Number Plate Recognition using Python and Rapid API

1. Introduction

1.1 Overview

1.2 Purpose

2. Literature Survey

2.1 Existing Problem

2.2 Proposed Solution

3. Theoretical Analysis

3.1 Block Diagram

3.2 Hardware/Software Requirement

4. Result

5. Advantages and Disadvantages

7. Application

8. Conclusion

1 Introduction:

1.1 Overview

Image recognition, in the context of machine vision, is the ability of software to identify objects, places, people, writing and actions in images. Computers can use machine vision technologies in combination with a camera and artificial intelligence software to achieve image recognition.

Image recognition is used to perform a large number of machine-based visual tasks, such as labeling the content of images with meta-tags, performing image content search and guiding autonomous robots, self-driving cars and accident- avoidance systems.

* LPR (License Plate Recognition) is an image-processing technology used to identify vehicles by their license plates. This technology is used in various security and traffic applications, such as the access-control system.
* It helps to finding the number plates by police and also helps for RTO.

1.2 Purpose

While human and animal brains recognize objects with ease, computers have difficulty with the task. Software for image recognition requires deep learning. Performance is best on convolutional neural net processor as the specific task otherwise requires massive amounts of power for its compute-intensive nature. Image recognition algorithm can function by use of comparative 3D Models, appearances from different angles using edge detection or by components. Image recognition algorithms are often trained on millions of pre-labeled pictures with guided computer learning.

* It reduces the crime rate of the day-to-day life
* It reduces the vehicle theft
* It can easily predict the number plates while the vehicle struck in the accident
* It reduces the insurance and fc fraud in RTO offices, In-case it happens, it is easily identified
* It can provide the large datasets for the number plates and it returns the maximum accuracy ratio in the overall success match
* It provides wide range of countries number plate and their number format of the number plate of each region of the country.

2 Literature Survey

2.1 Existing Problem

* Disadvantages of Existing System
* This was the main disadvantage of the existing system as it is not supported for all kinds of mobiles so later on, we developed a new version.
* Becoming too dependent on technology. By uploading personal photos to the Google Assistant app, it could violate, one’s privacy.

2.2 Proposed Problem

* It matches the nearest matches to the dataset to gain the knowledge about the vehicle number and owner of the vehicle.
* It also used to predict the anonymous number were used to committing crime in our day-to-day life
* In order to avoid the those thinks, we recognize all the details of the vehicle
* Matches only the nearest and exactly equal to corresponding owner of the vehicle
* It shows the percentage of the vehicle in the exact matches

Flow Of Execution:

* First it loads the html template given by the user and it is fully hosted locally in our PC and by flask framework.
* Then, Gives the input in the form of Image URL, which is copied from the any part of the website Like Google Images/Drives or any other sites.
* Then, Click the button to initiate the process.
* And, Rapid API will check and match their dataset to their exact matches
* Then, the image URL is evaluated then it shows the ratio of the accuracy in the bottom of the text box as x%.

Source Code:

App.py:

# -\*- coding: utf-8 -\*-

"""

Created on Fri Jan 29 10:19:28 2021

@author: Sanju Pranava

"""

from flask import Flask, request, render\_template

import requests

app = Flask(\_\_name\_\_)

def check(output):

url = "https://zyanyatech1-license-plate-recognition-v1.p.rapidapi.com/recognize\_url"

querystring = {"image\_url": output, "sourceType": "url"}

payload='''{\r\n \"image\_url\":"'''+output+'''" ,\r\n \"sourceType\": \"url\"\r\r}'''

headers = {

'x-rapidapi-key': "XXXXXXXXXXXXX",#RapidAPI unique key

'x-rapidapi-host': "zyanyatech1-license-plate-recognition-v1.p.rapidapi.com"

}

response = requests.request("POST", url, data=payload, headers=headers, params=querystring)

return response.json()["results"][0]["plate"], response.json()["results"][0]["confidence"]

@app.route('/')

def home():

return render\_template("base.html")

@app.route('/predict', methods = ['POST'])

def predict():

output = request.form['output']

plate, conf = check(output)

return render\_template("base.html", output=plate+" with confidence score : "+str(round(conf))+"%")

if \_\_name\_\_ =='\_\_main\_\_':

app.run()

Testing API:

# -\*- coding: utf-8 -\*-

"""

