

PROFESSIONAL TRAINING REPORT
at
Sathyabama Institute of Science and Technology
(Deemed to be University)

Submitted in partial fulfillment of the requirements for the award of
Bachelor of Engineering Degree in Computer Science and Engineering

By

NAME: YUTHIKA P
(Reg.No: 40731134)



DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
SCHOOL OF COMPUTING
SATHYABAMA INSTITUTE OF SCIENCE AND TECHNOLOGY
JEPPIAAR NAGAR, RAJIV GANDHI SALAI,
CHENNAI – 600119, TAMILNADU

APRIL 2023



SATHYABAMA

INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)

Accredited with Grade "A" by NAAC

(Established under Section 3 of UGC Act, 1956)

JEPPIAAR NAGAR, RAJIV GANDHI SALAI, CHENNAI- 600119

www.sathyabama.ac.in



DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

BONAFIDE CERTIFICATE

This is to certify that this Project Report is the bonafide work of **YUTHIKA P (40731134)** who carried out the project entitled "**VEHICLE NUMBER PLATE DETECTION AND RECOGNITION**" under my supervision from Feb 2023 to April 2023.

Internal Guide

Dr.J. Cruz Antony

Head of the Department

Dr.S.Vigneshwari M.E., Ph.D

Submitted for Viva voce Examination held on 17-04-2023

Internal Examiner

External Examiner

DECLARATION

I, **YUTHIKA P** hereby declare that the Project Report entitled as **VEHICLE NUMBER PLATE DETECTION AND RECOGNITION** done by me under the guidance of **Dr. J. CRUZ ANTONY** at **SATHYABAMA INSTITUTE OF SCIENCE AND TECHNOLOGY** is submitted in partial fulfillment of the requirements for the award of Bachelor of Engineering degree in computer Science and Engineering.

DATE: 17-04-2023



PLACE: CHENNAI

SIGNATURE OF THE CANDIDATE

ACKNOWLEDGEMENT

I am pleased to acknowledge my sincere thanks to **Board of Management** of **SATHYABAMA** for their kind encouragement in doing this project and for completing it successfully. I am grateful to them.

I convey my thanks to **Dr. T. Sasikala M.E., Ph.D., Dean**, School of Computing, **Dr. S. Vigneshwari M.E., Ph.D.**, Head of the Department of Computer Science and Engineering for providing me necessary support and details at the right time during the progressive reviews.

I would like to express my sincere and deep sense of gratitude to my Internship Guide **Dr.J.Cruz Antony** for his valuable guidance, suggestions, and constant encouragement that paved way for the successful completion of my project work.

I wish to express my thanks to all Teaching and Non-teaching staff members of the **Department of Computer Science and Engineering** who were helpful in many ways for the completion of the project.

TRAINING CERTIFICATE

TABLE OF CONTENTS

CHAPTER NO	TITLE	PAGE NO
I	ABSTRACT	i
II	LIST OF FIGURES	ii
1	INTRODUCTION	1
1.1	Introduction to Vehicle Number plate detection and recognition	1
1.2	Objectives of Vehicle Number plate detection and recognition	1
2	AIM AND SCOPE OF THE PROJECT	3
2.1	Aim of the Vehicle Number plate detection and recognition	3
2.2	Scope of Vehicle number plate detection and recognition	3
3	EXPERIMENTAL OR MATERIAL AND METHODS; ALGORITHMS USED	4
3.1	Approaches to detect the number plate	4
3.2	Text Extraction from the detected number plate	4
3.3	Project lifecycle	5
3.4	System Requirements	6
3.4.1	Hardware Requirements	6
3.4.2	Arduino UNO	6
3.4.3	Ultrasonic Sensor	7
3.5	Software Requirements	8
3.5.1	Visual Studio Code	8

3.5.2	Web Browser	9
3.5.3	Arduino IDE	9
3.5.4	MongoDB Compass	9
3.6	Programming Languages Required	10
3.6.1	Python	10
3.6.2	Python-Flask	11
3.6.3	HTML	11
3.6.4	CSS	11
3.7	Methods of character Recognition	12
3.7.1	Statistical Classifiers	12
3.7.2	ANN Classifiers	12
3.7.3	Pattern/Template Matching	13
4	RESULTS AND DISCUSSION, PERFORMANCE ANALYSIS	14
4.1	Advantages of the project	14
4.2	Limitations of the project	15
5	SUMMARY AND CONCLUSIONS	16
5.1	Summary	16
5.2	Applications	16
5.3	Conclusions	17
REFERENCES		
APPENDIX		
A.SCREENSHOTS		
B.SOURCE CODE		

LIST OF ABBREVIATIONS

SL NO	ABBREVIATION	EXPANSION
1	OCR	Optical Character Recognition
2	UNO	One
3	IDE	Integrated Development Environment
4	OPEN-CV	Open-Source Computer Vision Library
5	HTML	Hyper Text Markup Language
6	CSS	Cascading Style Sheets
7	JS	JavaScript
8	MYSQL	My Structured Query Language
9	SVM	Support Vector Machine
10	ANN	Artificial Neural Network

ABSTRACT

Vehicle number plate detection and recognition is a computer vision task that involves detecting and recognizing license plates in images or video frames captured by cameras mounted on vehicles or fixed cameras on roads. This technology has several applications such as toll collection, parking management, and law enforcement. The process involves several steps such as object detection, character segmentation, character recognition, and post-processing. The main objectives of vehicle number plate detection and recognition are to accurately detect and recognize license plates, improve efficiency and accuracy of toll collection and parking management, enhance law enforcement, increase safety, and provide real-time data on vehicle movements. With the advancements in deep learning algorithms and computer hardware, the accuracy and efficiency of vehicle number plate detection and recognition systems have significantly improved, making them increasingly popular in various industries.

LIST OF FIGURES

Fig	Name of Figure	PAGE NO
Fig 2.1	Workflow of the project	3
Fig 4.1	Project Lifecycle	4
Fig 5.1	Arduino UNO	6
Fig 5.2	Ultrasonic Sensor	7

CHAPTER -1

INTRODUCTION

1.1 INTRODUCTION TO THE VEHICLE NUMBER PLATE DETECTION AND RECOGNITION:

Vehicle_number plate detection and recognition is a technology that allows automatic identification and extraction of vehicle number plates from digital images or video streams captured by cameras. The primary objective of this technology is to enable efficient and accurate vehicle identification, which has a wide range of applications, including law enforcement, toll collection, parking management, and traffic management.

The process of number plate detection and recognition involves several stages. In the detection stage, a computer vision algorithm is used to identify and extract the number plate region from an image or video frame. This is achieved by analyzing various features of the image, such as color, texture, and shape.

Once the number plate region is detected, the recognition stage begins, where the characters on the number plate are identified and extracted. This is usually done using optical character recognition (OCR) technology, which converts the image of the characters into machine-readable text.

1.2 OBJECTIVES OF THE PROJECT

The main objectives of vehicle number plate detection and recognition are:
Accurately detect and recognize license plates: The primary objective of vehicle number plate detection and recognition is to accurately detect and recognize

license plates in images or video frames captured by cameras mounted on vehicles or fixed cameras on roads. This can be challenging due to variations in lighting, weather conditions, and vehicle speed, among other factors.

Improve efficiency and accuracy of toll collection and parking management: By automating the process of license plate detection and recognition, toll collection and parking management systems can operate more efficiently and accurately. This can reduce wait times for drivers and improve overall traffic flow.

Enhance law enforcement: Vehicle number plate detection and recognition can also be used for law enforcement purposes such as identifying vehicles involved in criminal activity or enforcing traffic laws.

Increase safety: By automating the process of license plate detection and recognition, the need for manual intervention in toll collection and parking management systems can be reduced, potentially increasing safety by reducing the risk of accidents caused by human error.

Provide real-time data: Vehicle number plate detection and recognition systems can provide real-time data on vehicle movements, which can be used for traffic monitoring and management purposes.

CHAPTER-2

AIM AND SCOPE OF THE PROJECT

2.1) AIM OF VEHICLE NUMBER PLATE DETECTION AND RECOGNITION:

The aim of vehicle number plate detection and recognition is to automatically identify and extract the license plate information from vehicles using computer vision and image processing techniques. This process involves the detection of the license plate region in an image or video frame, and then the extraction of the alphanumeric characters from the license plate.

