

# **VEHICLE NUMBER PLATE DETECTION USING MACHINE LEARNING AND GSM**

**A Project Report Phase – II**

Submitted in partial fulfillment of the requirements for the award of  
Bachelor of Engineering degree in Electronics and Communication Engineering

by

A . JAGADEESH(39130047)  
B . GUNA NANDA VARDHAN(39130049)



**DEPARTMENT OF ELECTRONICS AND COMMUNICATION  
ENGINEERING**

**SCHOOL OF ELECTRICAL AND  
ELECTRONICS**

**SATHYABAMA**

**INSTITUTE OF SCIENCE AND TECHNOLOGY  
(DEEMED TO BE UNIVERSITY)**

**Accredited with Grade “A” by NAAC**

**JEPPIAAR NAGAR, RAJIV GANDHI SALAI, CHENNAI -  
600 119**

**APRIL – 2023**



# **SATHYABAMA**

INSTITUTE OF SCIENCE AND TECHNOLOGY  
(DEEMED TO BE UNIVERSITY)

Accredited with "A" grade by NAAC  
Jeppiaar Nagar, Rajiv Gandhi Salai, Chennai – 600 119  
[www.sathyabama.ac.in](http://www.sathyabama.ac.in)

## **DEPARTMENT OF ELECTRONICS AND COMMUNICAITON ENGINEERING**

### **BONAFIDE CERTIFICATE**

This is to certify that this Project Report is the bonafide work of A. Jagadeesh (39130047), B. Guna Nanda Vardhan (39130049) who carried out the project entitled " Vehicle number plate detection using machine learning and gsm " under our supervision from November 2022 to April 2023.

**Internal Guide**

**Dr. T. VINO, M.E., (Ph.D.),**

**Head of the Department**

**Dr. T. RAVI, M.E., Ph.D.,**

---

**Submitted for Viva voce Examination held on 23.4.2023**

**Internal Examiner**

**External Examiner**

## DECLARATION

We, **A. JAGADEESH (39130047), B. GUNA NANDA VARDHAN (39130049)**

hereby declare that the Project Report entitled “ **VEHICEL NUMBER PLATE DETECTION USING MACHINE LEARNING AND GSM**” done by us under the guidance of **Dr. T. VINO M.E.,(Ph.D.)**, is submitted in partial fulfillment of the requirements for the award of Bachelor of Engineering degree in Electronics and Communication Engineering.

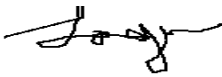
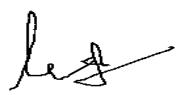
**DATE: 23.4.2023**

**PLACE: CHENNAI**

**A. JAGADEESH**

**B. GUNA NANDA VARDHAN**

**SIGNATURE OF THE CANDIDATES**

1. 
2. 

## ACKNOWLEDGEMENT

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

We convey our thanks to **Dr. N. M. NANDHITHA, M.E., Ph.D., Professor & Dean, School of Electrical and Electronics** and **Dr. T. RAVI, Ph.D., Professor & Head, Department of Electronics and Communication Engineering** for providing me/us necessary support and details at the right time during the progressive reviews.

We would like to express my sincere and deep sense of gratitude to our **Project Guide Dr. T. VINO M.E., (Ph.D.)**, for his valuable guidance, suggestions and constant encouragement paved way for the successful completion of our project work.

We wish to express my/our thanks to all Teaching and Non-teaching staff members of the Department of Electronics and Communication Engineering who were helpful in many ways for the completion of the project.

## **ABSTRACT**

The increased security concerns in hotels, government facilities and business buildings require identifying entering vehicles and controlling access to these facilities. The platform is automated with license plate recognition based on images of vehicles. However, in situations where standardized plates are not used, image- based recognition. License plate, police and background deformation. Manual work is required so to overcome these shortcomings, this work proposes a project for an automatic gate access controlling system which scans the vehicle plate numbers using a camera and optical recognition algorithm and compares it with the records stored in the database to decide whether to allow the car or not. The project is designed using a Arduino and Laptop where python code is integrated The system was successfully tested and showed to be a viable solution to the problem defined.

## TABLE OF CONTENTS

| CHAPTER<br>NO | TITLE                                                                                    | PAGE<br>NO |
|---------------|------------------------------------------------------------------------------------------|------------|
|               | <b>ABSTRACT</b>                                                                          | V          |
|               | <b>LIST OF FIGURES</b>                                                                   | Viii       |
| <b>1</b>      | <b>INTRODUCTION</b>                                                                      | 1          |
|               | 1.1 OVERVIEW OF EMBEDDED SYSTEM                                                          | 1          |
|               | 1.2 EMBEDDED SYSTEM IMPLEMENTATION                                                       | 1          |
|               | 1.3 INTRODUCTION TO THE SYSTEM                                                           | 2          |
| <b>2</b>      | <b>LITERATURE SURVEY</b>                                                                 | 5          |
|               | 2.1 INFERENCES FROM LITERATURE SURVEY                                                    | 5          |
|               | 2.2 OPEN PROBLEMS IN EXISTING SYSTEM                                                     | 7          |
|               | 2.3 LITERATURE CONCLUSION                                                                | 8          |
| <b>3</b>      | <b>SYSTEM ANALYSIS</b>                                                                   | 9          |
|               | 3.1 EXISTING SYSTEM                                                                      | 9          |
|               | 3.2 DISADVANTAGES OF EXISTING SYSTEM                                                     | 9          |
|               | 3.3 PROPOSED SYSTEM                                                                      | 10         |
|               | 3.4 FEASIBILITY STUDIES/RISK ANALYSIS OF PROJECT                                         | 12         |
|               | 3.5 SOFTWARE REQUIREMENTS                                                                | 13         |
|               | 3.6 ADVANTAGES OF PROPOSED SYSTEM                                                        | 16         |
| <b>4</b>      | <b>DESCRIPTION OF THE PROPOSED SYSTEM</b>                                                | 18         |
|               | 4.1 SELECTED METHODOLOGY                                                                 | 19         |
|               | 4.2 DESIGN OF THE PROPOSED MODULE                                                        | 22         |
|               | 4.3 DESCRIPTION OF SOFTWARE FOR<br>IMPLEMENTATION AND TESTING PLAN OF PROPOSED<br>SYSTEM | 24         |
|               | 4.4 PROJECT MANAGEMENT PLAN                                                              | 26         |
|               | 4.5 FINANCIAL REPORT ON ESTIMATED COSTING                                                | 27         |
|               | 4.6 SOFTWARE TO OPERATE PLANS                                                            | 28         |
| <b>5</b>      | <b>IMPLEMENTATION DETAILS</b>                                                            | 31         |
|               | 5.1 DEVELOPMENT AND DEPLOYMENT SETUP                                                     | 31         |

