latter using neural subjects. Fuzzy disciplines attempt to extract license plates from an input image, while neural subjects aim to identify the number present in a license plate.

- “Automatic License Plate Recognition System Based on Color Image Processing.” In: Gervasi O. et al. (eds) Computational Science and Its Applications – ICCSA 2005. ICCSA 2005. Lecture Notes in Computer Science, vol 3483. Springer, Berlin, Heidelberg.

  Authors: Shi X., Zhao W., Shen Y.

  The proposed system applied Color Image Processing for detecting the number plate. Their system can be divided into the following steps: preprocessing, plate region extraction, plate region thresholding, character segmentation, character recognition and post-processing.

- "Automatic license plate recognition using extracted features," 2016 4th International Symposium on Computational and Business Intelligence (ISCBI), 2016, pp. 221-225, doi: 10.1109/ISCBI.2016.7743288.

  Authors: N. Saleem, H. Muazzam, H. M. Tahir and U. Farooq

  The authors applied vertical edge detection algorithm and removed unwanted edges by image normalization technique. They used statistical and morphological image processing technique.

## EXISTING SYSTEM

Automatic number plate recognition is being used in various countries for multiple purposes like law enforcement, average-speed cameras, crime deterrent, enterprise security and services, traffic control and electronic toll collection. A few examples are:

•Australia: The Department of Justice (Victoria) and several state police forces employ both fixed and mobile ANPR systems. In 2005, the New South Wales Police Force Highway Patrol was the first in Australia to trial and utilize a fixed ANPR camera system. They started rolling out a mobile ANPR system (formally known as MANPR) with three infrared cameras to their Highway Patrol fleet in 2009. Unregistered and stolen cars, as well as disqualified or suspended drivers and other 'persons of interest,' such as those with outstanding warrants, are all identified by the system.

•France: Throughout the country, 180 gantries have been placed above main roadways. These, together with another 250 fixed cameras, will allow for the imposition of an environmental fee on vehicles weighing more than 3.5 tons. The method is presently being criticized, and while data on cars passing via the cameras may be collected, no environmental levy is levied.

•Saudi Arabia: In Saudi Arabia, automobile registration plates have a white backdrop, however various vehicle types may have a varied background. On the registration plates, there are just 17 Arabic letters.  The size of the digits in Saudi Arabia makes plate identification difficult. On certain plates, both Eastern Arabic numerals and their 'Western Arabic' counterparts are used. For APNR Arabic digits, a study with source code is provided.

At present there are a few Automatic license plate recognition systems in a few countries as mentioned above. Coming to India we have the following number plate recognition systems:

1. EFKON:

   EFKON India's Automatic Number Plate Recognition (ANPR) system is artificial intelligence-based, making it a reliable and easy-to-integrate solution for collecting various sorts of license plates. Our devices, which are placed across India, are capable of recognizing and recording the license plates of over 3,00,000 cars every day, and then sending the information to a central control room for processing. This assists in the enforcement of the law as well as maintaining the protection of the public.

2. KOTAI:

   Kotai's ANPR System is tiny in size and can connect to practically any IP camera. Many factors influence ANPR software, including language font, vehicle speed, illumination, camera angle, and camera position, among others. Also, it can be customized.

**3.1 DRAWBACKS**

- Real Time Monitoring & Surveillance

Because an autonomous system is now available, manned surveillance has become less important. Not only are there fewer traffic officers on the streets and at important intersections, but those that are there do not bother to check licence plates. This might result in a manned security lapse.

- Still and Video Footage

When combined with a lack of manned supervision, poor weather, or any other type of impediment or blockage, automatic number plate recognition systems can be rendered useless. In such instances, security measures may be rendered ineffective.

- Proactive and Reactive Security

The retention and storage of photos and documents for such a lengthy period creates privacy issues. It is possible to abuse the records of a person's location in all the footages. All such data can be misused by stalkers, data thieves, and persons with different malicious motives.

# PROPOSED SYSTEM

We aim to create a system with an algorithm which gives the best image of number plate from the provided image and high accuracy. The user should just give an image of the vehicle they intend to find the license number. The system will then process this image to give an output with the license plate number in text format and audio format. In solving this problem of ours we have used several libraries, algorithms and built a deep learning model. The system can take the image input in any format like .jpg, .jpeg, .png etc.

Our proposed solution is an automatic number plate recognition system which will give out the number plate of the vehicle whose picture is given as an input to the system. The system will take an input of the vehicle's image first. Later it will convert the image into a grey scale image. The system will then crop the number plate and remove all the unnecessary part of the image. Now from the cropped image the system will detect the text present in the image. It will give this text as an output which is the license plate number of the vehicle. It will also give a sounded output of the number.

## 4.1 ADVANTAGES

Automatic number plate identification eliminates the inconvenient and time-consuming task of manually recording the numbers. When a car drives by, just a few individuals can see the number plates and the actual registration digits. It is hard for anyone to keep track of all the numbers when there are several automobiles on the road. Automatic number plate recognition systems not only capture and picture numbers, but also do so in real time.

## ARCHITECTURE

The structure, behaviour, and viewpoints of a system are defined by the system architecture, which is a conceptual model. The architectural design for the Automatic Number Plate Recognition (ANPR) system is shown below. An automated number plate recognition (ANPR) system scans and processes video containing a vehicle number plate as input and recognises the number plate as output.