This technology has many practical applications, such as automated toll collection, parking management systems, and law enforcement applications such as speed detection and red light enforcement. Additionally, it can be used for security purposes, such as tracking stolen vehicles or identifying vehicles involved in criminal activity. Overall, the aim of vehicle number plate detection and recognition is to improve efficiency and accuracy in various fields where vehicle information needs to be collected and processed.

2.2) SCOPE OF VEHICLE NUMBER PLATE DETECTION AND RECOGNITION:

Vehicle number plate detection and recognition can be used in traffic management systems to monitor and manage traffic flow, track vehicle movement, and enforce traffic rules such as speed limits and red lights. This technology can be used to manage parking lots by automatically tracking and identifying vehicles as they enter and exit the lot, and determining parking duration and fees. Vehicle number plate detection and recognition can be used by law enforcement agencies to track and identify vehicles involved in criminal activity or to enforce traffic rules such as parking violations or toll violations. This technology can be used in automated toll collection systems to identify and bill vehicles passing through toll booths.

CHAPTER-3

EXPERIMENTAL OR MATERIAL AND METHODS; ALGORITHMS USED

3.1 APPROACHES TO DETECT THE LICENSE PLATE LOCATION FROM THE VEHICLE OBJECT

In order to detect the license plate from vehicle images, several machine learning models like deep learning architectures based upon convolutional neural networks can be utilized. This problem statement of object detection is indeed a fast-paced progressing research topic in modern AI.

This system will detect and recognize vehicle license plates from a user's uploaded file. The system can also detect vehicle number plates in real-time, by just clicking on the device's camera. The front-end involves Html, CSS, and JavaScript and the back-end involves Python. The framework used is Django and the database is MySQL.

3.2 TEXT EXTRACTION FROM THE DETECTED NUMBER PLATE

Once the vehicle license plate is detected, the subsequent step is to infer the text in the license plate and segment the alphanumeric characters from it to know the vehicle identity.

Text extraction is one of the most important processes for the automatic identification of license plates because if it fails, the vehicle recognition will not be correct. In order to achieve this objective, OCR can be utilized to decipher the text and employ natural language processing that recognizes, translates, and interprets the text effectively.

We have implemented OpenCV and Python-tesseract libraries. Python-tesseract is a wrapper for Google's Tesseract-OCR Engine. It is also useful as a stand-alone invocation script to tesseract, as it can read all image types supported.

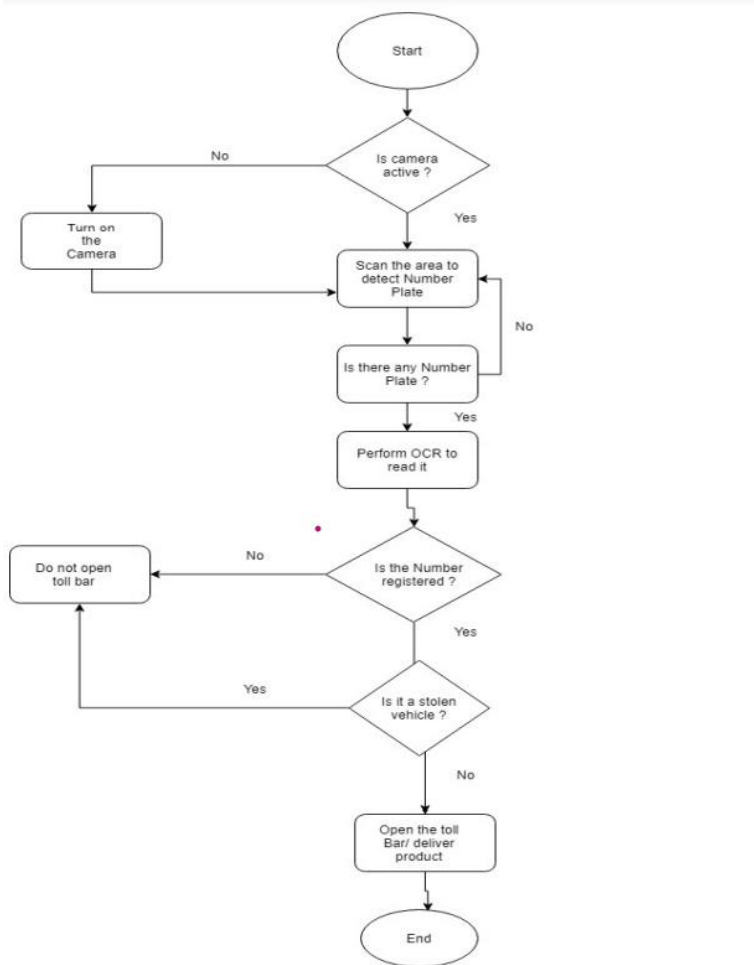


Fig 3.1

3.3PROJECT LIFE CYCLE

The waterfall model is a classical model used in the system development life cycle to create a system with a linear and sequential approach. It is termed a waterfall because the model develops systematically from one phase to another in a downward fashion. The waterfall approach does not define the process to go back to the previous phase to handle changes in requirements. The waterfall approach is the earliest approach that was used for software development.

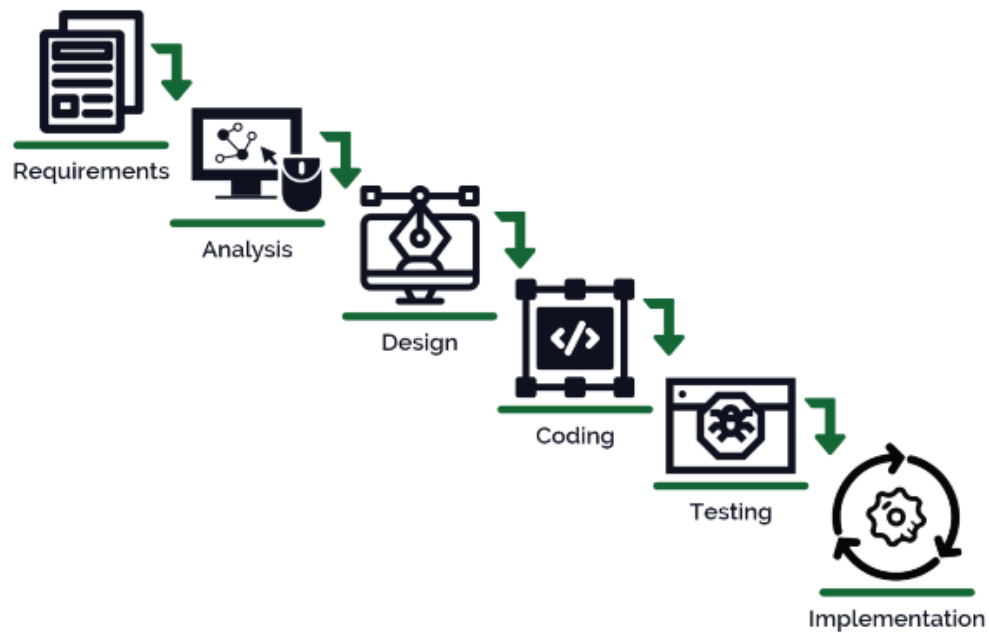


Fig 3.3

3.4 SYSTEM REQUIREMENTS

3.4.1 HARDWARE REQUIREMENTS

3.4.2 Arduino UNO:

Arduino Uno is a microcontroller board based on the ATmega328P microcontroller. It is the most popular board in the Arduino family due to its simplicity and versatility. The board comes with 14 digital input/output pins, 6 analog input pins, and 6 PWM (Pulse Width Modulation) pins.

The board also features a USB interface for programming and power supply, a power jack, an ICSP header, and a reset button. The Arduino Uno can be powered by either the USB interface or an external power supply.

The ATmega328P microcontroller on the Arduino Uno runs at 16 MHz and has 32 KB of flash memory for storing the user's code, 2 KB of SRAM, and 1 KB of EEPROM. The board can be programmed using the Arduino integrated development environment (IDE), which is available for Windows, macOS, and Linux. It is described as a tool, component, or device that notices environmental changes. In the form of a signal, the sensors send such modifications to the electronic devices. Electronic gadgets and sensors constantly function together. The output signal is simple for anyone to read.