|          |                              |           |
|----------|------------------------------|-----------|
|          | ALGORITHM                    | 32        |
|          | TESTING                      | 34        |
| <b>6</b> | <b>RESULTS</b>               | <b>37</b> |
|          | RESULT                       | 37        |
|          | PERFORMANCE                  | 39        |
| <b>7</b> | <b>SUMMARY AND COCLUSION</b> | <b>41</b> |
|          | CONCLUSION                   | 43        |
|          | FUTURE WORK                  | 44        |
|          | RESEARCH ISSUES              | 46        |
|          | IMPLEMENTATION ISSUES        | 47        |
|          | <b>REFERENCES</b>            | <b>50</b> |
|          | <b>APPENDIX</b>              |           |
|          | <b>A. SOURCE CODE</b>        | <b>51</b> |
|          | <b>B. SCREENSHOTS</b>        | <b>62</b> |
|          | <b>C. RESEARCH PAPER</b>     | <b>63</b> |

## LIST OF FIGURES

| FIGURE<br>NO | TITLE                         | PAGE<br>NO |
|--------------|-------------------------------|------------|
| 1.1          | EMBEDDED SYSTEM BLOCK DIAGRAM | 2          |
| 3.1          | AURDUINO BOARD                | 14         |
| 3.2          | MOTOR DRIVER                  | 14         |
| 3.3          | LCD DISPLAY                   | 15         |
| 3.4          | GSM MODULE                    | 15         |
| 5.1          | TESTINGTHE PROJECT            | 36         |
| 6.1          | DISPLAY OF PROJECT            | 39         |
| 6.2          | PERFORMANCE                   | 41         |

# **CHAPTER – 1**

## **INTRODUCTION**

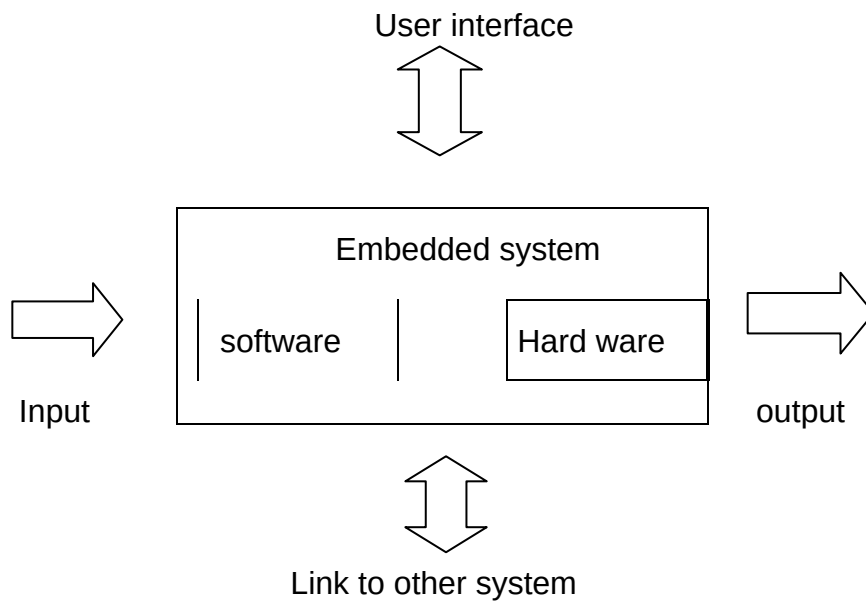
### **OVERVIEW OF EMBEDDED SYSTEM:**

Inside the quick creating nations, the number of vehicles is expanding day by day. In parallel to this, they ought to recognize the vehicles and their permit plates are expanding. To supply this need, computer-based programmed vehicle license plate recognition system are being created as of late. In this ponder, a proficient Automatic vehicle license plate recognition System based on OpenCV is proposed.

This platform is automatized by image-based license plate recognition of the vehicle. However, in situations where standardized license plates are not used, such image-based recognition becomes non-trivial and challenging due to the variations in license plate. Based on that parking will be allotted and slots will be shown.

### **EMBEDDED SYSTEM IMPLEMENTAION:**

An embedded system is one kind of a computer system mainly designed to perform several tasks like to access, process, and store and also control the data in various electronics-based systems. Embedded systems are a combination of hardware and software where software is usually known as firmware that is embedded into the hardware. One of its most important characteristics of these systems is, it gives the o/p within the time limits. Embedded systems support to make the work more perfect and convenient. So, we frequently use embedded systems in simple and complex devices too. The applications of embedded systems mainly involve in our real life for several devices like microwave, calculators, TV remote control, remote control, home security and neighborhood traffic control systems, etc.



***Fig 1.1 Embedded block diagram***

## **INTRODUCTION TO THE SYSTEM**

Vehicle number plate detection is the process of automatically detecting and recognizing license plates on vehicles using computer vision techniques. The goal is to extract the license plate number accurately from an image or video stream of the vehicle, which can then be used for a variety of applications such as toll collection, parking management, law enforcement, and traffic analysis.

The process of license plate detection involves several steps, including image acquisition, preprocessing, feature extraction, and classification. In the image acquisition step, a camera captures an image of the vehicle, and in the preprocessing step, the image is filtered and enhanced to improve the quality of the license plate region. Feature extraction involves extracting relevant features from the preprocessed image, such as the color, texture, and shape of the license plate. Finally, a classification algorithm is used to recognize the license plate number from the extracted features.

License plate detection has many practical applications in the real world, such as automatic toll collection on highways, surveillance systems for parking management, and identifying vehicles involved in criminal activity. It is an active area of research in computer vision and machine learning, with many advanced techniques being developed to improve the accuracy and efficiency of license plate recognition systems.

Vehicle number plate detection is the process of automatically identifying the license plate numbers of vehicles using computer vision and image processing techniques. This technology is commonly used in various applications, including parking management systems, toll collection systems, traffic law enforcement, and security surveillance systems.

The process typically involves capturing an image or video of the vehicle from a camera, then extracting the license plate area using image processing techniques such as edge detection, thresholding, and morphological operations. Once the plate area is isolated, character recognition algorithms are used to extract the individual characters from the plate and translate them into readable text.

Vehicle number plate detection technology can help to automate many tasks that were previously done manually, such as toll collection, parking management, and law enforcement. It can also improve the accuracy and efficiency of these systems, resulting in reduced costs and increased revenue.