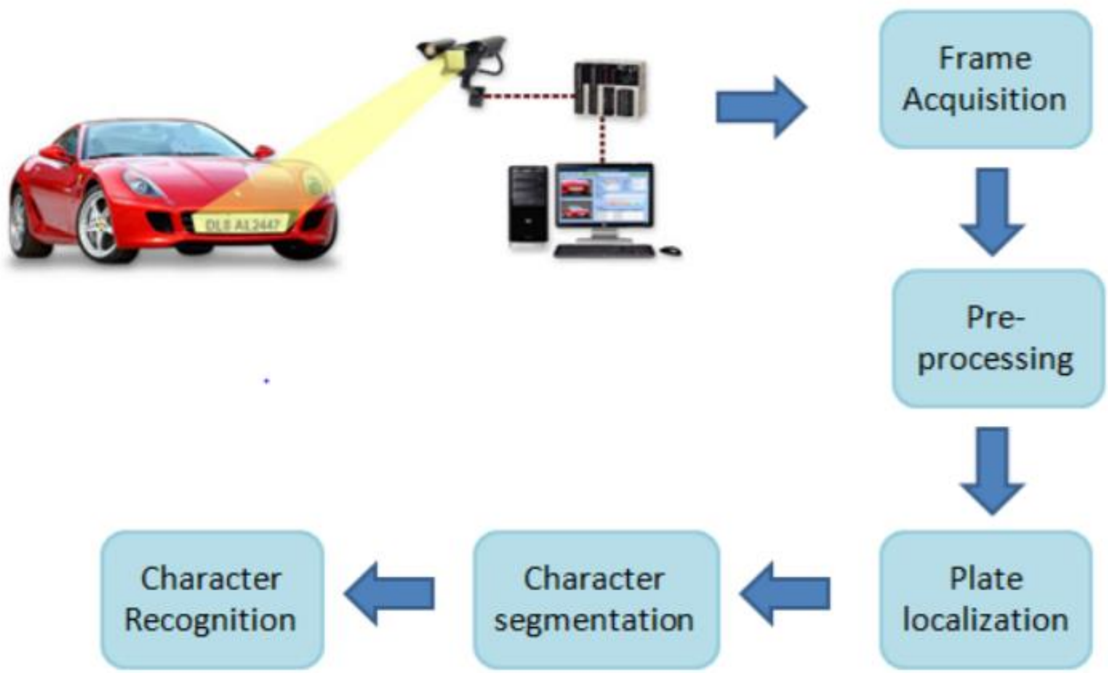


Fig 5.1: Architecture of the system

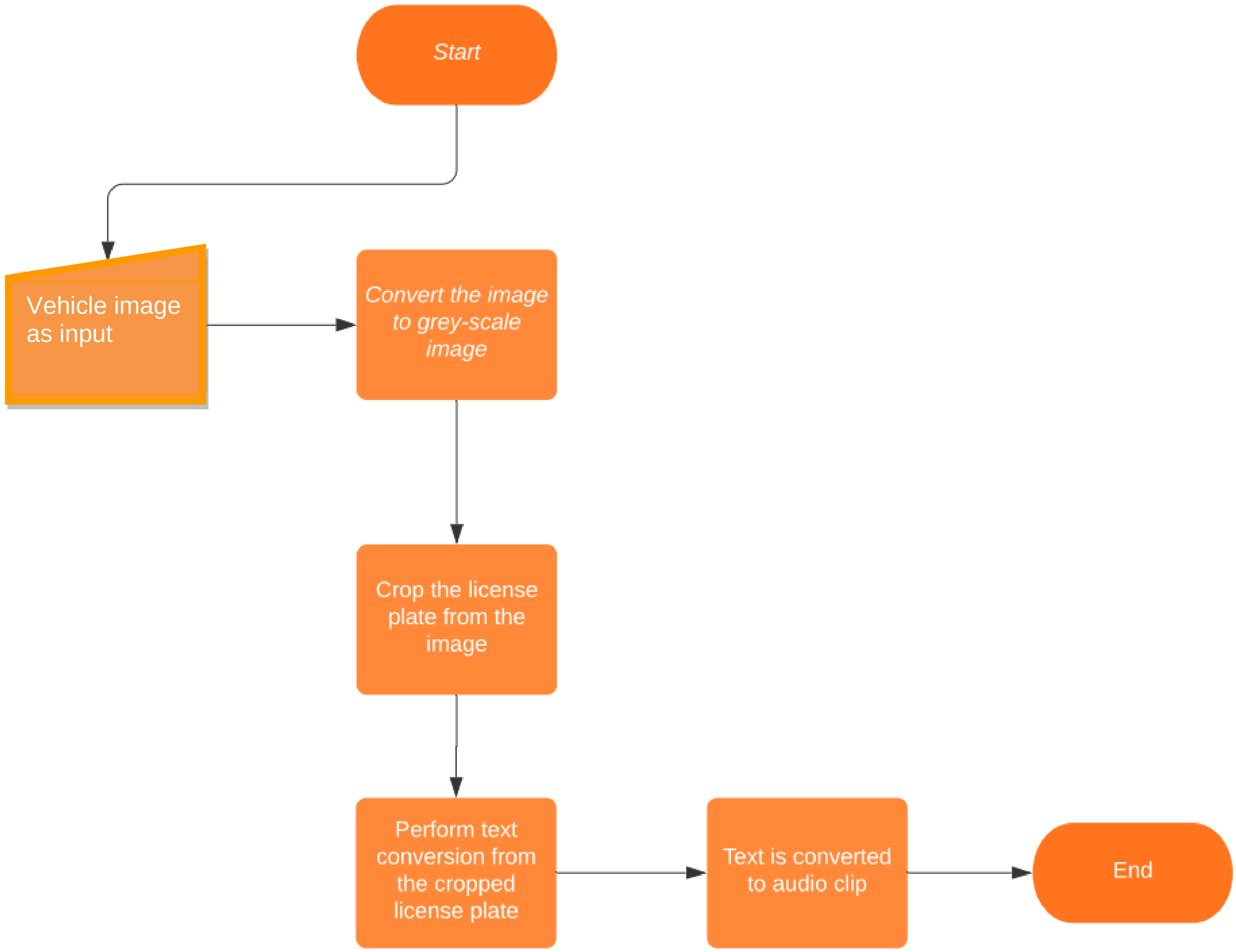


Fig 5.2: Flowchart of system

The system architecture diagram shows the general layout of the software system, as well as the interactions, restrictions, and boundaries that exist between its many components. The user need to input the photo of the vehicle to the system. The system will convert this image to a grey-scale image. After the conversion the image is cropped to the number plate of the vehicle. Once the image is cropped text is recognized from the cropped image. This text is converted to an audio clip too. The text of the number plate and its audio are given as an output to the user by the system.

## MODULES

Our project has a very simple workflow. First, we have collected dataset for our problem statement. We then normalized the data we have collected so far. Then the data was trained using the algorithm. After the training is done, we have exported the trained data into the model. Once the exporting is completed, we deploy the model.

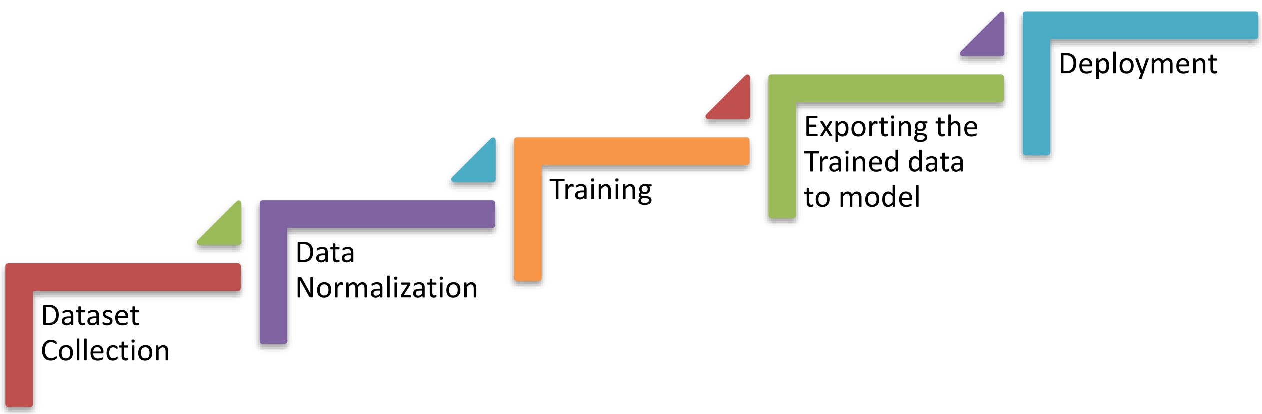


Fig 6.1: Workflow of system

If we get deep into our project workflow, we can divide it into three modules. They are:

**EDGE DETECTION MODULE**

This module consists of three main steps. We have used Canny edge detection algorithm in our project for the edge detection. In this edge detection module first the system is given an image as an input. OpenCV library comes in hep here. OpenCV library can read any image of any format.

After accepting the input this image is converted to a gray scale image. This image will be helpful in detecting the edges of the number plate of the vehicle using the canny edge algorithm. Using this algorithm, the converted image is cropped. Now the cropped image only has the number plate of the vehicle.

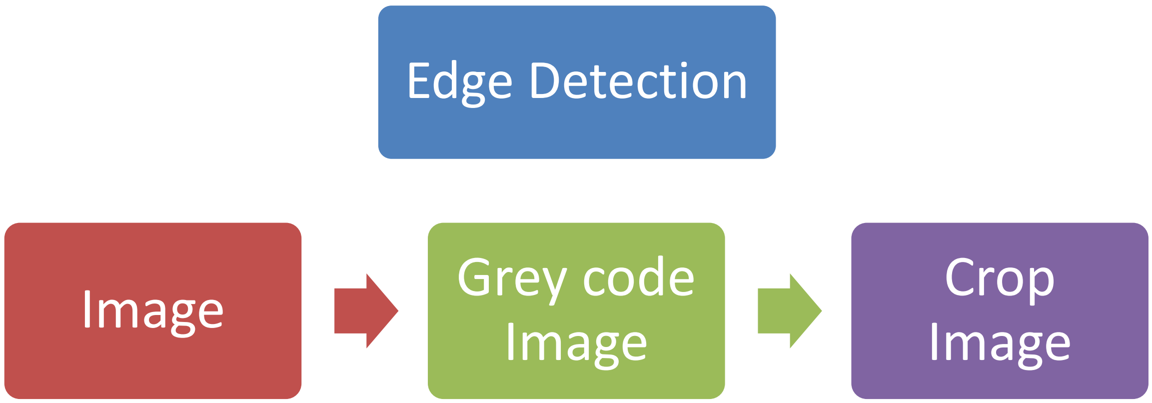


Fig 6.2: Workflow of module_1

**TEXT CONVERSION MODULE**

This module just consists of a single step conversion. From the previous module we receive a cropped image of just the number plate of the vehicle. Here make use of Pytesseract library. Using this library, we convert the text present in the cropped image into text format.

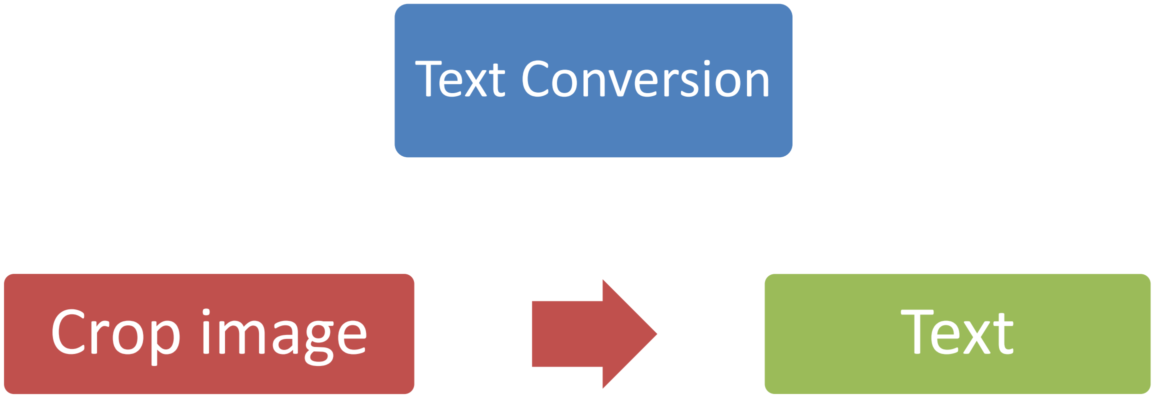


Fig 6.3: Workflow of module_2

**AUDIO CONVERSION MODULE:**

After we get the license plate number of the vehicle in text format our project’s functionality is done. But we have included another feature to help the people who have difficulties in reading text. After the number is detected, it is converted into audio format using gTTS package.