Fig 3.4.2

3.4.3 Ultrasonic sensor:

By measuring the time it takes for the sound waves to bounce back, the ultrasonic sensor can calculate the distance between itself and the object. Ultrasonic sensors are commonly used in robotics, automation, and security applications, where they can be used to detect the presence of objects, measure distance, or monitor movement. They

are relatively inexpensive, easy to use, and can be integrated with a variety of microcontrollers and development boards.

Except for very soft materials like wool, which would absorb sound, ultrasonic sensors can detect things regardless of their colour, surface, or composition. Ultrasonic sensors are a viable option for detecting transparent objects and other things where optical methods might not work.



Fig 3.4.3

3.5 SOFTWARE REQUIREMENTS

3.5.1 Visual Studio Code:

Visual Studio Code (VS Code) is a free, open-source code editor developed by Microsoft. It is widely used by developers for editing and debugging code across different programming languages and platforms. Some of the key features of VS Code include syntax highlighting, code completion, debugging support, Git integration, extensions support, and a wide range of customization options. It is also cross-platform, which means it can run on Windows, macOS, and Linux operating systems. VS Code is known for its ease of use, speed, and efficiency, making it a popular choice among developers of all levels.

3.5.2 Web Browser:

A web browser is a software application that allows users to access and view content on the World Wide Web. It works by sending requests to web servers and receiving responses containing HTML, CSS, and JavaScript code that is then rendered by the browser into a user-friendly interface. Browsers allow users to navigate to different websites, view and interact with web pages, download files, and perform other online activities. Some popular examples of web browsers include Google Chrome, Mozilla Firefox, Microsoft Edge, and Apple Safari. In addition to displaying web content, web browsers also offer features such as bookmarking, tabbed browsing, and privacy settings to enhance the user experience.

3.5.3 Arduino IDE:

Arduino IDE (Integrated Development Environment) is a software development tool used to program and develop software for Arduino boards. Arduino boards are microcontroller-based development boards that are popular among hobbyists and makers, as well as professionals working in fields like electronics, robotics, and Internet of Things (IoT) applications. The Arduino IDE provides a simple and user-friendly environment for programming Arduino boards using the Arduino programming language. The IDE includes a code editor, a compiler, a debugger, and a serial monitor. It also provides a library of pre-built code and examples that can be used to jump-start projects. The Arduino programming language is based on C and C++, but has simplified syntax and functions that make it easier to learn for beginners. The IDE includes a text editor with syntax highlighting, auto-completion, and error checking features to help programmers write and debug code more efficiently.

3.5.4 MongoDB Compass:

MongoDB Compass is a visual GUI tool for MongoDB, a popular NoSQL database management system. It provides a user-friendly interface for interacting with MongoDB databases and collections, allowing users to perform various database operations

without needing to write complex code. Users can connect to MongoDB databases running on local or remote servers, making it easy to manage multiple databases from a single interface. MongoDB Compass provides a graphical representation of the data stored in MongoDB collections, allowing users to easily explore and analyze data. Users can view data in various formats, filter data based on specific criteria, and run queries to extract specific information. Users can create and modify documents within collections directly through the Compass interface.

The interface allows users to add, remove, and edit fields, and provides data validation to ensure data integrity. MongoDB Compass includes a performance monitoring feature that provides real-time metrics on database activity, including query performance, resource usage, and server status. Users can create and manage indexes on collections to improve query performance. The interface provides tools for creating and deleting indexes, as well as monitoring index usage and performance.

3.6 PROGRAMMING LANGUAGES REQUIRED

3.6.1 Python:

Python is a high-level, interpreted programming language that is widely used for general-purpose programming. It was first released in 1991 by Guido van Rossum and has since become one of the most popular programming languages in the world. Python emphasizes code readability and simplicity, making it an ideal language for beginners and experienced programmers alike. It supports a wide range of programming paradigms, including procedural, functional, and object-oriented programming.

Python has a large standard library and a vast ecosystem of third-party libraries and frameworks, making it versatile and suitable for a wide range of applications, from web development to scientific computing to data analysis and machine learning. Some of the key features of Python include dynamic typing, automatic memory management,

and support for multiple platforms, including Windows, macOS, and Linux. Python code is typically written in plain text files with a .py extension and can be run on a Python interpreter or compiled into standalone executable files.

3.6.2 Flask:

A web framework is a software architecture that contains tools and libraries used to develop a web application in a fast and efficient way. Flask is a microframework written in Python. It was developed by Armin Ronacher and has a BSD license. It is based on the Werkzeug toolkit and Jinja2 template. Flask is often used for building small to medium-sized web applications, such as blogs, forums, and e-commerce sites. Its simplicity and flexibility make it a popular choice for developers who want to build web applications quickly and easily, without being tied to a particular architecture or framework.

3.6.3 HTML:

HTML, or HyperText Markup Language, is a markup language used for creating web pages and other information that can be displayed in a web browser. It was developed by the World Wide Web Consortium (W3C) and is used as a standard for creating web pages and applications.

HTML consists of a series of elements, which are used to structure content on a web page. Each element is surrounded by tags, which describe the element's purpose and how it should be displayed on the page. For example, the <p> tag is used to create a paragraph, and the tag is used to display an image.

3.6.4 CSS:

CSS, or Cascading Style Sheets, is a style sheet language used to describe the presentation of web pages written in HTML and XML. It allows developers to separate the presentation of a web page from its structure and content, making it easier to make changes to the appearance of a page without affecting its underlying structure.

CSS works by selecting HTML elements and applying styles to them, such as color, font, spacing, and layout. This is done by creating rules that define how a particular element should be displayed, based on its class, ID, or tag. CSS can also be used to create animations, transitions, and other visual effects on a web page.

CSS is an essential tool for web developers and designers, as it allows them to create attractive and functional web pages that are easy to maintain and update. It is supported by all major web browsers and is constantly evolving with new features and updates to the standard.

the characters, distance of left and rightmost characters from 258 vertical edge of plate and distance of characters from horizontal edge of the plate

3.7 METHODS OF CHARACTER RECOGNITION

Character recognition fully relies on isolated characters. Inefficiency of character segmentation influences the success of character to be recognized. For this work, various techniques have been proposed which are based on statistical classifiers, computational intelligence architecture and pattern matching.

3.7.1 STATISTICAL CLASSIFIERS

To eliminate the inefficiency of OCR and pattern-matching techniques statistical classifiers employ Support Vector Machine for recognition. Authors in [15] employ SVM to recognize numbers. The algorithm starts from a collection of samples of numbers from number plates. Each character is recognized by an SVM, which is trained by some known samples in advance. In order to recognize a number plate correctly, all numbers are tested one by one using the trained model. The recognition results are achieved by finding the maximum value between the outputs of SVMs.

3.7.2 ANN CLASSIFIERS

A two-layer Markov network is proposed to integrate these two tasks into one statistical framework. Both low level cues and high-level prior knowledge are incorporated in terms of probabilities. An efficient non-iterative belief propagation algorithm is used for state estimation. Recognition algorithm uses both techniques of template matching and NN (Perceptron). Compared with these two techniques, NN have provided better matching for recognition s compared to template matching (91% against 86%).

3.7.3 PATTERN/TEMPLATE MATCHING

The pattern matching technique is suitable for the recognition of shape invariant, non-rotated characters. P. Comelli et al., cross co-relation to find out the matching between normalized characters and templates. This algorithm is implemented on Italian cars passing through tollgate. Templates are designed for two-letter province, single prototypes for character string. Algorithm tested more than three thousand real images acquired under different weather and illumination condition obtaining a recognition rate near to 91%. Authors developed an OCR technique, for recognition of Thai car license plates, which are generally shape complicated, using the essential elements-based method. This algorithm recognizes the character patterns relying on the essential elements of characters. Using this method, majority of characters are reduced to a set of essential strokes. Matching of characters is based on edge Hausdorff Distance (HD), also called max-min distance. It is a measure of similarity between two arbitrary point sets.