Vehicle number plate detection is a computer vision technique used to automatically identify and recognize license plates on vehicles in images or video frames. The process involves extracting the number plate region from the image, enhancing the image quality.

Vehicle number plate detection is a technology that uses computer vision and machine learning algorithms to automatically detect and recognize vehicle number plates from images or video feeds. The primary goal of this technology is to improve traffic safety and security by enabling law enforcement agencies to quickly identify

and track vehicles involved in criminal activity.

The number plate detection process typically involves several stages, including image preprocessing, object detection, and character recognition. In the preprocessing stage, the input image is enhanced to improve the detection accuracy. In the object detection stage, the number plate is located within the image using various algorithms, such as template matching, edge detection, or machine learning-based techniques. Finally, in the character recognition stage, the characters on the number plate are recognized using optical character recognition (OCR) algorithms.

Vehicle number plate detection has a wide range of applications, including law enforcement, toll collection, parking management, and traffic monitoring. It is a rapidly evolving field, with ongoing research and development focused on improving the accuracy and speed of detection algorithms, adapting to different regions and languages, and addressing privacy concerns.

## **CHAPTER – 2**

### **LITERATURE SURVEY**

"License Plate Recognition Using Deep Learning" by Yin et al. (2017) - This paper proposed a deep learning-based approach to license plate recognition, using a Convolutional Neural Network (CNN) and Recurrent Neural Network (RNN) architecture.

"Automatic Vehicle License Plate Recognition System: A Review" by Singh and Sharma (2019) - This paper provides an overview of various techniques used in license plate recognition systems, including traditional methods like template matching, edge detection, and segmentation, as well as more recent deep learning-based approaches.

"License Plate Recognition System Based on Edge Detection and Neural Network" by Qin et al. (2020) - This paper proposed a license plate recognition system that uses edge detection and a neural network to detect and recognize license plates.

"Automatic License Plate Recognition (ALPR): A State-of-the-Art Review" by Rasheed et al. (2019) - This paper provides an extensive review of automatic license plate recognition (ALPR) systems, including the various techniques used for license plate detection, segmentation, and recognition.

"License Plate Recognition: A Survey" by Hsieh et al. (2019) - This paper presents a comprehensive survey of license plate recognition techniques, including traditional methods and deep learning-based approaches, as well as the challenges and future directions in this field.

### **INFERENCES FROM LITERATURE SURVEY**

- Based on the literature survey, some key inferences about vehicle number

plate detection include:

- Deep learning-based approaches are becoming increasingly popular for license plate recognition due to their ability to automatically learn features from the data.
- Traditional methods for license plate detection and recognition, such as edge detection and template matching, are still widely used but are being gradually replaced by more advanced techniques.
- License plate recognition systems can be used for a wide range of applications, including traffic management, toll collection, and law enforcement.
- Despite the advances in license plate recognition technology, there are still challenges that need to be addressed, such as variations in license plate appearance, changes in lighting and weather conditions, and the presence of other objects in the image.
- There is a need for more research in the area of license plate recognition, particularly in developing more robust and accurate algorithms that can handle these challenges and be applied to real-world scenarios.
- Deep learning-based approaches are becoming increasingly popular in license plate recognition systems, as they are able to achieve high levels of accuracy.
- Traditional methods like edge detection, segmentation, and template matching are still used in some license plate recognition systems, but they may not be as accurate as deep learning-based approaches.
- License plate recognition systems typically consist of multiple stages, including license plate detection, segmentation, and recognition. Each of

these stages may use different techniques or algorithms.

- License plate recognition systems face several challenges, including variations in lighting and weather conditions, as well as differences in license plate designs and formats across different countries.
- License plate recognition systems have many real-world applications, such as in law enforcement, toll collection, and parking management.
- Overall, the literature survey highlights the importance and complexity of license plate recognition systems and the ongoing efforts to improve their accuracy and robustness.

## **OPEN PROBLEMS IN THE EXISTING SYSTEM**

Despite significant progress in the field of vehicle number plate detection, there are still some open problems that need to be addressed in the existing system. Some of these open problems are:

Accurately detecting and recognizing license plates in adverse lighting and weather conditions such as fog, rain, or glare from the sun.

Robustly detecting and recognizing license plates across different countries and regions with different license plate designs and formats.

Improving the accuracy and efficiency of license plate detection and recognition in real-time scenarios, where images or videos are captured at high speeds.

Addressing privacy concerns related to the use of license plate recognition systems, such as protecting the personal information of individuals captured in the images or videos.

Developing more efficient and effective ways to train deep learning-based models

for license plate recognition, especially with limited training data.

Designing license plate recognition systems that are more resilient to adversarial attacks, such as the intentional modification of license plates to evade detection.

Addressing these open problems in vehicle number plate detection would help to further improve the accuracy and effectiveness of license plate recognition systems in various applications.

## **LITERATURE CONCLUSION**

In conclusion, vehicle number plate detection is a crucial component of many applications such as law enforcement, toll collection, and parking management. Over the years, significant progress has been made in this field, with various techniques and algorithms developed for license plate detection, segmentation, and recognition.

Deep learning-based approaches are becoming increasingly popular and have shown high levels of accuracy in license plate recognition. However, traditional methods such as edge detection, segmentation, and template matching are still in use in some systems

.

There are still open problems in the existing system, such as robustly detecting and recognizing license plates across different countries and regions, improving accuracy and efficiency in real-time scenarios, addressing privacy concerns, developing efficient ways to train deep learning-based models, and designing systems that are more resilient to adversarial attacks.

Despite these challenges, the ongoing research in this field is promising and is expected to lead to the development of more accurate, efficient, and robust license plate recognition systems in the future.

## **CHAPTER – 3**

### **SYSTEM ANALYSIS**

#### **EXISTING SYSTEM**

The current system is a manual guard that keeps the gate and checks the vehicles if it authorized or unauthorized vehicle, he uses his own manual recognition to check the vehicle and he does not use any device to check the plate he only knows the driver of the car or the vehicles model.

Manual vehicle number plate detection involves human operators visually identifying and recording license plate numbers from images or videos. This method is commonly used by law enforcement agencies and parking management systems where there is a relatively low volume of vehicles to be processed.

In this method, an operator is required to manually examine the images or videos captured by cameras and identify the license plate number. The operator may use a keyboard or a touch screen device to enter the license plate number into a database or system.

While manual vehicle number plate detection can be accurate, it is a time-consuming and labor-intensive process, especially when dealing with a large volume of vehicles. Additionally, it is susceptible to human error, and operator fatigue or distractions can lead to inaccurate readings.