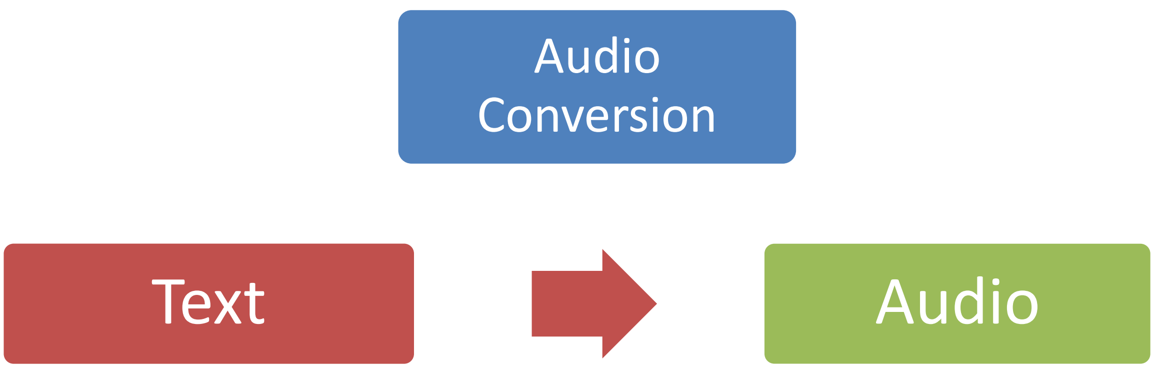


Fig 6.4: Workflow of module_3

# FEASIBILITY ANALYSIS

A feasibility analysis, as the name indicates, is used to establish the viability of a concept, such as verifying that a project is legally, technically, and economically feasible. It informs us if a project is worthwhile—in certain situations, a project may be impossible to complete. There are a variety of reasons for this, including the need for too many resources, which not only stops those resources from completing other duties, but also may cost more than the company would make back by taking on a project that isn't lucrative.

The following are three important factors to consider while doing a feasibility study:

1. Economic Feasibility
2. Technical Feasibility
3. Operational Feasibility

The economic, technical, and operational feasibility of our proposed concept in the actual world will determine its development and deployment. The sections that follow go over each of them in depth, outlining the existing flaws and so demonstrating what percentage of our product is genuinely usable on a daily basis.

## 7.4 ECONOMIC FEASIBILITY

The cost and logistical prognosis for this project are referred to as economic feasibility.

As a result, the developed system is also under budget, which was made possible by the fact that most of the technologies employed are freely available on the internet. The economic analysis looks at data to see if the expense is eventually worthwhile for the user. And also, there aren’t many hardware requirements; the only costly requirement is a camera. The dataset required is also not huge but just a few photographs. So, it is economically feasible.

### 7.2 TECHNICAL FEASIBILITY

As it is the foundation of any project, technical feasibility is one of the most important factors to verify before it begins. Because every system that is constructed has a high technological need. The core of this application may be constructed entirely in Python, which is a free and open-source programming language. It is the most versatile language with a comprehensible grammar. For image processing and accessing data saved on a computer, libraries such as OpenCV and Pytesseract are employed. As a result, the availability of free and open-source technologies makes the project more technically viable.

### 7.3 OPERATIONAL FEASIBILITY

Assessing operational feasibility entails determining whether or not the proposed system will address the User's issues or capitalize on possibilities. It is critical to comprehend how precisely the number plate of the image provided as input is detected. Process, Evaluation, Implementation, and Resistance are all areas where operational feasibility studies are commonly used. Python also allows developers to quickly deploy programs and prototypes, speeding up the development process. If required, a project can be translated to more advanced languages such as Java or C once it is on its way to becoming an analytical tool or application.

# SOFTWARE & HARDWARE REQUIREMENTS

## 8.1 SOFTWARE REQUIREMENTS

We had two essential software requirements in this process. They are: Jupyter Notebook, Python and Pytesseract. We have used Jupyter Notebook to write the code of our recognition system. We have coded this recognition system in Python language. Pytesseract is a wrapper for Tesseract-OCR Engine. We have used it to read the images.

### 8.1.1 Libraries

The process of creating our license plate recognition system required a few libraries. They are: OpenCV, Pytesseract.

- OpenCV: OpenCV is a large open-source library for computer vision, machine learning, and image processing, and it currently plays a critical part in real-time operations, which are critical in today's systems. It may be used to detect items, people, and even human handwriting in photos and movies.
- Pytesseract: Pytesseract is a Tesseract-OCR Engine wrapper. It can read any image formats supported by the Pillow and Leptonica imaging libraries, including jpeg, png, gif, bmp, tiff, and others, making it usable as a standalone tesseract invocation script.

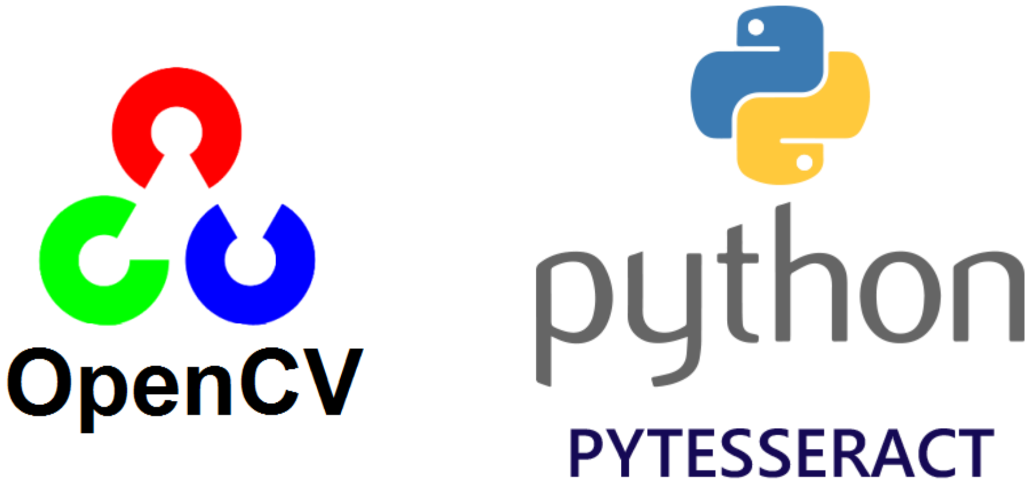


Fig 8.1: Libraries

### 8.1.2 Algorithm

Edge Detection:

In a digital image, edges are large local variations in intensity. A group of linked pixels that creates a border between two discontinuous areas is known as an edge.

Edge Detection is a technique for segmenting a picture into discontinuous parts. It's a common approach in digital image processing, such as:

- pattern recognition
- image morphology
- feature extraction

**Canny Edge is used in our project.**

Canny edge detection is a technique for extracting relevant structural information from various visual objects while reducing the quantity of data to be processed considerably. It's been used in a variety of computer vision systems. Canny discovered that the criteria for using edge detection on a variety of vision systems are quite similar. As a result, an edge detection system that meets these requirements can be used in a variety of scenarios. The following are some general criteria for edge detection:

1. Edge detection has a low error rate, which indicates that the detection should catch as many of the image's edges as feasible.
2. The edge point recognized by the operator should be accurate in locating the edge's center.
3. Image noise should not cause spurious edges, and a specific edge in the image should only be marked once.

Canny utilized the calculus of variations to meet these conditions, which is a technique for determining the function that maximizes a given functional. The optimum function in Canny's detector is defined by the sum of four exponential terms, although the first derivative of a Gaussian may be used to approximate it.

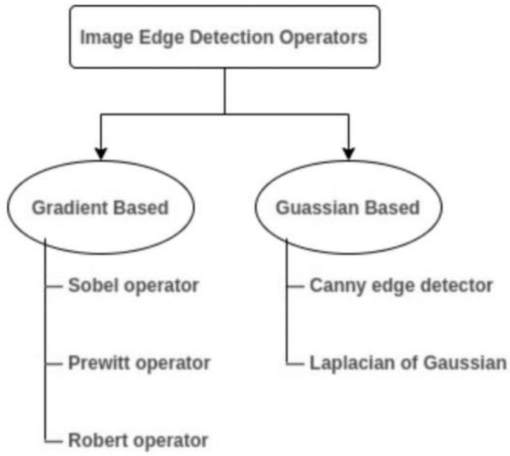


Fig 8.2: Edge Detection Operators

## 8.2 HARDWARE REQUIREMENTS

- Minimum 4 Gigabyte (GB) RAM (used for processing)
- Minimum 16 Megapixel (MP) Resolution camera (for capturing vehicle's image)
- 30 MB Memory space (approximate value)