CHAPTER – 4

RESULTS AND DISCUSSION, PERFORMANCE ANALYSIS

4.1 ADVANTAGES OF THE PROJECT

- **Improved Traffic Management:** Vehicle number plate detection and recognition can help in improving traffic management by enabling automated monitoring of traffic flow, detecting traffic violations, and issuing penalties in real-time.
- **Enhanced Security:** Vehicle number plate recognition can assist in enhancing security measures by identifying suspect vehicles, detecting stolen vehicles, and tracking the movement of vehicles used in criminal activities.
- **Faster Toll Collection:** Toll collection can be made faster and more efficient by using vehicle number plate recognition technology, which enables automatic toll payment and reduces traffic congestion.
- **Improved Parking Management:** Parking management can be made more efficient and streamlined by using vehicle number plate recognition technology to monitor parking lots, detect unauthorized vehicles, and issue penalties for parking violations.
- **Improved Public Safety:** Vehicle number plate detection and recognition can help in improving public safety by identifying vehicles that are involved in accidents, tracking hit-and-run drivers, and identifying vehicles that are involved in terrorist activities.

- **Reduced Manual Labor:** Vehicle number plate recognition eliminates the need for manual labor in tasks such as traffic monitoring, toll collection, and parking management, thus reducing costs and increasing efficiency..
- It can detect vehicle license plates from an image and videos.

4.2 LIMITATIONS OF THE PROJECT

- **Limited Accuracy:** Vehicle number plate recognition technology is not 100% accurate and may produce errors in identifying the correct number plate, especially in cases of damaged, obscured or dirty plates.
- **High Cost:** The installation and maintenance of the infrastructure needed for vehicle number plate recognition technology can be expensive, especially for large-scale projects.
- **Limited Range:** Vehicle number plate recognition technology may have limited range, which may require more infrastructure to be installed, increasing the cost of the system.
- **Limited Applicability:** Vehicle number plate recognition technology may not be applicable in all areas, such as in areas with low traffic volume or in rural areas where vehicle number plates may not be standardized.
- **Privacy Concerns:** The use of vehicle number plate recognition technology may raise privacy concerns as it involves the collection and storage of personal data, including vehicle location, and can be used for surveillance purposes.
- **Vulnerability to Hacking:** Vehicle number plate recognition systems are vulnerable to hacking and can be used for malicious purposes, such as stealing personal data or manipulating traffic data.

CHAPTER – 5

SUMMARY AND CONCLUSIONS

5.1 SUMMARY

Vehicle number plate detection and recognition is a technology that uses computer vision and image processing techniques to automatically identify and extract license plate information from vehicles. The aim of this technology is to improve efficiency and accuracy in various fields such as traffic management, parking management, law enforcement, toll collection, border control, and commercial applications. The scope of vehicle number plate detection and recognition is quite vast and has many practical applications.

5.2 APPLICATION

- This application detects and recognizes the vehicle number plate automatically by just uploading an image or video
Traffic Management: Vehicle number plate detection and recognition can be used to monitor traffic flow, detect traffic violations, and issue penalties in real-time. This can help to improve traffic management and reduce traffic congestion.
- Parking Management: Vehicle number plate recognition technology can be used to monitor parking lots, detect unauthorized vehicles, and issue penalties for parking violations.
- Toll Collection: Vehicle number plate recognition technology can be used to automate toll collection, reducing traffic congestion and improving efficiency.
- Law Enforcement: Vehicle number plate recognition technology can assist in law enforcement by identifying suspect vehicles, detecting stolen vehicles, and tracking the movement of vehicles used in criminal activities.

- **Border Control:** Vehicle number plate recognition technology can be used for border control by identifying vehicles that are entering or exiting the country, and checking them against a database of known vehicles.
- **Public Safety:** Vehicle number plate recognition technology can help in improving public safety by identifying vehicles that are involved in accidents, tracking hit-and-run drivers, and identifying vehicles that are involved in terrorist activities.
- **Parking Space Allocation:** Vehicle number plate recognition technology can be used to allocate parking spaces to authorized vehicles, reducing parking congestion in busy areas.

5.3 CONCLUSION

In conclusion, vehicle number plate detection and recognition technology has revolutionized the way we monitor and manage traffic. With the help of advanced algorithms and machine learning techniques, it is now possible to accurately detect and recognize license plates from various angles, lighting conditions, and even under low-resolution images. This technology has numerous applications in law enforcement, toll collection, parking management, and other areas where accurate identification of vehicles is essential. Moreover, with the increasing adoption of autonomous vehicles, license plate recognition technology will become even more critical to ensure safe and efficient transportation. However, there are also concerns about privacy and security with this technology. It is essential to ensure that the data collected through license plate recognition is used ethically and that the information is stored securely.

Overall, vehicle number plate detection and recognition technology has come a long way and will continue to evolve and improve in the future, providing us with better ways to manage and optimize our transportation systems.

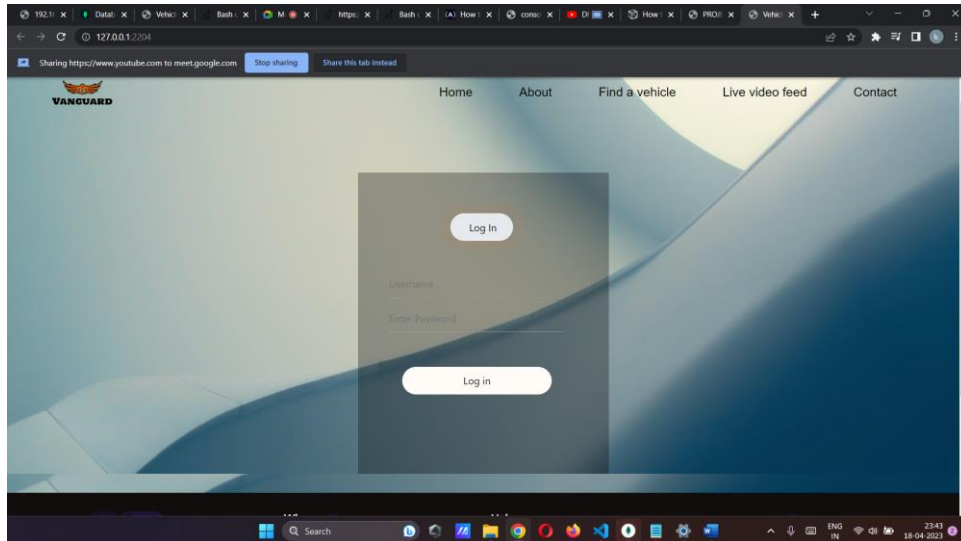
REFERENCES

- <https://www.researchgate.net/publication/334424122> *Research and Application of License Plate Recognition Technology Based on Deep Learning*
- <https://www.researchgate.net/publication/236888959> *Automatic Number Plate Recognition System ANPR A Survey*
- <https://mospace.umsystem.edu/xmlui/bitstream/handle/10355/43016/research.pdf>