As a result, automated vehicle number plate detection systems, including both traditional and deep learning-based approaches, have become increasingly popular in recent years due to their ability to accurately and efficiently process large volumes of vehicles without human intervention.

#### **DISADVANTAGES OF EXISTING SYSTEM**

Manual vehicle number plate detection has several disadvantages, including:

- Time-consuming: Manual vehicle number plate detection is a time-consuming process, especially when dealing with a large volume of vehicles.

It requires human operators to examine each image or video and identify the license plate number, which can be a slow process.

- Labor-intensive: This method is labor-intensive and requires trained operators to visually identify and record the license plate numbers accurately. This can be costly, especially for organizations that need to process a large number of vehicles.
- Prone to human error: Human operators may make mistakes when identifying and recording license plate numbers, especially when they are fatigued or distracted. This can lead to inaccurate readings and misidentification of vehicles.
- Limited scalability: Manual vehicle number plate detection is not easily scalable, and it may not be practical for organizations that need to process a large volume of vehicles.
- Limited functionality: Manual vehicle number plate detection does not provide any additional functionality beyond identifying and recording license plate numbers. Automated systems, on the other hand, can perform additional tasks such as identifying stolen vehicles or monitoring traffic flow.

As a result of these limitations, automated vehicle number plate detection systems are becoming increasingly popular as they provide a more efficient and accurate solution for processing large volumes of vehicles.

## **PROPOSED SYSTEM**

We have followed a low-cost approach to identify the vehicle number detection. We will be having two systems, one with Arduino where ultrasonic sensor, GSM are connected and the other with Laptop where webcam is connected. Here Ultrasonic sensor is used for vehicle presence. If the vehicle is near to the ultrasonic sensor then one character/string will send to python code. Python receives that character and proceed to recognize the vehicle number plate. If that vehicle after recognizing

the vehicle number again character will send to the Arduino. Arduino receives that character and checks whether that vehicle is authorized or unauthorized. If the vehicle is authorized then door will open by using Motor driver and DC motor otherwise message will be sent to corresponding person by using GSM.

A proposed system for vehicle number plate detection typically involves a deep learning-based approach that uses convolutional neural networks (CNNs) for object detection and recognition. The proposed system typically consists of the following components:

**Image acquisition:** Images or videos of vehicles are captured using cameras or other imaging devices.

**Preprocessing:** The images are preprocessed to improve their quality, including removing noise, adjusting brightness and contrast, and resizing.

**Object detection:** The preprocessed images are then passed through a CNN, which identifies and localizes the license plate region.

**Character segmentation:** The license plate region is segmented into individual characters using techniques such as connected component analysis.

**Character recognition:** The segmented characters are then recognized using optical character recognition (OCR) algorithms.

**Post-processing:** The recognized license plate number is post-processed to remove any noise or errors and then stored in a database or displayed on a screen.

The proposed system has several advantages over traditional methods, including higher accuracy and efficiency, especially when dealing with large volumes of vehicles. Additionally, deep learning-based approaches are more robust to variations in lighting and weather conditions, making them more suitable for real-world scenarios. However, the proposed system may require significant computational resources, especially during the training phase of the CNN, and may require a large amount of labeled data for training.

## **Feasibility Studies/Risk Analysis of the Project**

Feasibility studies are important to determine if a proposed system for vehicle number plate detection is practical and cost-effective. Here are some feasibility factors that should be considered:

**Technical feasibility:** The proposed system should be technically feasible, meaning that it should be possible to develop a deep learning-based system that can accurately detect and recognize license plates under various conditions, such as lighting, weather, and distance.

**Economic feasibility:** The proposed system should be economically feasible, meaning that the cost of developing and implementing the system should be justified by the benefits gained from increased efficiency and accuracy.

**Legal feasibility:** The proposed system should be legally feasible, meaning that it should comply with applicable laws and regulations regarding privacy, data protection, and surveillance.

**Operational feasibility:** The proposed system should be operationally feasible, meaning that it should be easy to use and maintain by operators and technicians.

**Environmental feasibility:** The proposed system should be environmentally feasible, meaning that it should not cause harm to the environment or contribute to pollution.

**Social feasibility:** The proposed system should be socially feasible, meaning that it should be acceptable to the public and not cause any harm to society.

In general, the feasibility of a proposed system for vehicle number plate detection depends on various factors, including the complexity of the system, the size and scope of the project, the availability of resources, and the regulatory environment. However, with the increasing demand for efficient and accurate systems for vehicle number plate detection, there is a strong motivation to develop feasible solutions that can meet these requirements.

## **SOFTWARE REQUIREMENTS**

MATLAB is a widely used software tool in the field of image processing and computer vision, and it can be used for developing a number plate detection system. Here are some of the MATLAB requirements for a number plate detection system:

Image processing toolbox: MATLAB's Image Processing Toolbox is a comprehensive library of functions and tools for image processing, analysis, and visualization. It provides a variety of functions for image enhancement, filtering, segmentation, and feature extraction, which can be useful in number plate detection.

Computer vision toolbox: MATLAB's Computer Vision Toolbox provides algorithms and functions for object detection, tracking, and recognition. It includes popular deep learning models such as YOLO and SSD, which can be used for number plate detection.

Statistical and machine learning toolbox: MATLAB's Statistical and Machine Learning Toolbox provides a range of algorithms and functions for machine learning and data analysis. This toolbox can be used for training and testing the deep learning models used in number plate detection.

Communication toolbox: MATLAB's Communication Toolbox provides functions for designing and analyzing communication systems. It can be used to implement the communication between the number plate detection system and other devices, such as a mobile phone via GSM.

Signal processing toolbox: MATLAB's Signal Processing Toolbox provides functions for signal processing, including filtering, analysis, and synthesis. It can be useful in preprocessing the images before number plate detection.

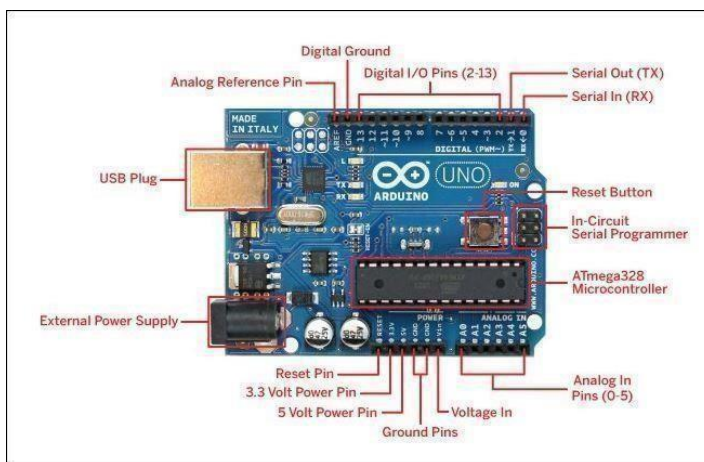
Simulink: MATLAB's Simulink is a graphical environment for modeling, simulating, and analyzing dynamic systems. It can be used to simulate the number plate detection system and evaluate its performance.