# DESIGN

## 9.1 STATE DIAGRAM

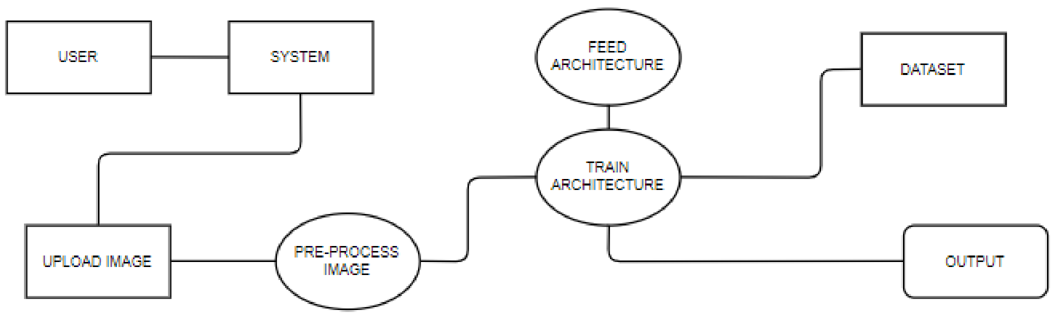


Fig 9.1: State Diagram

## 9.2 USECASE DIAGRAM

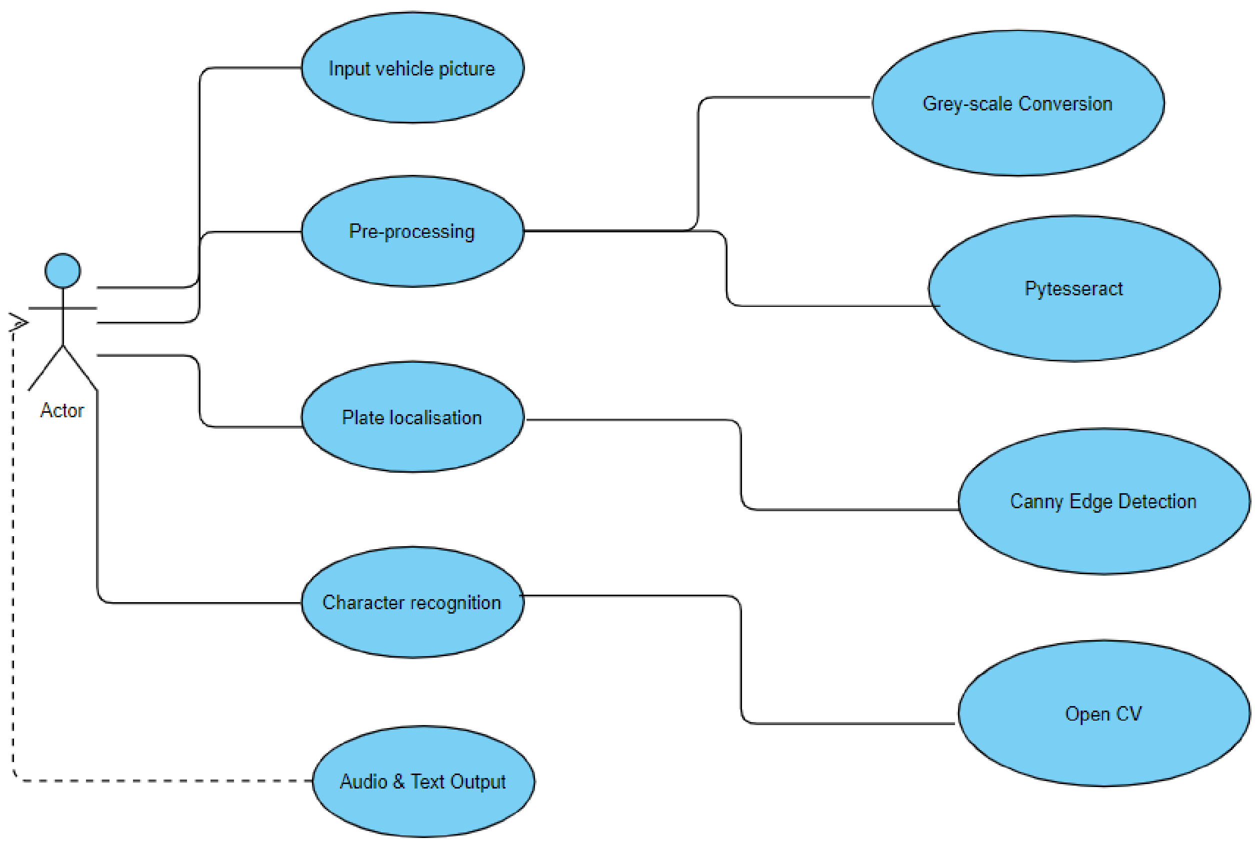


Fig 9.2: Use case Diagram

## 9.3 DEPLOYEMENT DIAGRAM

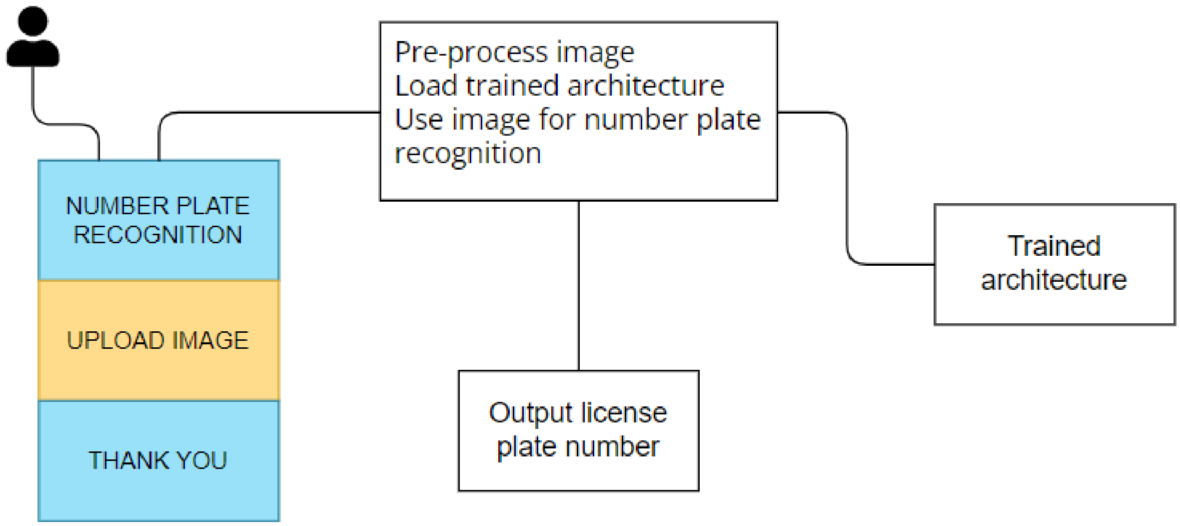


Fig 9.3: Deployment Diagram

## 9.4 ACTIVITY DIAGRAM

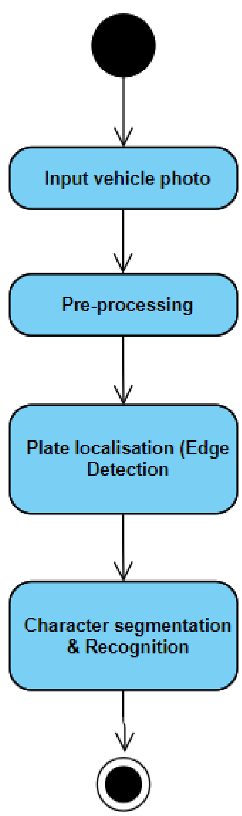


Fig 9.4: Activity Diagram

# REFERENCES

[1]https://www.researchgate.net/publication/349939759_Automatic_number_plate_recognition_using_deep_learning

[2] J. V. Bagade Sukanya, Kamble Kushal, Pardeshi Bhushan, Punjabi Rajpratap Singh Automatic Number Plate Recognition System: Machine Learning Approach IOSR Journal of Computer Engineering pp. 34-39

[3]https://www.irjet.net/archives/V7/i3/IRJET-V7I3878.pdf

[4]https://www.academia.edu/38137741/AUTOMATIC_NUMBER_PLATE_RECOGNITION_USING_CANNY_EDGE_DETECTOR_SLIDING_CONCENTRIC_WINDOWS_AND_LINEAR_DISCRIMINANT_ANALYSIS