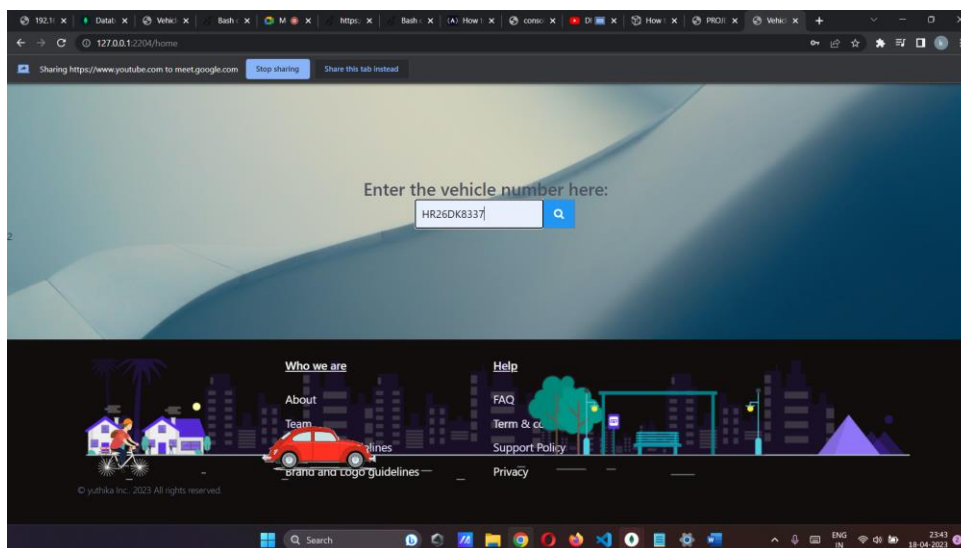
APPENDIX

A. SCREENSHOTS:

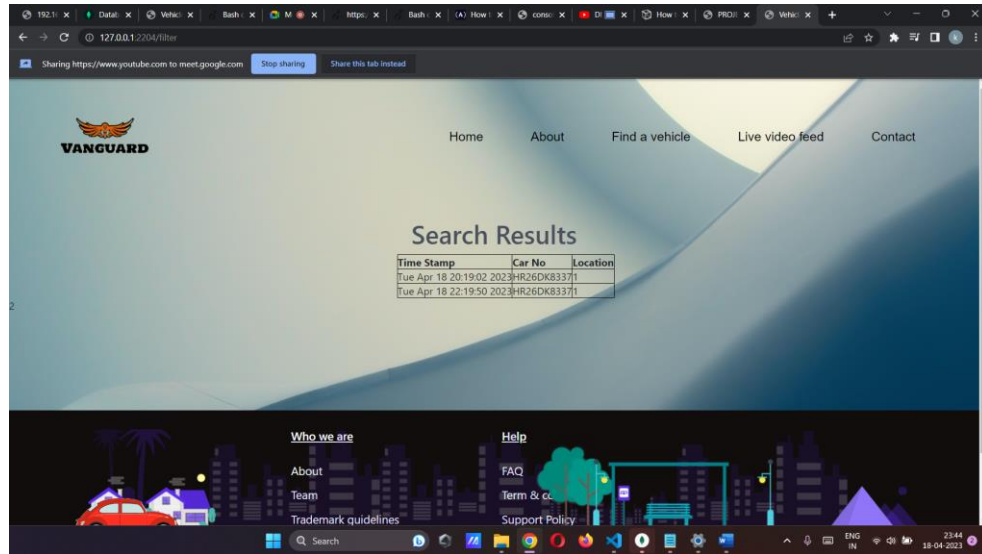
Homepage:



Find vehicle page:



Result of vehicle search:

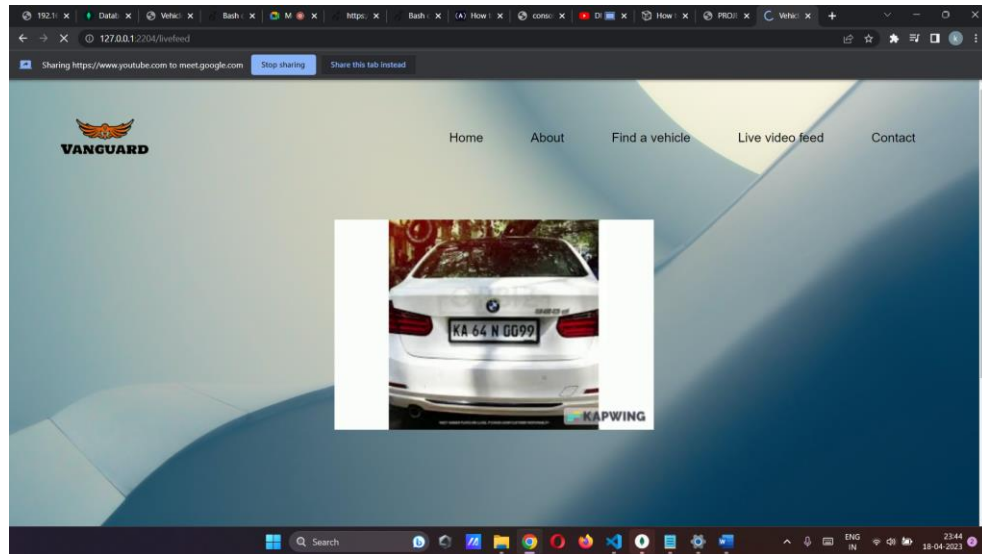


The screenshot shows the Vanguard website interface. The top navigation bar includes links for Home, About, Find a vehicle, Live video feed, and Contact. The main content area displays "Search Results" with a table containing two entries:

Time Stamp	Car No	Location
Tue Apr 18 20:19:02 2023	HR26DK83371	
Tue Apr 18 22:19:50 2023	HR26DK83371	

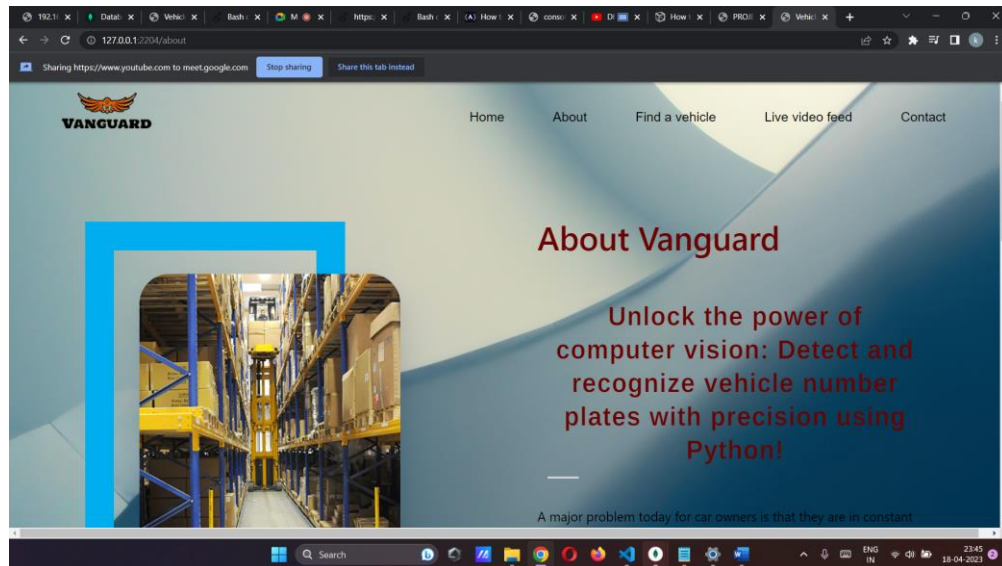
The footer contains links for "Who we are" (About, Team, Trademark guidelines) and "Help" (FAQ, Term & conditions, Support Policy). The browser's address bar shows the URL "127.0.0.1:2204/fiber".

Live video feed page:

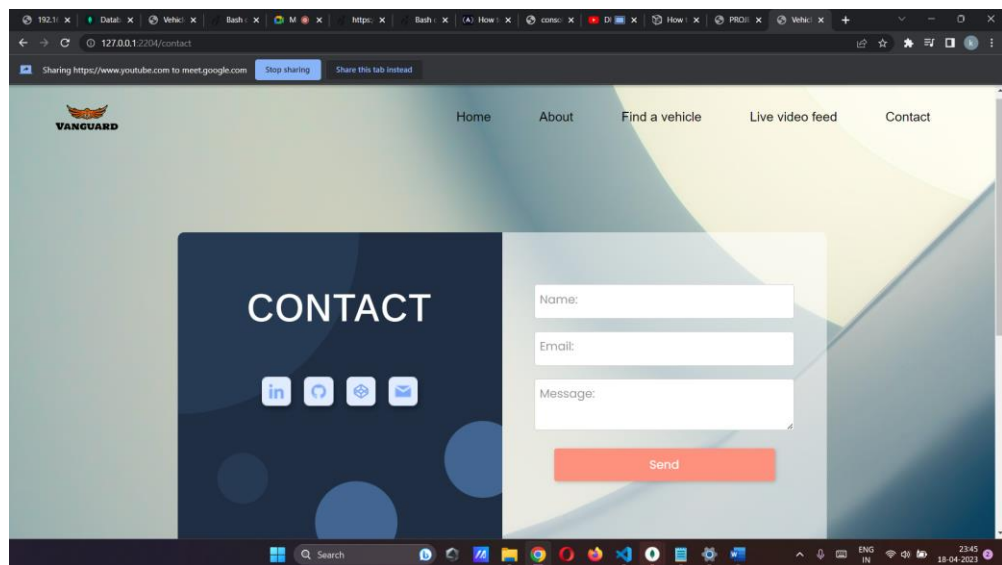


The screenshot shows the Vanguard website interface with the "Live video feed" page selected. The main content area displays a live video feed of a white car with the license plate "KA 64 N GG99". The car is parked in front of a building with a red roof. The video feed is framed by a white border. The browser's address bar shows the URL "127.0.0.1:2204/livefeed".