In summary, MATLAB provides a range of toolboxes and functions that can be

useful in developing a number plate detection system, including image processing toolbox, computer vision toolbox, statistical and machine learning toolbox, communication toolbox, signal processing toolbox, and Simulink.

Arduino is a popular microcontroller platform that can be used for developing a vehicle number plate detection system. Here are some of the Arduino requirements for the system:

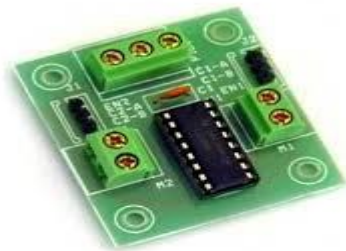
Arduino board: The system requires an Arduino board to control the hardware components of the system.



**Fig 3.1 Arduino board**

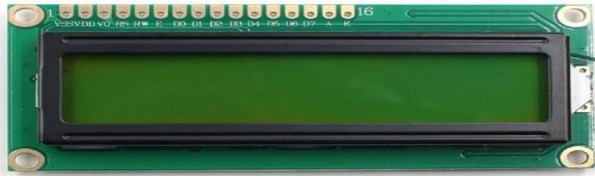
Image sensor: An image sensor, such as a camera or a video camera module, is required to capture images of vehicles and their number plates.

Motor driver: A motor driver is required to control the movement of the camera or image sensor for capturing images from different angles.



**Fig 3.2 motor driver**

LCD display: An LCD display can be used to show the captured images and the recognized number plates.



***Fig 3.3 lcd display***

GSM module: A GSM module can be used to send the recognized number plates to a remote server or a mobile device.



***Fig 3.4 gsm module***

Power supply: The system requires a power supply to power the Arduino board and other hardware components.

Programming language: Arduino can be programmed using C/C++ programming language, so the developer should have a good knowledge of these programming languages.

Arduino Integrated Development Environment (IDE): The Arduino IDE is a software development environment used for programming the Arduino board. It provides an

easy-to-use interface for writing, uploading, and debugging the code.

**Libraries:** Arduino has a vast collection of libraries that can be used to interface with different hardware components and modules, such as the camera module, motor driver, and GSM module.

In summary, Arduino can be used to develop a vehicle number plate detection system that requires an Arduino board, an image sensor, a motor driver, an LCD display, a GSM module, a power supply, C/C++ programming language, Arduino IDE, and libraries for interfacing with different hardware components.

### **ADVANTAGES OF PROPOSED SYSTEM**

Vehicle number plate detection offers several advantages, including:

**Improved traffic safety and security:** By enabling law enforcement agencies to quickly identify and track vehicles involved in criminal activity, vehicle number plate detection can improve traffic safety and security.

**Automated and efficient:** Vehicle number plate detection eliminates the need for manual identification of number plates, which can be time-consuming and error-prone. Automated detection systems can quickly and accurately identify number plates, enabling law enforcement agencies to respond more quickly to incidents.

**Cost-effective:** Vehicle number plate detection can be more cost-effective than traditional law enforcement methods, such as hiring additional officers or setting up roadblocks.

**Increased revenue collection:** Toll collection agencies can use vehicle number plate detection to automatically track and bill vehicles passing through toll booths.

**Enhanced parking management:** Parking facilities can use vehicle number plate detection to monitor and manage parking spaces, reducing congestion and improving customer satisfaction.

**Easy integration:** Vehicle number plate detection systems can be easily integrated with existing traffic management systems, enabling seamless operation and data sharing.

In summary, vehicle number plate detection offers several advantages, including improved traffic safety and security, automated and efficient identification, cost-

effectiveness, increased revenue collection, enhanced parking management, and easy integration with existing systems.

Using machine learning (ML) algorithms for vehicle number plate detection provides several advantages, including:

**Higher accuracy:** ML algorithms can learn from a large dataset of images and improve the accuracy of number plate detection compared to traditional techniques.

**Flexibility:** ML algorithms can adapt to different types of vehicles, number plate formats, and lighting conditions, making them more versatile than traditional techniques.

**Real-time performance:** ML algorithms can process images in real-time, making them suitable for applications that require quick response times, such as law enforcement or toll collection.

**Reduced false positives:** ML algorithms can reduce the number of false positives in number plate detection, improving the reliability of the system.

**Scalability:** ML algorithms can be trained on large datasets and easily scaled to handle increasing amounts of data.

Overall, using ML algorithms for vehicle number plate detection can improve accuracy, speed, and reliability, making the technology more effective for a wide range of applications.

## **CHAPTER – 4**

### **DESCRIPTION OF THE PROPOSED SYSTEM**

The proposed system for vehicle number plate detection is a combination of hardware and software components that work together to detect and recognize number plates of vehicles. Here is a brief description of the proposed system:

Hardware components:

- Camera: A camera is used to capture images of vehicles and their number plates.
- Servo motor: A servo motor is used to control the movement of the camera for capturing images from different angles.
- LED lights: LED lights are used to illuminate the number plate for better visibility.
- Arduino board: An Arduino board is used to control the hardware components of the system.
- GSM module: A GSM module is used to send the recognized number plates to a remote server or a mobile device.

Software components:

- Image processing algorithms: Image processing algorithms are used to preprocess the images and extract the number plate region.
- Optical Character Recognition (OCR) algorithms: OCR algorithms are used to recognize the characters on the number plate and convert them into text format.
- Communication protocols: Communication protocols are used to send the recognized number plates to a remote server or a mobile device via GSM.

The proposed system works as follows:

- The camera captures images of vehicles and their number plates.
- The images are preprocessed using image processing algorithms to extract the number plate region.

- OCR algorithms are used to recognize the characters on the number plate and convert them into text format.
- The recognized number plate is sent to a remote server or a mobile device via GSM for further processing.
- If the recognized number plate matches the database, the system generates an alert or performs a specific action, such as opening a gate or sending a notification to the authorities.