[5]https://www.hindawi.com/journals/acisc/2020/8535861/#introduction

## APPENDIX

### A. CODE

```
pip install imutils
pip install gtts
pip install  pytesseract
import sys
import cv2
import pytesseract
import time
import numpy as np
import imutils
from gtts import gTTS
import os
import pybind11
pytesseract.pytesseract.tesseract_cmd = 'C:\\Program Files\\Tesseract-OCR\\tesseract.exe'
# Read the image file
image = cv2.imread('sdfg (1).jpeg')
image = imutils.resize(image, width=500)
# Convert to Grayscale Image
cv2.imshow("Orginal Image",image)
cv2.waitKey(20000)
cv2.destroyAllWindows()
gray=cv2.cvtColor(image, cv2.COLOR_BGR2GRAY)
cv2.imshow("Gray scale image",gray)
cv2.waitKey(20000)
cv2.destroyAllWindows()
gray = cv2.bilateralFilter(gray, 11, 17, 17)
cv2.imshow("Bilateral Filter", gray)
cv2.waitKey(20000)
cv2.destroyAllWindows()
```

```
edged = cv2.Canny(gray, 170, 200)
cv2.imshow("Canny Edges", edged)
cv2.waitKey(20000)
cv2.destroyAllWindows()
(cnts, new) = cv2.findContours(edged.copy(), cv2.RETR_LIST, cv2.CHAIN_APPROX_SIMPLE)
image1=image.copy()
cv2.drawContours(image1, cnts, -1 , (0,255,0),3)
cv2.imshow("canny after contouring",image1)
cv2.waitKey(20000)
cv2.destroyAllWindows()
cnts=sorted(cnts, key = cv2.contourArea, reverse = True)[:30]
NumberPlateCnt=0
image2=image.copy()
cv2.drawContours(image2, cnts, -1 , (0,255,0),3)
cv2.imshow("Top 30 Contours",image2)
cv2.waitKey(20000)
cv2.destroyAllWindows()
count=0
name=1

for i in cnts:
    perimeter = cv2.arcLength(i, True)
    approx = cv2.approxPolyDP(i, 0.02 * perimeter, True)
     # if the approximated contour has four points, then assume that screen is found
    if len(approx) == 4:
        NumberPlateCnt = approx
        x,y,w,h=cv2.boundingRect(i)
        crp_img=image[y:y + h, x:x + w]

        cv2.imwrite(str(name)+ '.jpeg',crp_img)
```

```
        name+=1
        break
cv2.drawContours(image,[NumberPlateCnt],0,(0,255,0),3)
cv2.imshow("Final Image",image)
cv2.waitKey(20000)
cv2.destroyAllWindows()
crop_img_loc='1.jpeg'
cv2.imshow("cropped Image",cv2.imread(crop_img_loc))
cv2.waitKey(20000)
cv2.destroyAllWindows()
text1 = pytesseract.image_to_string(crop_img_loc)
res=' '.join(list(text1))
print("Number is:",res)
text=pytesseract.image_to_data(crop_img_loc, output_type='data.frame')
print(text)
conf=text.groupby(['level'])['conf'].mean()
conf
audio = gTTS(text = res, lang = 'en', slow= False)
audio.save("saved_audio.wav")
os.system("saved_audio.wav")
```