About page:

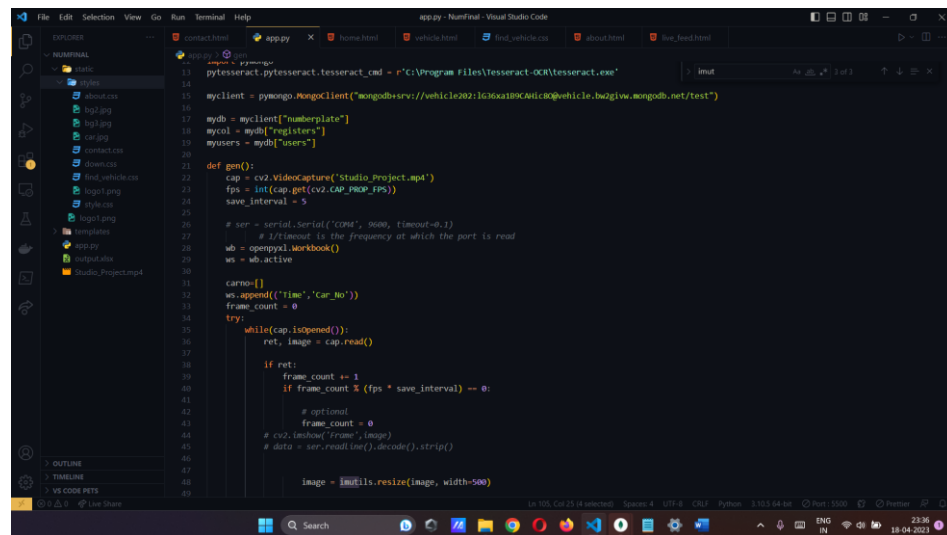


Contact page:



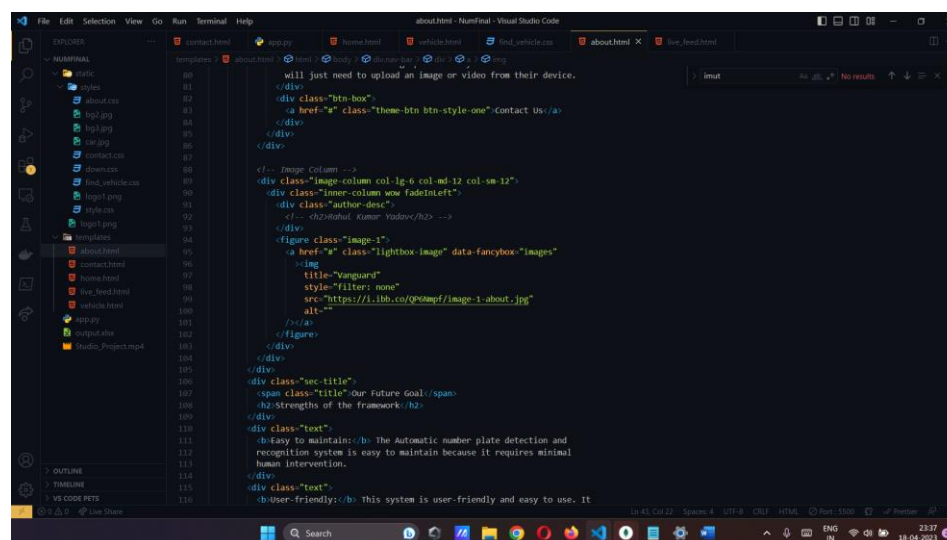
B. SOURCE CODE

App.py:



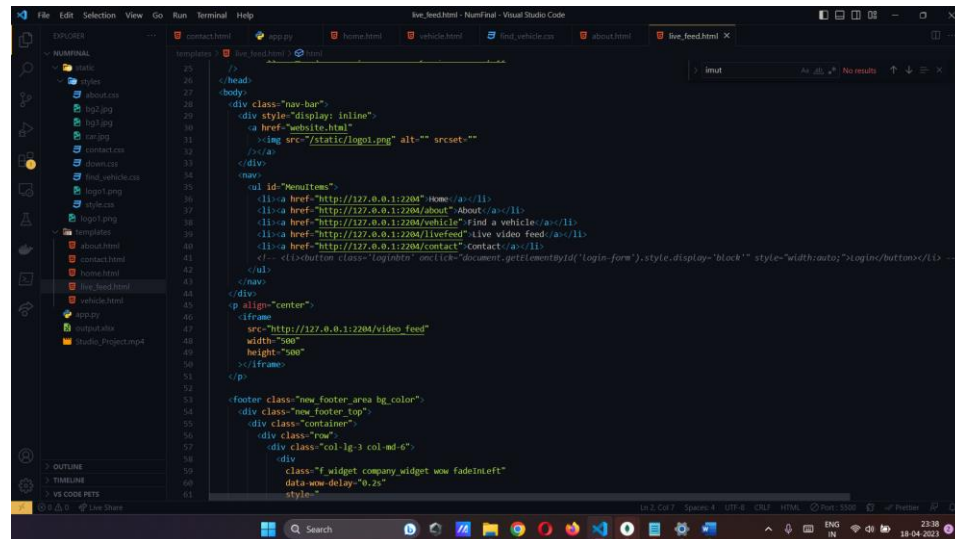
```
13 import cv2
14 import pytesseract
15 pytesseract.pytesseract.tesseract_cmd = r"C:\Program Files\Tesseract-OCR\tesseract.exe"
16 myclient = pymongo.MongoClient("mongodb+srv://vehicle2021:lgMxa189CAmC8Q@vehicle.b2giw.mongodb.net/test")
17 mydb = myclient["numberplate"]
18 mycol = mydb["registers"]
19 myusers = mydb["users"]
20
21 def gen():
22     cap = cv2.VideoCapture('Studio Project.mp4')
23     fps = int(cap.get(cv2.CAP_PROP_FPS))
24     save_interval = 5
25
26     # ser = serial.Serial('COM4', 9600, timeout=0.1)
27     # 1/fps is the frequency at which the port is read
28     ws = openpyxl.Workbook()
29     ws = ws.active
30
31     carno = []
32     ws.append(("Time", "Car_No"))
33     frame_count = 0
34     try:
35         while cap.isOpened():
36             ret, image = cap.read()
37
38             if ret:
39                 frame_count += 1
40                 if frame_count % (fps * save_interval) == 0:
41                     # optional
42                     frame_count = 0
43                     # cv2.imshow("frame", image)
44                     # data = ser.readline().decode().strip()
45                     image = cv2.resize(image, width=500)
```

About.html

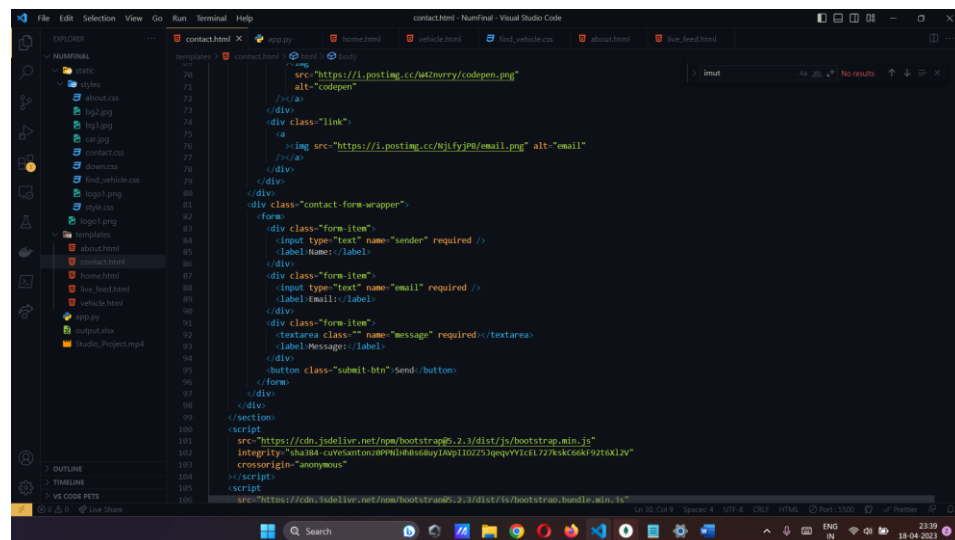


```
100 will just need to upload an image or video from their device.
101 </div>
102 <div class="btn-box">
103     <a href="#" class="theme-btn btn-style-one">Contact Us</a>
104 </div>
105 </div>
106
107 <!-- Image column -->
108 <div class="image-column col-lg-6 col-md-12 col-sm-12">
109     <div class="inner-column wow fadeInLeft">
110         <div class="author-desc">
111             <!-- ch2shubal Kumar Yadav/h2 -->
112         </div>
113         <figure class="image-1">
114             <a href="#" class="lightbox-image" data-fancybox="images">
115                 
121             </figure>
122         </div>
123     </div>
124     <div class="sec-title">
125         <span class="title">Our Future Goal</span>
126         <h2>Strengths of the framework</h2>
127     </div>
128     <div class="text">
129         <b>Easy to maintain:</b> The Automatic number plate detection and
130         recognition system is easy to maintain because it requires minimal
131         human intervention.
132     </div>
133     <div class="text">
134         <b>User-friendly:</b> This system is user-friendly and easy to use. It
```