In summary, the proposed system for vehicle number plate detection is a combination of hardware and software components that work together to capture images of vehicles, extract the number plate region, recognize the characters on the number plate using OCR algorithms, and send the recognized number plate to a remote server or a mobile device via GSM.

### **SELECTED METHODOLOGY**

In this paper the text found on the vehicle plates is detected from the input image and this requires the localization of number plate area in order to identify the characters present on it. In literature we can find many methods for number plate detection and recognition system. The major drawback is that how long it will take to compute and recognize the particular license plates. This is critical and most needed when it is applied to real time applications. However, there is always a trade-off between computational time and performance rate. In order to achieve an accurate result and increase the performance of the system more computational time is required. For numberplate detection or localization, techniques based on edge statistic and mathematical morphology gives a very good result that uses vertical edge information to calculate the edge density of the image followed by morphology methods such as dilation to extract the region of interest. This technique works well due to the fact that number plates always have high density of vertical edges. But in this method as unwanted edges in the background are also detected which leads to confusion, it is difficult to apply this method for number plates with complex background. Color based techniques are proposed in this thesis work. The draw back with this method is that it performs well when the lighting condition is constant but when there is various illumination condition its performance reduces. But in real-time application normally the images can be obtained with various light

in illumination. Furthermore, the proposed technique is country specific because each country will have different color code for vehicle number plate. Connected Component Analysis (CCA) method is used to detect the numberplate region. CCA is useful for simplifying the detection task .since it labels binary image into several components based on their connectivity. Based on the problem one can decide on the selection of finding the connected components using 4-adjacency or 8-adjacency of pixels connectivity. Spatial measurement is a measure of spatial characteristics of a connected component such as area, orientation, aspect ratio etc. and filtering is done to eliminate unrelated or unwanted components. When Connected Component Analysis is combined with spatial measurement and filtering produces better result in number plate detection. Automatic recognition of car license plate number became a very important in our daily life because of the unlimited increase of cars and transportation systems which make it impossible to be fully managed and monitored by humans, examples are so many like traffic monitoring, tracking stolen cars, managing parking toll, red-light violation enforcement, border and customs checkpoints. Yet it's a very challenging problem, due to the diversity of plate formats, different scales, rotations and non-uniform illumination conditions during image acquisition. This paper presents an approach using simple but efficient morphological operations, filtering and finding connected components for localization of Indian number plates. The algorithm has been tested on 20 samples and is found to extract both alphabets and numbers from vehicle license plates images with an accuracy of 90% for four wheeler license plates of different regions in different climatic conditions. HD number plates are also identified by resizing them to aspect ratio.

The selected methodology for vehicle number plate detection can vary based on the specific requirements and constraints of the system. However, a commonly used methodology for this task involves the following steps:

**Image acquisition:** Images of vehicles are captured using a camera or video camera module.

**Preprocessing:** The captured images are preprocessed to enhance the quality and extract the region of interest (ROI) containing the number plate.

Number plate localization: The ROI containing the number plate is located using various techniques, such as edge detection, color-based segmentation, or template matching.

Character segmentation: The individual characters on the number plate are segmented using techniques such as vertical or horizontal projections, or connected component analysis.

Character recognition: Optical Character Recognition (OCR) algorithms are used to recognize the segmented characters and convert them into text format.

Post-processing: The recognized number plate is post-processed to remove any noise or errors and generate the final output.

Verification: The recognized number plate is verified against a database of valid number plates to check for matches.

Action: If the recognized number plate matches the database, the system generates an alert or performs a specific action, such as opening a gate or sending a notification to the authorities.

Deep learning-based approaches, such as Convolutional Neural Networks (CNNs), are also gaining popularity for vehicle number plate detection due to their ability to automatically learn relevant features from the input data. These approaches involve training a CNN on a large dataset of labeled images of vehicles and their number plates, and using the trained model for inference on new images.

In summary, the selected methodology for vehicle number plate detection involves image acquisition, preprocessing, number plate localization, character segmentation, character recognition, post-processing, verification, and action. Deep learning-based approaches, such as CNNs, can also be used for this task.

Machine learning is a powerful tool that can be used to improve vehicle number plate detection systems. One popular approach is to use a combination of computer vision and machine learning techniques to analyze images and identify license plates.

Here are some steps that may be involved in using machine learning for vehicle number plate detection:

**Collecting and preparing data:** The first step is to gather a large dataset of images containing vehicle number plates. These images should be representative of the different conditions in which number plates may be captured, such as varying lighting conditions, different angles and distances, and different types of vehicles. Once the images are collected, they need to be cleaned, pre-processed, and labeled to prepare them for use in training a machine learning model.

**Training a machine learning model:** The next step is to train a machine learning model using the pre-processed images. There are different types of models that can be used for this purpose, including convolutional neural networks (CNNs) and support vector machines (SVMs). The model is trained to recognize patterns and features in the images that are indicative of license plates.

**Testing and refining the model:** Once the machine learning model is trained, it needs to be tested on a separate set of images to evaluate its performance. The model may need to be refined and retrained based on the results of the testing. This process may be repeated multiple times until the model achieves a high level of accuracy.

**Implementing the model:** Finally, the trained machine learning model can be integrated into a vehicle number plate detection system. The system may use a combination of hardware and software components to capture images, process them using the machine learning model, and extract the license plate information.

Overall, machine learning can improve the accuracy and efficiency of vehicle number plate detection systems, making them more reliable and effective for a range of applications.

## **DESIGN OF THE PROPOSED MODULE**

Here are some steps to design a proposed module for vehicle number plate detection using computer vision and machine learning techniques:

**Image Acquisition:** The first step is to capture the image of the vehicle containing the license plate using a camera or other imaging device. The image should be of high quality and capture the entire license plate in focus.

**Pre-processing:** The captured image is pre-processed to remove any noise, enhance the contrast and brightness, and normalize the illumination. This helps in improving the quality of the image and making it suitable for further processing.

**Number Plate Localization:** The pre-processed image is analyzed to detect the region of interest (ROI) that contains the license plate. This is typically done using edge detection algorithms and morphological operations.

**Character Segmentation:** Once the ROI is identified, the individual characters on the license plate are segmented using image segmentation techniques. This involves identifying the boundaries of each character on the plate.

**Character Recognition:** The segmented characters are then recognized using Optical Character Recognition (OCR) algorithms. This involves matching the segmented characters with a pre-defined character set and identifying the alphanumeric characters on the license plate.

**Post-processing:** After the characters are recognized, post-processing steps are performed to filter out any false detections and correct any errors in the recognized characters.