Created on Fri Jan 29 08:22:15 2021

@author: Sanju Pranava

"""

import requests

url = "https://zyanyatech1-license-plate-recognition-v1.p.rapidapi.com/recognize\_url"

querystring = {"image\_url":"http://eslamoda.com/wp-content/uploads/sites/2/2014/11/america-carro-600x600.jpg"}

headers = {

'x-rapidapi-key': "62c69479c1msh7557e81fc0cc36dp15127djsn652813f9417a",

'x-rapidapi-host': "zyanyatech1-license-plate-recognition-v1.p.rapidapi.com"

}

response = requests.request("POST", url, headers=headers, params=querystring)

print(response.text)

templates/base.html:

<html>

<head>

<title>Automatic Number Plate Recognition</title>

<link rel="preconnect" href="https://fonts.gstatic.com">

<link href="https://fonts.googleapis.com/css2?family=Roboto:ital,wght@1,100&display=swap" rel="stylesheet">

<style>

body{

margin: 0;

padding:0;

/\*box-sizing:border-box;\*/

background: linear-gradient(-70deg,#000000,#001a1a, #004d4d,#001a33,#000d1a,#000000);

margin-left: 10px;

margin-top: 20px;

}

html{

font-size: 10px;

font-family: "Roboto",Arial,sans-serif;

}

/\*section{

width: 100%;

height:100vh;

color:#fff;

/\*background: linear-gradient(-45deg,#23D5AB, #009999,#006666, #006699);\*/

/\*background: linear-gradient(-45deg,#33ffff,#00ffff,#00cccc, #008080,#004d4d);

background-size: 400% 400%;

}\*/

h1{

font-size: 5rem;

text-transform: uppercase;

letter-spacing: 2px;

position: top;

top:2%;

left:50%;

color:#fff;

}

p{

font-size:1.9rem;

/\*color:#b3b3b3;\*/

color:#ffffe6;

font-family: 'Segoe UI', Tahoma, Geneva, Verdana, sans-serif;

}

h3{

font-size: 2.3rem;

text-transform: uppercase;

color:#fff;

}

img{

height:250;

width:300;

}

input[type=text]{

width:40%;

height: 4.7%;

border:2px solid black;

border-radius: 4px;

}

input[type="submit"]{

position:inherit;

text-align:center;

width:350px;

padding:0.1px;

font-size:35px;

color:#1a1a1a;

font-family:poppins;

font-weight: 400;

border:5px solid #00cccc;

text-transform:uppercase;

letter-spacing:2px;

cursor:pointer;

border-radius:100px;

transition:1.5s;

background-color:#66ffff;

}

input[type="submit"]:hover{

box-shadow: 0 1px 0 #15f4ee inset,0 1px 50px 0 #15f4ee,

0 1px 0 #15f4ee inset,0 1px 50px 0 #15f4ee

}

/\*.button{

position: relative;

text-align:center;

width:250px;

padding:40px;

font-size:35px;

color:#15f4ee;

font-family:poppins;

font-weight: 400;

border:5px solid #15f4ee;

text-transform:uppercase;

letter-spacing:15px;

cursor:pointer;

border-radius:100px;

transition:1.5s;

}\*/

</style>

</head>

<body background=" linear-gradient(-45deg,#33ffff,#00ffff,#00cccc, #008080,#004d4d)">

<form action = "http://localhost:5000/predict" method = "post">

<center><h1>Automatic Number Plate Recognition</h1></center>

<br>

<p>LPR(License Plate Recognition) is an image-processing technology used to identify vehicles by

their license plates. This technology is used in various security and traffic applications,

such as the access-control system.

</p>

<h3>Project Description :</h3>

<p>Number Plate Recognition System is an image processing technology which uses number (license)

plate to identify the vehicle. The objective is to build a web application that can efficient

automatic authorized vehicle identification system by using the vehicle number plate.

Number plate recognition (NPR) can be used in various fields such as vehicle tracking,

traffic monitoring, automatic payment of tolls on highways or bridges, surveillance systems,

tolls collection points, and parking management systems.