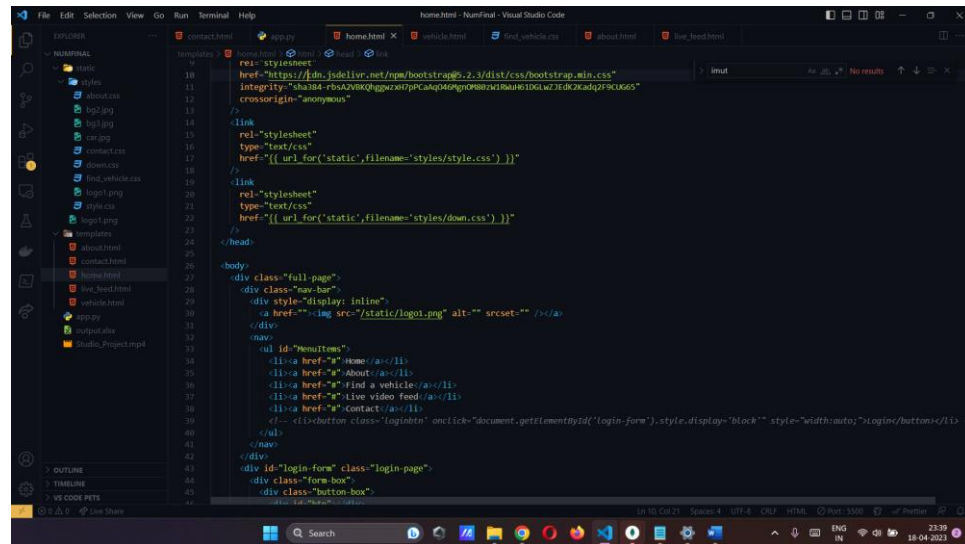
Livefeed.html:



Contact.html:

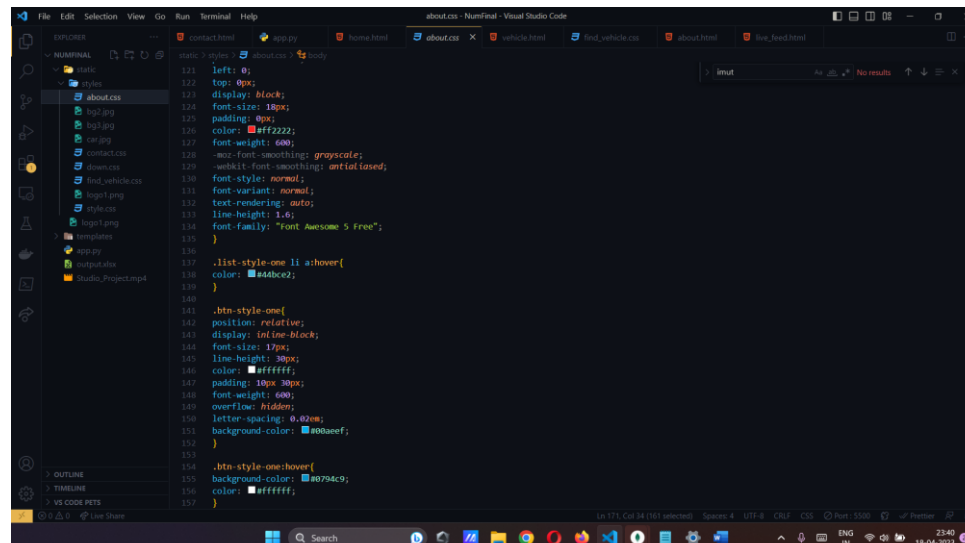


Home.html:



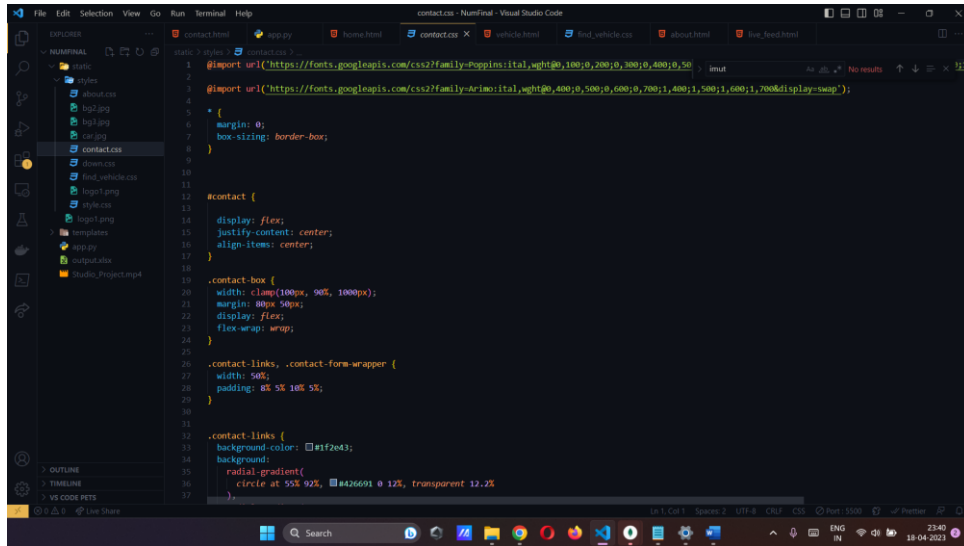
```
10  <!-- Stylesheet -->
11  <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.2.3/dist/css/bootstrap.min.css"
12  </link>
13  </head>
14  <body>
15  <div class="nav-bar">
16  <div style="display: inline-block; width: 50%; text-align: center;">
17  
18  </div>
19  <div style="display: inline-block; width: 50%; text-align: right;">
20  <ul style="list-style-type: none; padding: 0; margin: 0;">
21  <li><a href="#">Home</a></li>
22  <li><a href="#">About</a></li>
23  <li><a href="#">Find a vehicle</a></li>
24  <li><a href="#">Live video feed</a></li>
25  <li><a href="#">Contact</a></li>
26  </ul>
27  </div>
28  </div>
29  <div class="form-box">
30  <div class="login-form">
31  <div class="login-page">
32  <div class="button-box">
33  <button type="button" value="Login" />
34  </div>
35  </div>
36  </div>
37  </div>
38  </body>
39  </html>
```

About.css:



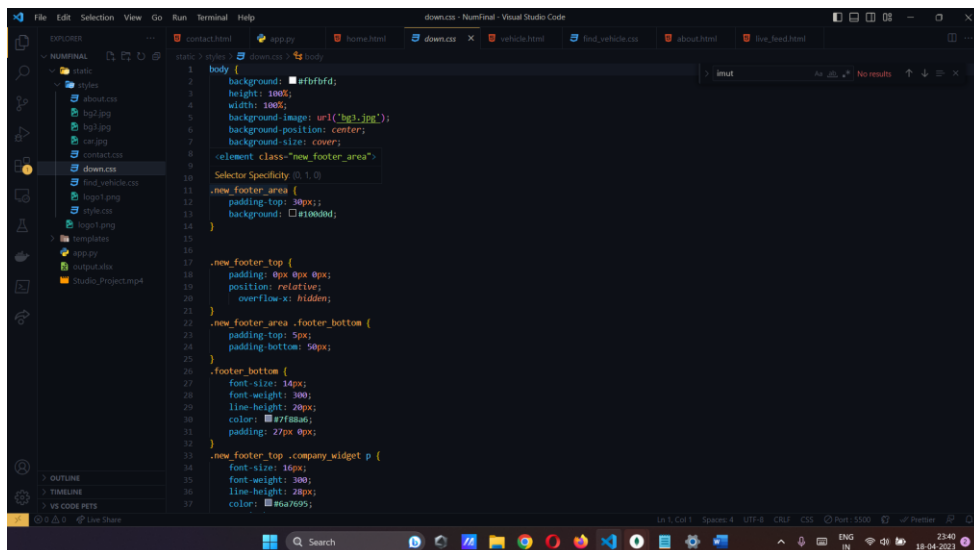
```
121  .list-style-one li a:active{
122  color: #444444;
123  }
124  .btn-style-one{
125  position: relative;
126  display: inline-block;
127  font-size: 17px;
128  line-height: 30px;
129  color: #ffffff;
130  padding: 10px 30px;
131  font-weight: 600;
132  overflow: hidden;
133  letter-spacing: 0.02em;
134  background-color: #000aeef;
135  }
136  .btn-style-one:active{
137  background-color: #00796c;
138  color: #ffffff;
139  }
```