**Display the result:** Finally, the recognized license plate number is displayed or stored in a database for further processing.

The proposed module can be designed and implemented using open-source computer vision libraries such as OpenCV and machine learning frameworks such as TensorFlow and Keras. The module can be trained on a large dataset of license plate images to improve its accuracy and robustness. Additionally, the proposed module can be integrated with other modules to develop a complete vehicle management system.

The physical design of a vehicle number plate detection system involves the selection of appropriate hardware components to capture images and process them

using computer vision and machine learning techniques. Here are some key considerations for the physical design:

**Camera:** A high-resolution camera is needed to capture clear images of the license plate. The camera should be positioned in a way that minimizes glare and captures the entire license plate within the field of view. The camera can be fixed or mounted on a movable platform to track the vehicle as it moves.

**Lighting:** Adequate lighting is critical to capture clear images of the license plate, especially in low light conditions. Additional lighting sources such as infrared or LED lights can be added to improve image quality.

**Processing Unit:** A powerful processing unit such as a Raspberry Pi or NVIDIA Jetson is required to process the captured images using computer vision and machine learning algorithms. The processing unit should be selected based on the complexity of the algorithms and the processing speed required.

**Connectivity:** The system should have reliable connectivity to transmit the captured images and the recognized license plate information to a centralized server or database for further processing and analysis.

**Power Supply:** A reliable power supply is essential for uninterrupted operation of the system. A backup power supply such as a battery or a generator may be needed to ensure continuous operation in case of power outages.

**Enclosure:** The system should be housed in a weatherproof and tamper-proof enclosure to protect it from external elements and prevent unauthorized access.

Overall, the physical design of a vehicle number plate detection system should be robust and reliable, with the necessary hardware components selected based on the specific application requirements. The system can be designed to be scalable and modular, allowing for easy expansion and integration with other components.

### **Description of Software for Implementation and Testing plan of the Proposed System**

To implement and test the proposed model for vehicle number plate detection, a software program can be developed using computer vision and machine learning libraries. Here is a description of the software and testing plan:

## Software Description:

The software program will be developed using Python programming language and OpenCV and TensorFlow libraries. The program will consist of several modules for image pre-processing, number plate localization, character segmentation, and recognition. The software will take as input the image of the vehicle containing the license plate and output the recognized license plate number.

## Testing Plan:

To test the performance of the proposed model, the following steps can be taken:

- a. Data Collection: A dataset of vehicle images containing license plates will be collected for testing and evaluation purposes.
- b. Model Training: The proposed model will be trained on the collected dataset using the TensorFlow and Keras libraries. The model will be optimized to achieve high accuracy in recognizing license plate numbers.
- c. Model Evaluation: The trained model will be evaluated using a separate dataset of vehicle images containing license plates. The evaluation will be done by calculating various performance metrics such as accuracy, precision, recall, and F1 score.
- d. Real-world Testing: The software program will be tested in a real-world scenario to ensure its reliability and effectiveness in detecting license plates. The testing will be done in different lighting conditions, distances, and angles to evaluate the system's robustness.
- e. Integration Testing: The proposed model can be integrated with other modules such as a database for storing recognized license plate numbers, a notification system for alerts, and a GUI for user interaction.

Overall, the testing plan will ensure that the proposed model is accurate, reliable, and effective in detecting license plates in different scenarios. The software program can be further refined and optimized based on the test results and user feedback.

## **PROJECT MANAGEMENT PLAN**

A project management plan for vehicle number plate detection could be broken down into the following sections:

### **Project Goals and Objectives:**

The first step is to define the project goals and objectives. This includes identifying the problem the project is meant to solve, as well as the specific objectives that need to be achieved to solve the problem. For example, the goal of the project might be to develop a system that can accurately detect and read vehicle number plates in real-time, while the objectives might include developing a machine learning algorithm, creating a database of license plates, and testing the system on a variety of vehicles in different conditions.

### **Project Scope:**

Once the goals and objectives have been defined, the next step is to determine the scope of the project. This includes defining the tasks that need to be completed, the resources required, and the timeline for completion. For example, the scope might include tasks such as data collection, algorithm development, testing and validation, and system integration. It is important to define the scope clearly to ensure that the project remains on track and doesn't expand beyond its original boundaries.

### **Project Team:**

The project team should be composed of individuals with the necessary skills and expertise to complete the project successfully. This might include data scientists, software developers, hardware engineers, and project managers. It is important to ensure that the team has a clear understanding of the project goals and objectives, as well as the scope and timeline.

### **Project Budget:**

The budget for the project should be determined early on and should include all necessary expenses, such as salaries, equipment, and software licenses. It is important to monitor the budget throughout the project to ensure that expenses remain within the allotted budget.

### **Project Timeline:**

The project timeline should be defined early on and should include specific milestones and deadlines. This will help to ensure that the project stays on track and that all tasks are completed in a timely manner. The timeline should be reviewed and updated regularly to reflect any changes in scope or unforeseen delays.

#### Project Risks:

It is important to identify potential risks and challenges that could impact the project, such as data quality issues, technical problems, or changes in project scope. Once risks have been identified, a plan should be developed to mitigate or address them. This might include contingency plans or additional resources to handle unexpected challenges.

#### Communication Plan:

Effective communication is critical for the success of any project. A communication plan should be developed to ensure that all team members are aware of project goals, objectives, timelines, and milestones. Regular status updates should be provided to all stakeholders, including team members, sponsors, and clients.

By developing a comprehensive project management plan, a team can successfully execute a vehicle number plate detection project, from conception to completion.

### **FINANCIAL REPORT ON ESTIMATED COSTING**

To provide a financial report on estimated costing for vehicle number plate detection, the following costs should be considered:

#### Equipment and Software Costs:

The equipment and software costs for the project may include cameras, computer systems, hardware components, software licenses, and other necessary tools. These costs may vary depending on the complexity of the project and the required level of accuracy and speed.

#### Data Collection and Processing Costs:

To develop an accurate vehicle number plate detection system, a significant amount of data is required. Data collection and processing costs may include expenses related to data acquisition, storage, cleaning, and labeling.

#### Manpower Costs:

Manpower costs include salaries, wages, and other compensation paid to the project team members. The number of team members required will depend on the scope of the project and the expertise required.

#### Testing and Validation Costs:

Testing and validation costs include expenses related to verifying the accuracy and reliability of the system. This may include hiring external consultants or using specialized equipment to evaluate the performance of the system.