## B. RESULTS

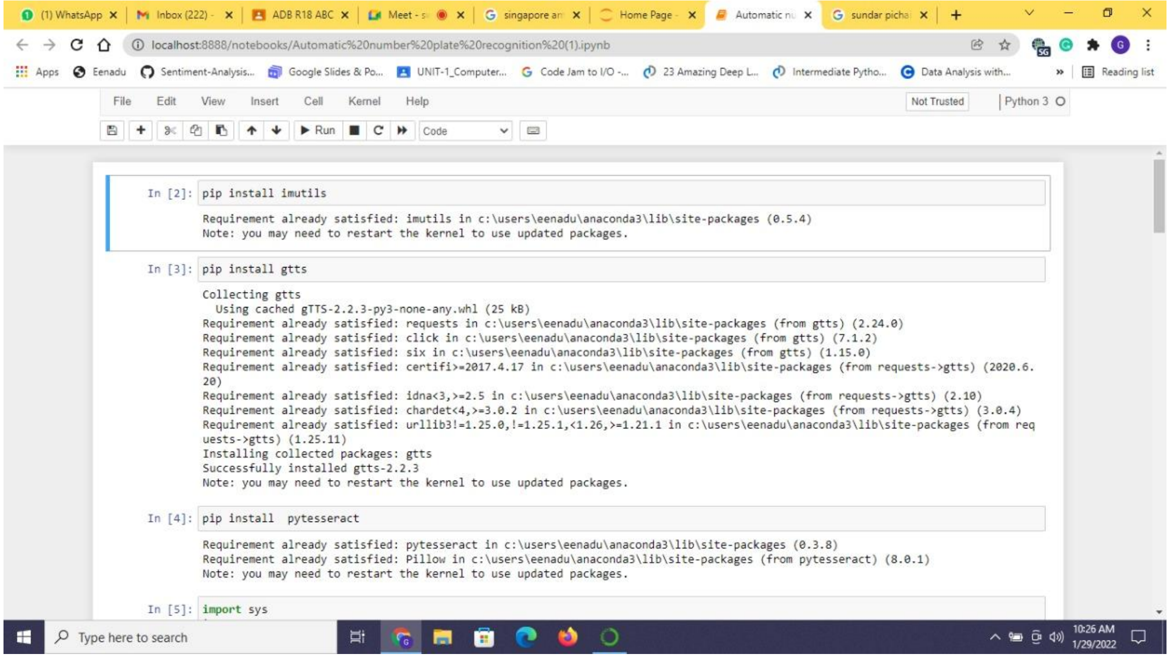

Fig B.1: Output_1

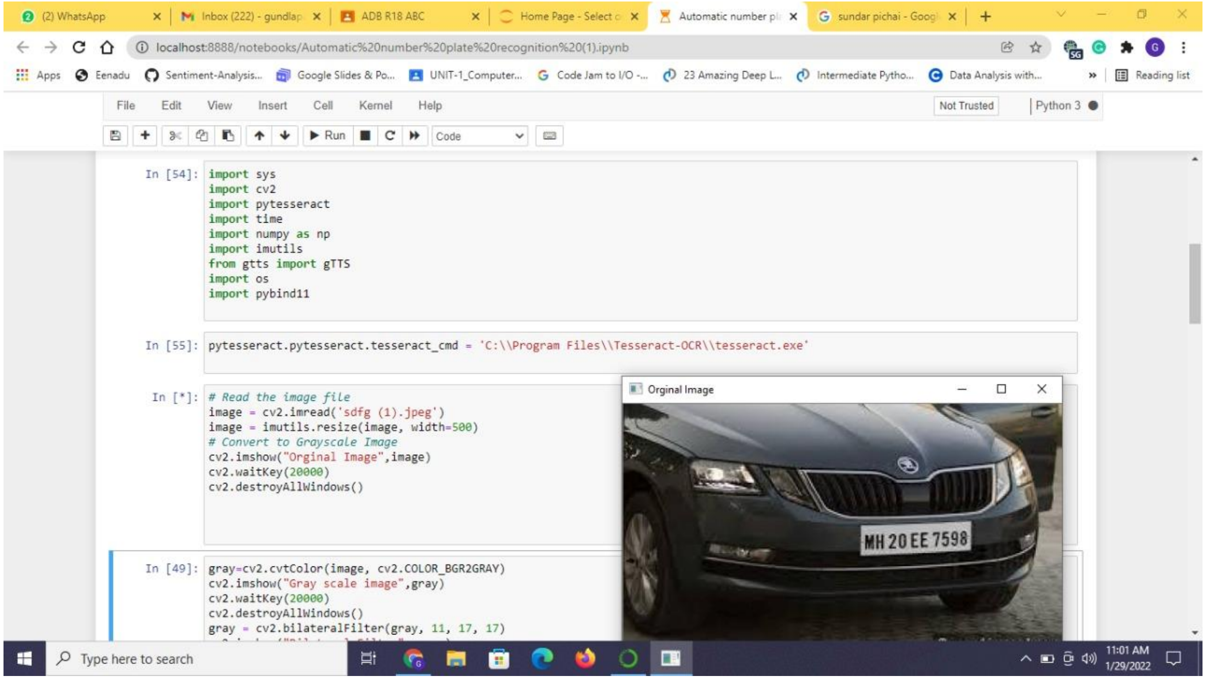

Fig B.2: Output_2

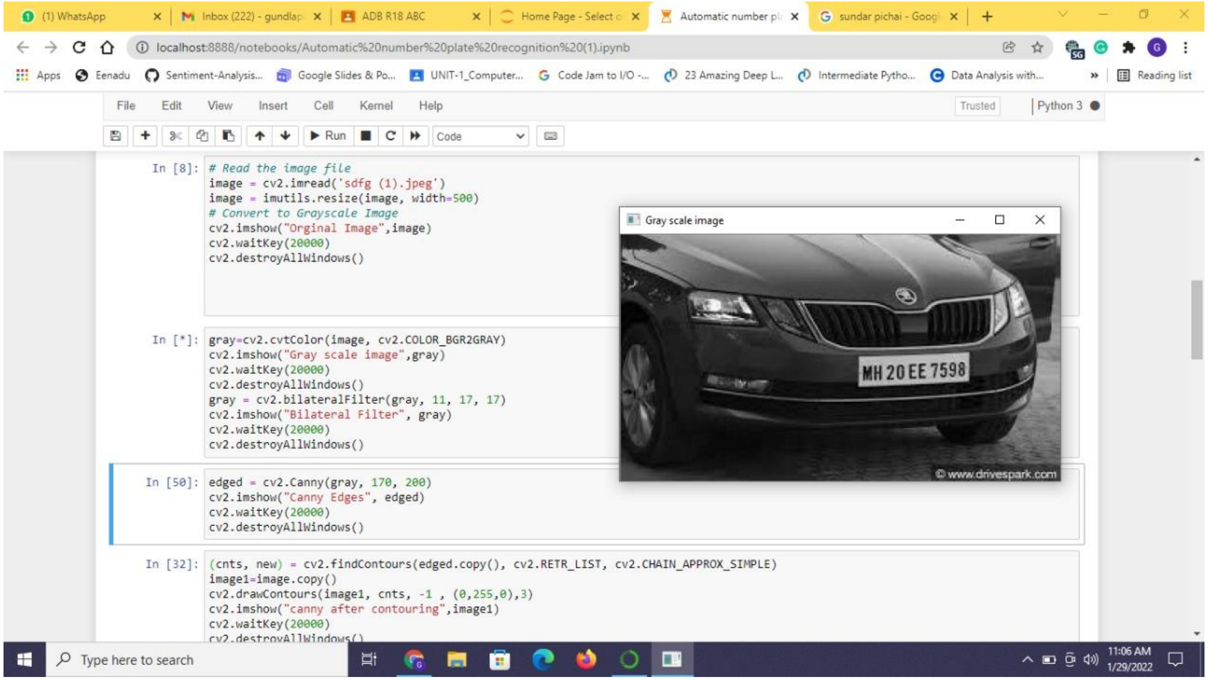


Fig B.3: Output_3

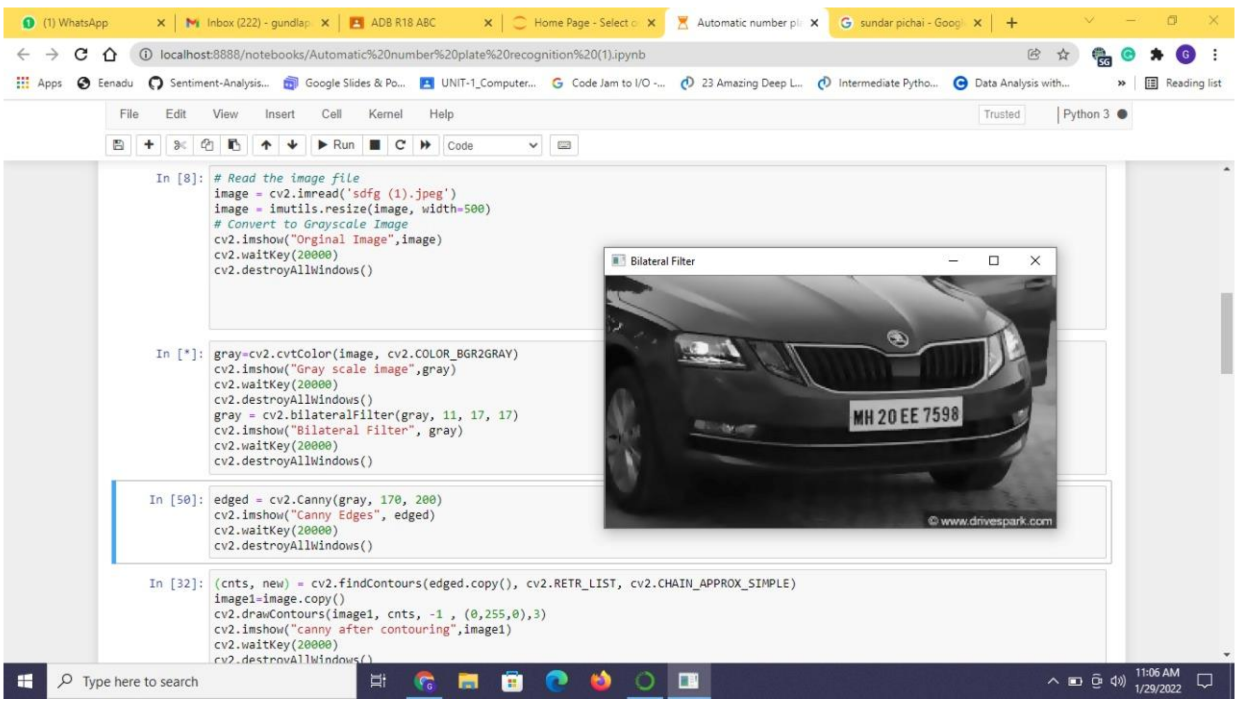


Fig B.4: Output_4