</p>

<br>

<br>

<center>

<h3> Drop in the image URL to get the Recognised License Plate Number : </h3>

<p><input type = "text" placeholder="URL" name = "output"/></p>

<div class="container">

<div class="button">

<p><input type = "submit" value = "Submit"/></p>

</div>

</div>

</center>

</form>

<center>

<h3>Recognised Number Plate is :</h3>

<h3><b>{{output}}</b></h3>

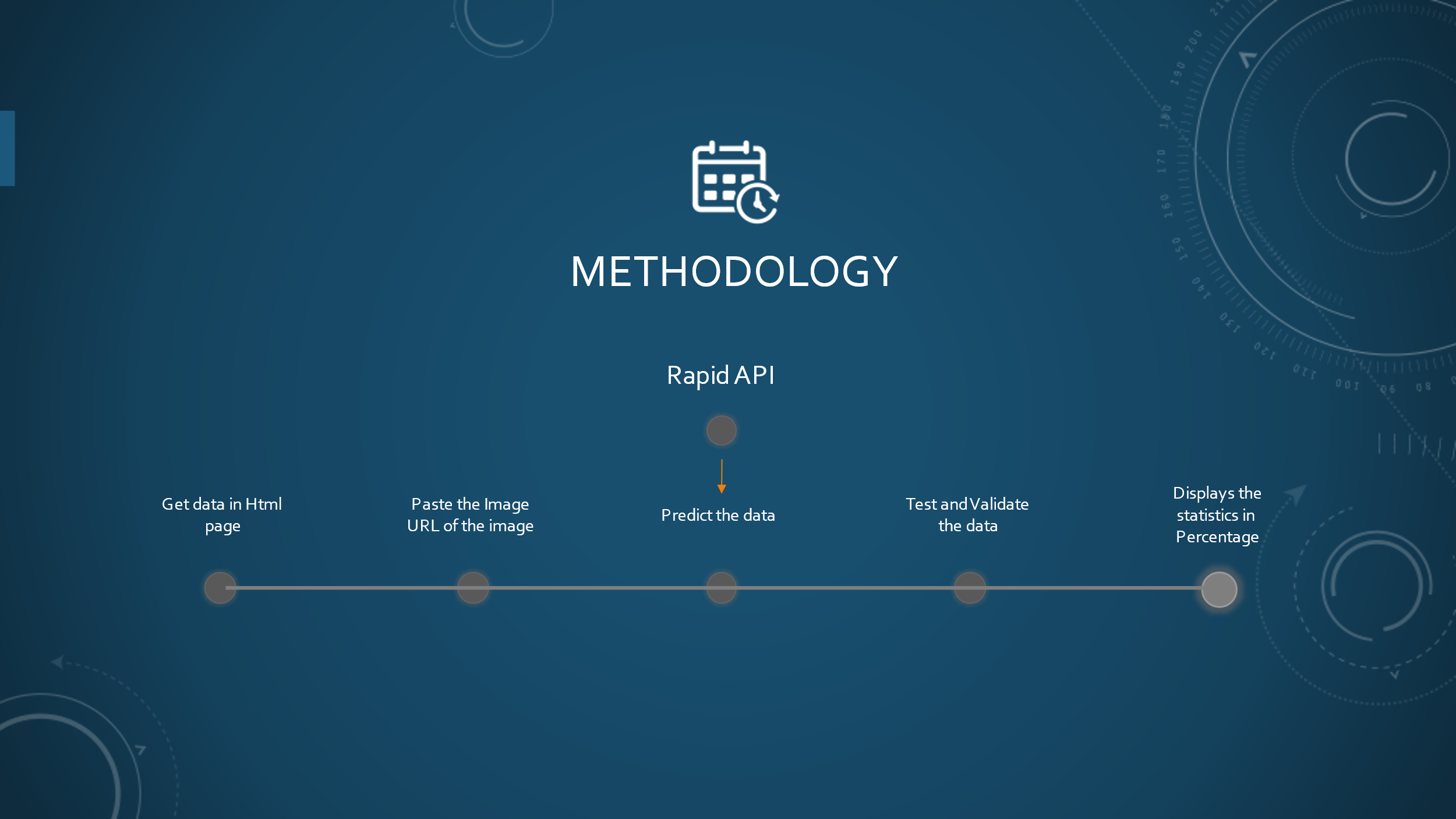
</center>

</body>

</html>

3. Theoretical Analysis

3.1 Block Diagram

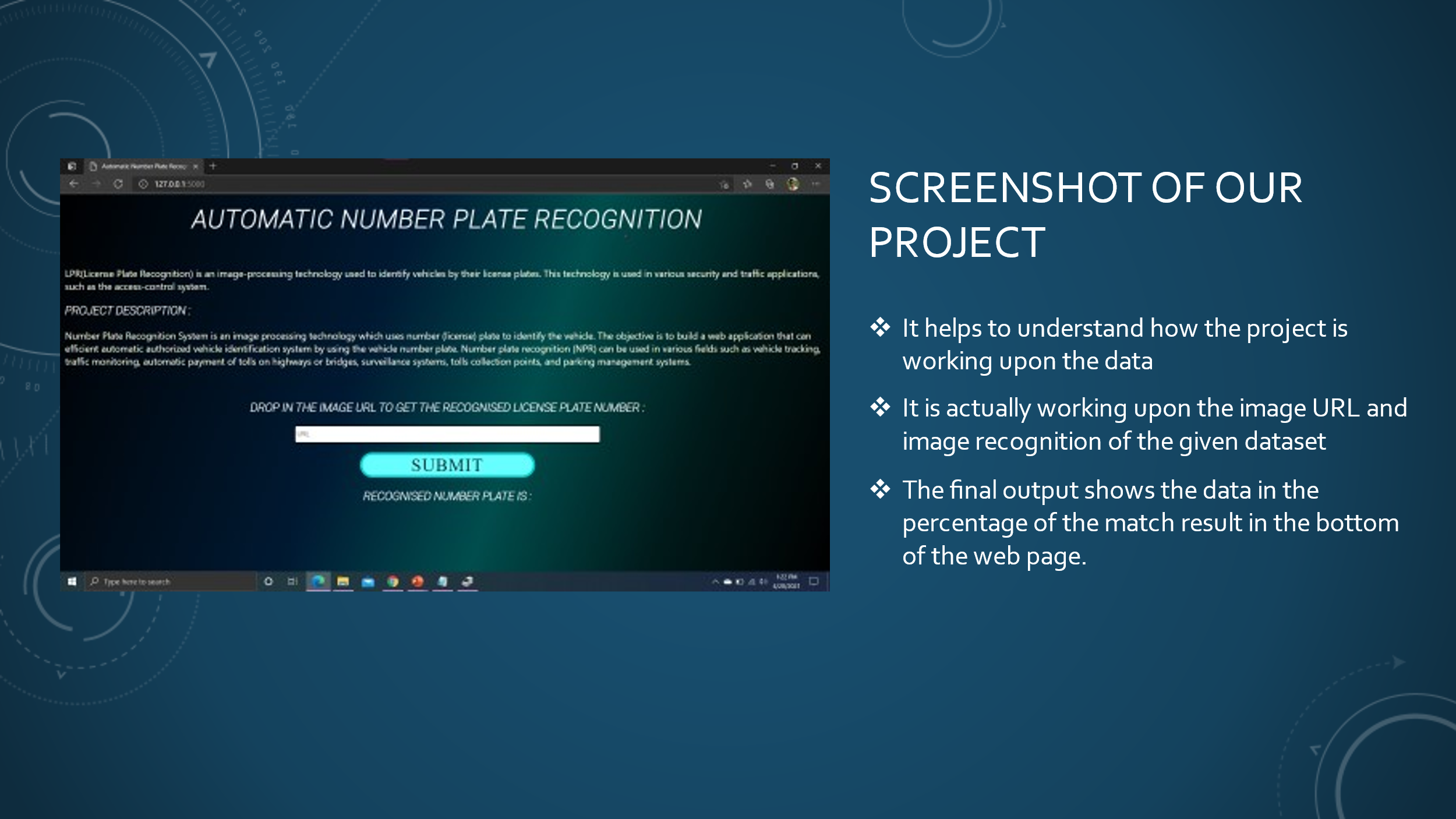


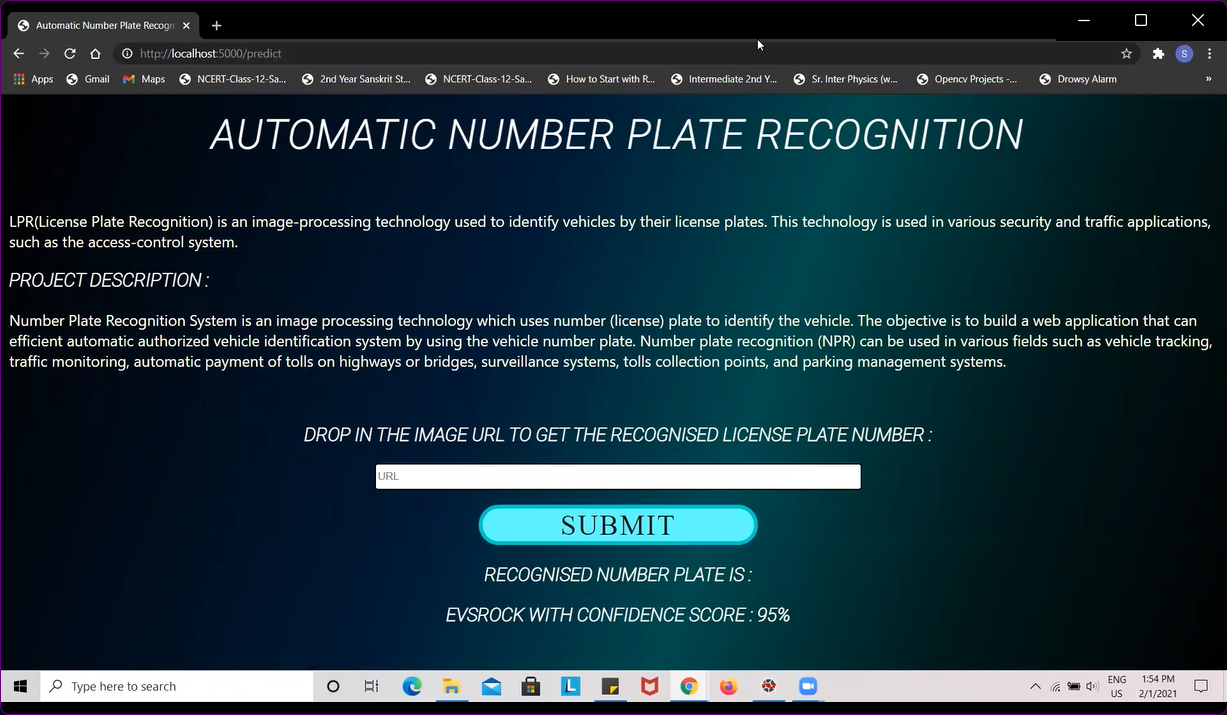
3.2 Hardware/Software Requirement

Anaconda, Visual studio code/Pycharm, flask framework (Python Package), requests, Chrome/Microsoft Edge Browser.

4. Result:

Screenshot for Image recognition





5. Advantages and Disadvantages

Advantages:

* remove noises.
* Correct image density and contrast.
* Helps to easily store and retrieve in computers.
* Image can be made available in any desired formats like black and white, negative image.