Contact.css:



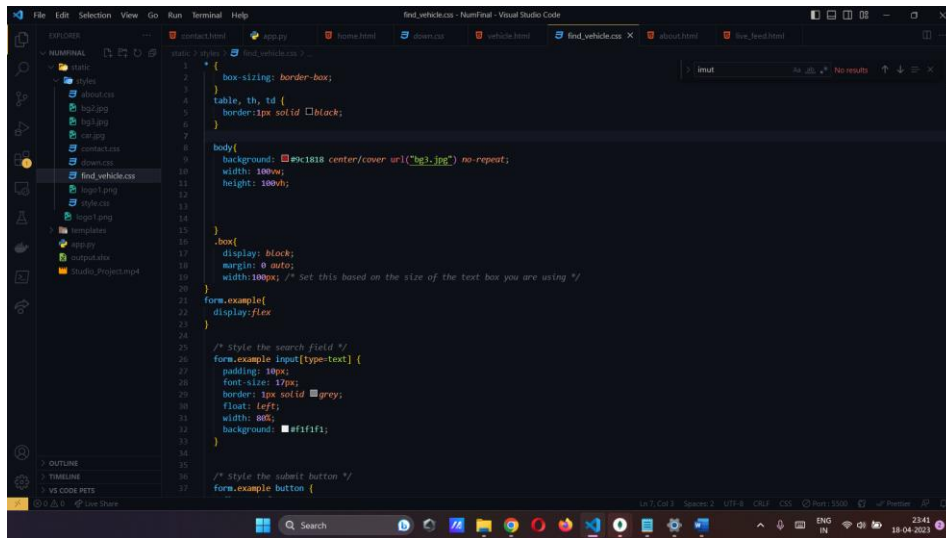
```
1 @import url('https://fonts.googleapis.com/css2?family=Poppins:ital,wght@0,100;0,200;0,300;0,400;0,500;0,600;0,700;0,800;0,900&display=swap');
2
3 @import url('https://fonts.googleapis.com/css2?family=Arimo:ital,wght@0,400;0,500;0,600;0,700;1,400;1,500;1,600;1,700&display=swap');
4
5 * {
6   margin: 0;
7   box-sizing: border-box;
8 }
9
10
11
12 #contact {
13   display: flex;
14   justify-content: center;
15   align-items: center;
16 }
17
18
19 .contact-box {
20   width: clamp(100px, 90%, 1000px);
21   margin: 80px 50px;
22   display: flex;
23   flex-wrap: wrap;
24 }
25
26 .contact-links, .contact-form-wrapper {
27   width: 50%;
28   padding: 8% 5% 10% 5%;
29 }
30
31
32 .contact-links {
33   background-color: #f1f2f3;
34   background:
35     radial-gradient(
36       circle at 55% 92%, #a46691 0 12%, transparent 12.2%
37     ),
38     radial-gradient(
39       circle at 55% 92%, #a46691 0 12%, transparent 12.2%
40     );
41 }
```

Down.css:



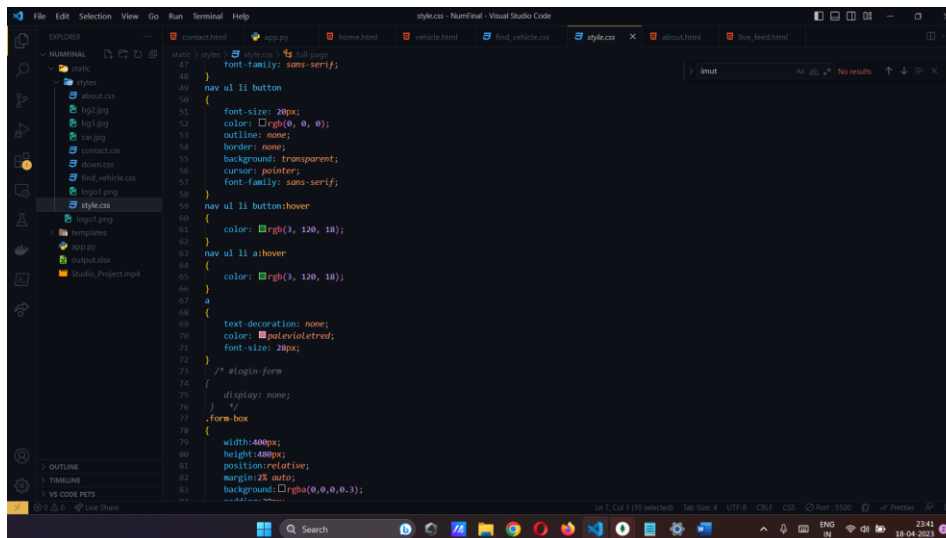
```
1 body {
2   background: #f1f2f3;
3   height: 100%;
4   width: 100%;
5   background-image: url('bg1.jpg');
6   background-position: center;
7   background-size: cover;
8   <element class="new_footer_area">
9
10  Selector Specificity (0, 1, 0)
11
12  .new_footer_area {
13    padding-top: 30px;
14    background: #f1f2f3;
15  }
16
17  .new_footer_top {
18    padding: 0px 0px 0px;
19    position: relative;
20    overflow-x: hidden;
21  }
22
23  .new_footer_area .footer_bottom {
24    padding-top: 5px;
25    padding-bottom: 50px;
26  }
27
28  .footer_bottom {
29    font-size: 14px;
30    font-weight: 300;
31    line-height: 20px;
32    color: #7f8c8d;
33    padding: 27px 0px;
34  }
35
36  .new_footer_top .company_widget_p {
37    font-size: 16px;
38    font-weight: 300;
39    line-height: 20px;
40    color: #7f8c8d;
41  }
```

Find_vehicle.css:



```
1 {
2   box-sizing: border-box;
3 }
4 table, th, td {
5   border: 1px solid black;
6 }
7
8 body {
9   background: #000000 center/cover url("bg3.jpg") no-repeat;
10  width: 1000px;
11  height: 1000px;
12 }
13
14 .box {
15   display: block;
16   margin: 0 auto;
17   width: 100%; /* Set this based on the size of the text box you are using */
18 }
19
20 form, example {
21   display: flex;
22 }
23
24
25 /* Style the search field */
26 form, example input[type="text"] {
27   padding: 10px;
28   font-size: 17px;
29   border: 1px solid grey;
30   float: left;
31   width: 80%;
32   background: #ffffff;
33 }
34
35 /* Style the submit button */
36 form, example button {
```

Style.css:



```
1 {
2   font-family: sans-serif;
3 }
4
5 nav ul li button {
6   font-size: 20px;
7   color: rgb(0, 0, 0);
8   outline: none;
9   border: none;
10  background: transparent;
11  cursor: pointer;
12  font-family: sans-serif;
13 }
14
15 nav ul li button:hover {
16   color: rgb(1, 120, 18);
17 }
18
19 nav ul li a:hover {
20   color: rgb(1, 120, 18);
21 }
22
23 a {
24   text-decoration: none;
25   color: #000000;
26   font-size: 20px;
27 }
28
29 /* #login-form
30 {
31   display: none;
32 }
33
34 .form-box {
35   width: 400px;
36   height: 400px;
37   position: relative;
38   margin: 20px auto;
39   background: rgb(0, 0, 0, 0.3);
40 }
```