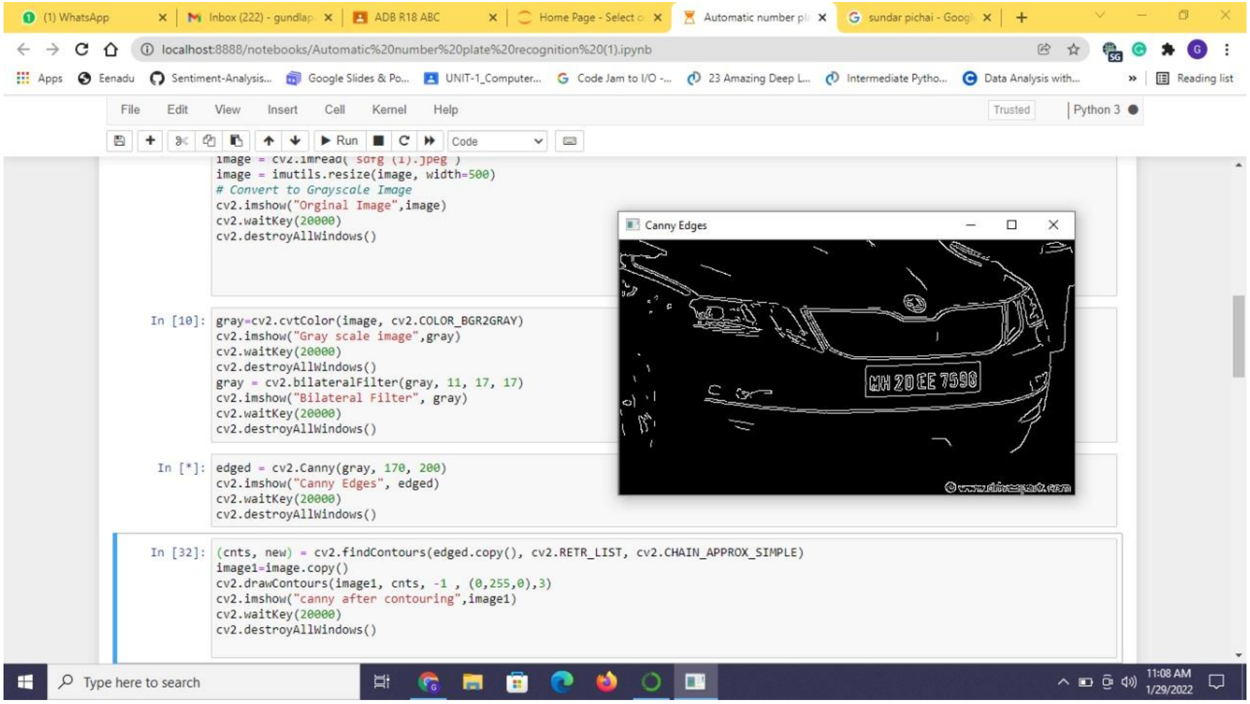


Fig B.5: Output_5

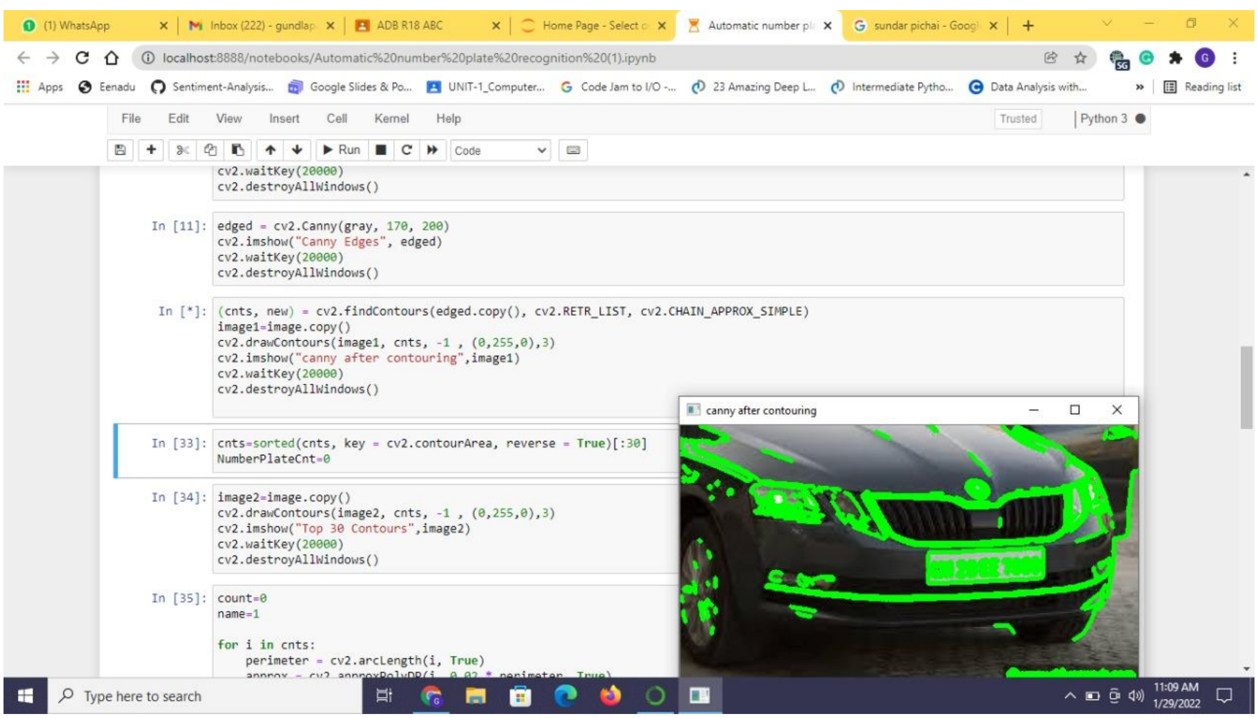


Fig B.6: Output_6

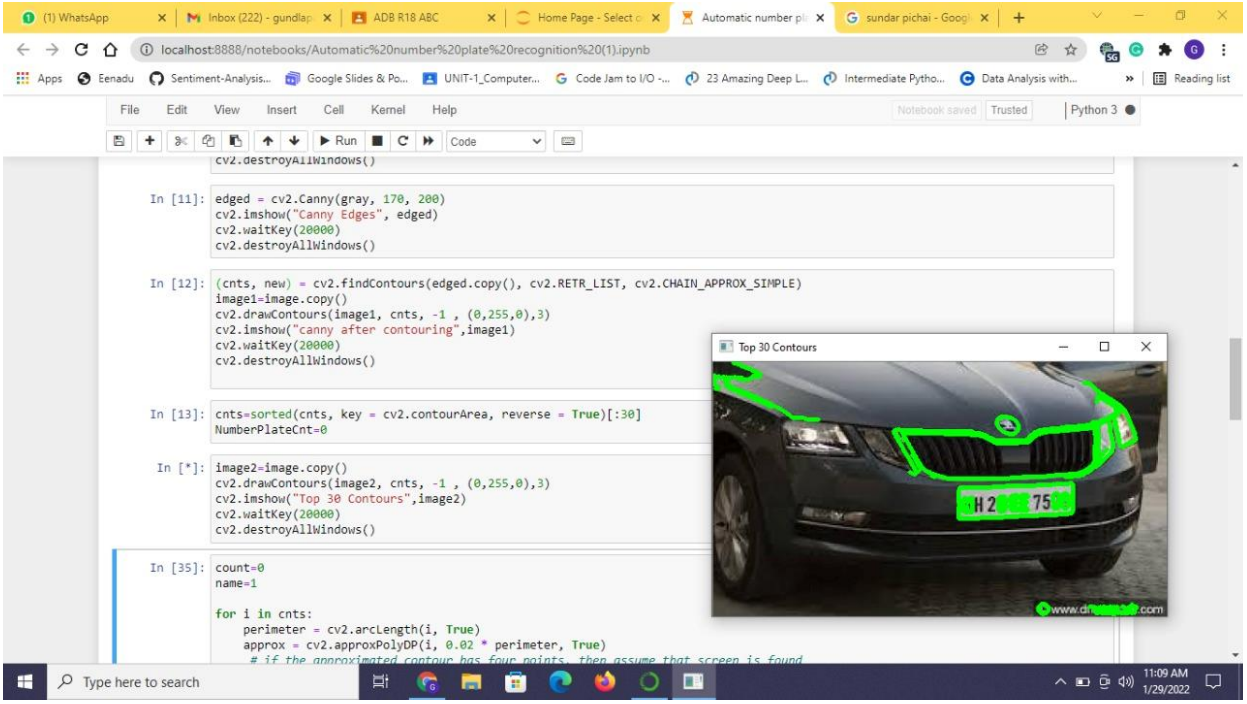

Fig B.7: Output_7

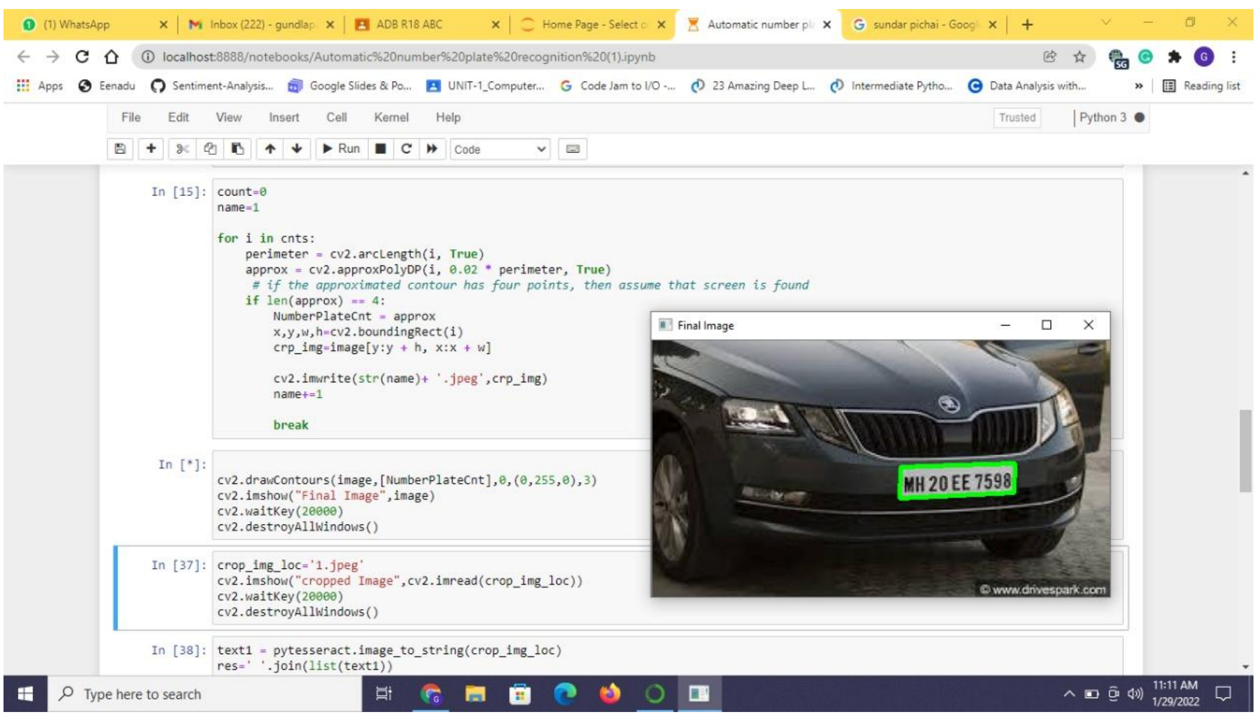

Fig B.8: Output_8