* It reduces the crime rate of the day to day life
* It reduces the vehicle theft
* It can easily predict the number plates while the vehicle struck in the accident.
* It reduces the insurance and fc fraud in RTO offices, In-case it happens, it is easily identified.
* It can provide the large datasets for the number plates and it returns the maximum accuracy ratio in the overall success match
* It provides wide range of countries number plate and their number format of the number plate of each region of the country.

Demerits:

* It induces anonymous person or hacker get the information about the vehicle may possible, it needs some deep encryption and decryption on the user data.
* It should give the guidelines and policy to use this system, keep always authenticated person can take this system
* It may be possible in Dos effect on this system, each and every time fix the bugs in this application.
* It should take more cost while maintaining this project.

Applications:

* + Improving Augmented Reality Gaming and Applications.
  + Predicting Consumerism Behavior.

Future Implements:

* In the important ceremony, VIP guest of the car must identify and allowed only if found the exact means
* It should be prohibited to use anonymous vehicle can be encountered and reduce daily life crime rates
* It may be possible in Dos effect on this system, each and every time fix the bugs in this application.

Conclusion

We implemented the project only object only in the static image that ca only recognizes, we can’t implement with the live images that is GIF imaged, I will implement only for theft identification in the CCTV System and IOT for Agriculture to find the defective crops in cultivation, I will soon create and update my project to the global standard and dynamic images and that is benefit for all domains. Then, overall conclusion is that every one should know little bit of AI in daily world.