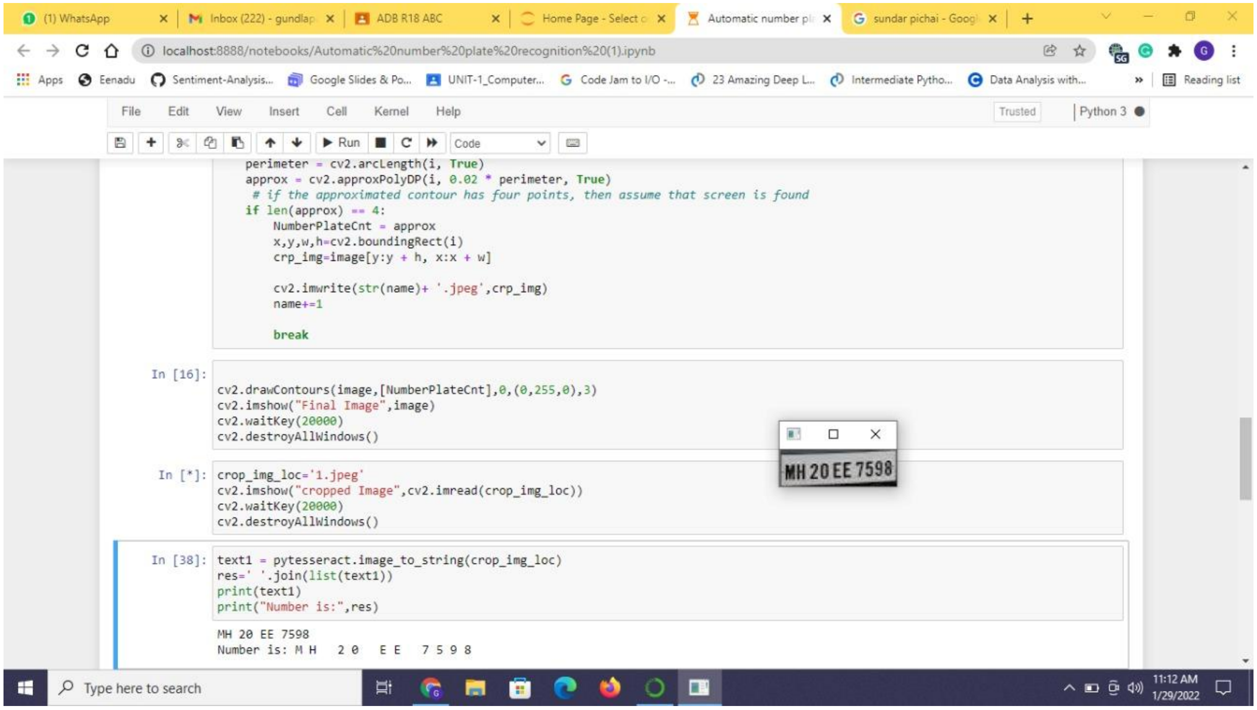


Fig B.9: Output_9

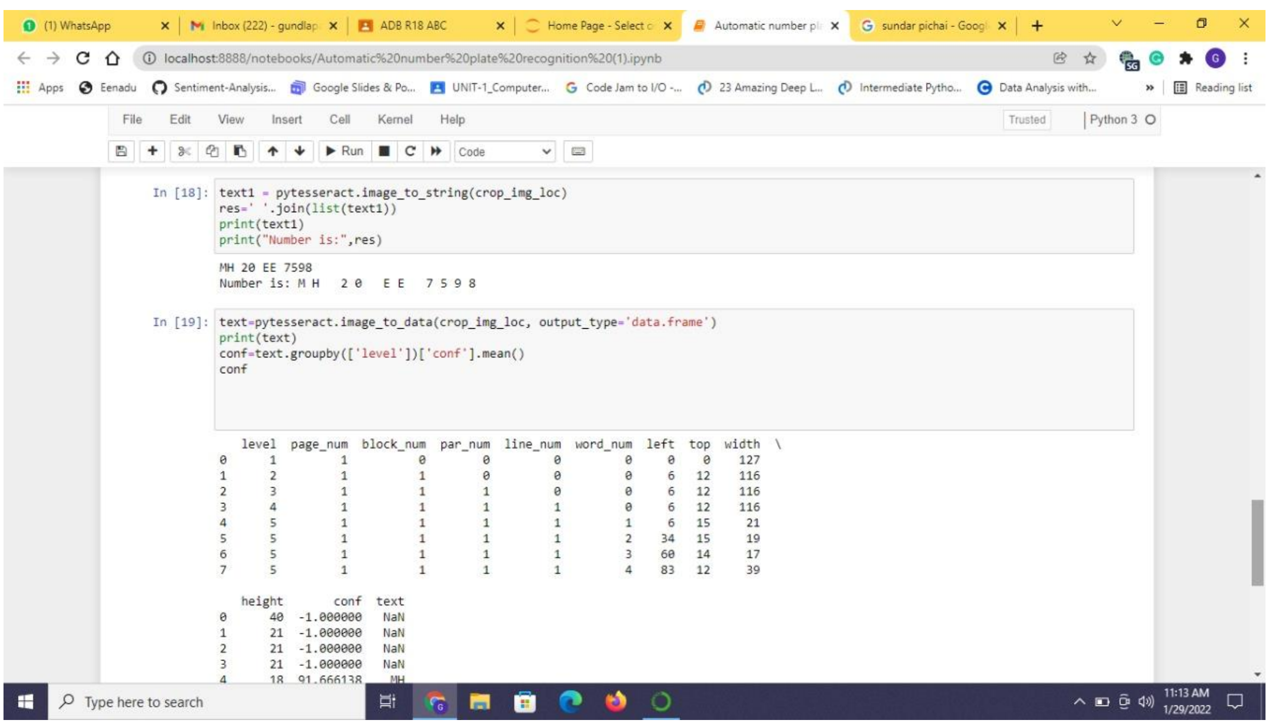


Fig B.10: Output_10

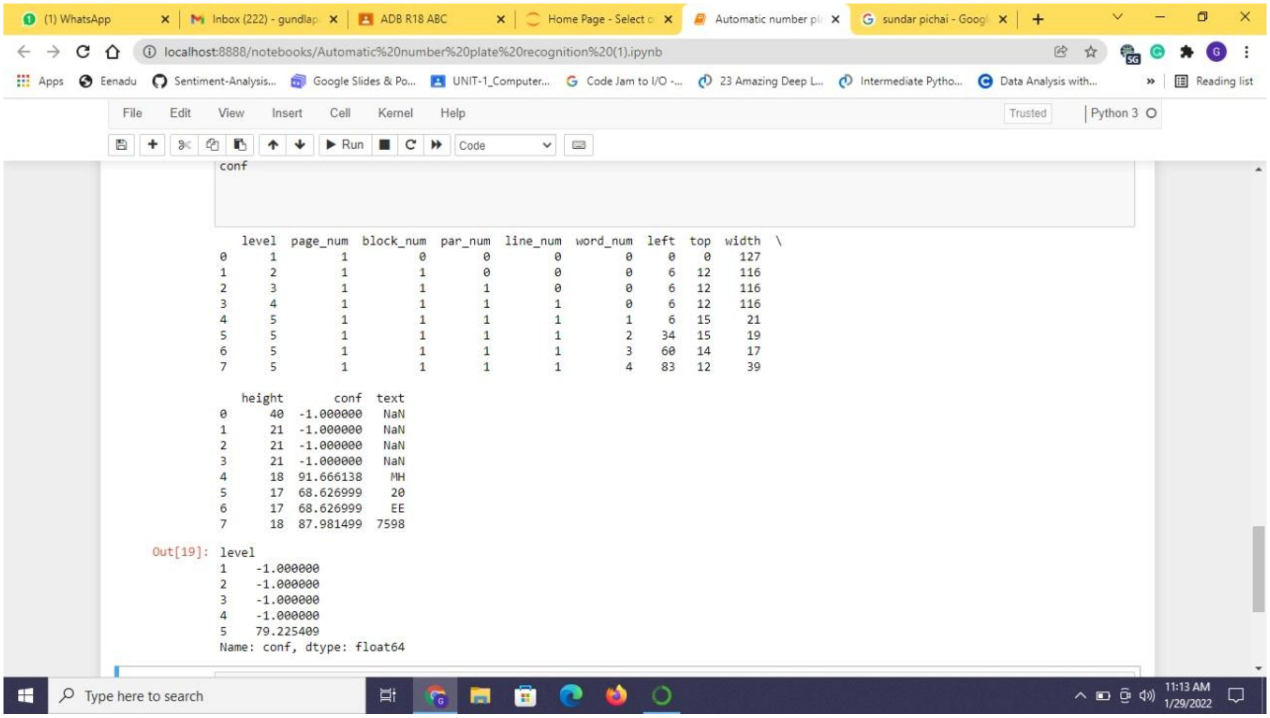

Fig B.11: Output_11

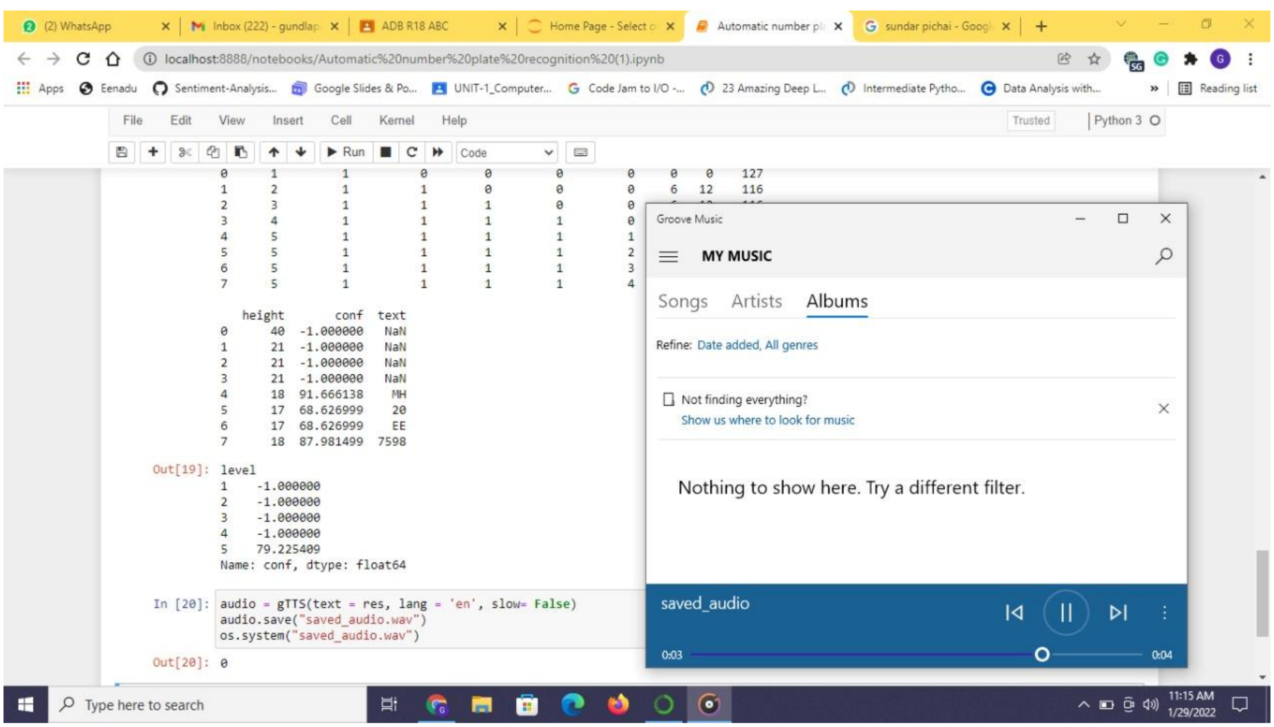

Fig B.12: Output_12