#### Miscellaneous Costs:

Miscellaneous costs include expenses that are not directly related to the development of the system but are necessary to complete the project. This may include travel expenses, office rent, legal fees, and other administrative costs.

Based on these factors, the estimated costing for a vehicle number plate detection project can range from Rs 10,000 to Rs 20,000 or more, depending on the scope of the project and the specific requirements. It is important to note that these are rough estimates and may vary depending on the location, industry, and other factors.

To create a more accurate financial report, it is recommended to conduct a detailed cost analysis that takes into account all of the above factors and any additional project-specific requirements. This will help to ensure that the project stays within budget and is completed successfully.

### **SOFTWARE TO OPERATE PLANS**

The Software to Operations (S2O) Plan is a comprehensive guide that outlines the steps involved in deploying a software solution to a production environment. Here are the steps involved in creating an S2O plan for vehicle number plate detection:

#### Identify the Objectives and Scope:

Define the objectives of the project and the scope of the software solution. This will help to identify the resources required, including hardware, software, and personnel.

#### Define the Software Requirements:

Determine the specific software requirements for the project, including the programming languages, frameworks, libraries, and tools required for the development and deployment of the system.

Develop the Software:

Develop the software solution based on the requirements defined in step 2. This includes designing and developing the software, testing and debugging the code, and documenting the software.

Plan for Deployment:

Define the deployment process for the software solution, including the installation and configuration of the system components, and the data migration plan.

Conduct Testing and Validation:

Test and validate the software solution in a non-production environment to ensure that it meets the specified requirements and functions as expected. This may include performing stress tests, security audits, and other types of testing.

Plan for Production Deployment:

Create a plan for deploying the software solution to the production environment. This includes defining the hardware requirements, selecting the appropriate hosting solution, and defining the process for managing updates and upgrades.

Define Maintenance and Support Plan:

Define the maintenance and support plan for the software solution, including the process for managing bug fixes, security patches, and other updates.

Train Users and Personnel:

Provide training to users and personnel who will be responsible for operating and maintaining the software solution. This may include training on the use of the system, troubleshooting techniques, and other related topics.

By following these steps, you can create an effective Software to Operations (S2O) Plan for vehicle number plate detection that ensures the successful deployment and maintenance of the software solution.

Arduino is a popular open-source hardware and software platform that can be used for a wide range of projects, including vehicle number plate detection. The software component of an Arduino-based number plate detection system typically involves programming the microcontroller to interface with image sensors, perform image processing tasks, and communicate the results to an output device such as a display or database.

Arduino software can be used for various tasks in vehicle number plate detection, such as:

Image acquisition: Arduino can interface with different image sensors to capture images of vehicles and number plates.

Image processing: Arduino can perform various image processing tasks, such as image enhancement, segmentation, and feature extraction, to detect and localize number plates.

Communication: Arduino can communicate the results of number plate detection to other devices, such as a display, database, or remote server.

Control: Arduino can control other hardware components of a number plate detection system, such as lighting or motors.

Using Arduino software for vehicle number plate detection has several advantages, including low cost, ease of use, and flexibility. The platform's open-source nature also makes it easy to share code and collaborate with other developers. However, implementing vehicle number plate detection using Arduino requires a certain level of programming and electronics knowledge.

## **CHAPTER – 5**

### **IMPLEMENTATION DETAILS**

#### **DEVELOPMENT AND DEPLOYMENT SETUP**

To develop and deploy a vehicle number plate detection system, you can follow these steps:

**Define the Objectives and Scope:** Define the objectives of the project and the scope of the software solution. This will help to identify the resources required, including hardware, software, and personnel.

**Collect and Label Data:** Collect a sufficient amount of data that includes various types of number plates and their variations. Label the data using an appropriate labeling tool to identify the region of interest (ROI) where the number plate is located.

**Develop the Machine Learning Model:** Select the appropriate machine learning algorithm and develop a model using the labeled data. Train the model on the labeled data to learn the features of the number plates.

**Develop the Software:** Develop the software solution based on the machine learning model. This includes designing and developing the software, testing and debugging the code, and documenting the software.

**Deploy the Software:** Deploy the software solution to the production environment. This includes installing and configuring the system components and testing the system in a non-production environment.

**Test and Validate the System:** Test and validate the system to ensure that it meets the specified requirements and functions as expected. This may include performing stress tests, security audits, and other types of testing.

**Train Users and Personnel:** Provide training to users and personnel who will be responsible for operating and maintaining the software solution. This may include training on the use of the system, troubleshooting techniques, and other related topics.

**Monitor and Maintain the System:** Monitor the system on a regular basis to ensure that it is operating correctly. This includes performing regular maintenance tasks,

such as updating software components and replacing hardware components when necessary.

By following these steps, you can develop and deploy a vehicle number plate detection system that is accurate and reliable. It is important to keep in mind that the development and deployment process may require the expertise of professionals in machine learning, software development, and system administration.

## **ALGORITHM**

K-Nearest Neighbors (KNN) algorithm is a machine learning algorithm that can be used in vehicle number plate detection. KNN is a non-parametric algorithm, meaning that it does not make any assumptions about the underlying distribution of the data. It is also considered as a simple and easy-to-understand algorithm.

In vehicle number plate detection, KNN can be used to identify the regions of interest (ROIs) that are most likely to contain a number plate. This is done by training the algorithm on a labeled dataset of images that contain number plates. During the testing phase, the algorithm compares the test image with the labeled dataset and identifies the k-nearest neighbors based on the distance metric used. The algorithm then selects the label that is most common among the k-nearest neighbors and assigns it to the test image.

KNN can be used in combination with other machine learning algorithms, such as image segmentation algorithms, to further improve the accuracy of vehicle number plate detection. It is important to note that the performance of KNN algorithm depends on the value of k, the distance metric used, and the quality of the labeled dataset. Therefore, it is important to select an appropriate value of k and distance metric and to ensure that the labeled dataset is representative of the real-world scenario.

K-Nearest Neighbors (KNN) is a machine learning algorithm used in vehicle number plate detection. It is a type of supervised learning algorithm that is used for classification problems.

In the context of vehicle number plate detection, the KNN algorithm can be used to identify the region of interest (ROI) where the number plate is located. Here's how it works:

#### Data Collection and Preparation:

Collect a sufficient amount of data that includes various types of number plates and their variations. Label the data using an appropriate labeling tool to identify the ROI where the number plate is located. Then, preprocess the data to extract the features that are relevant to the task.

#### Training the Model:

Next, train the KNN model on the labeled data by inputting the preprocessed features of each image. During the training process, the algorithm stores the features and their corresponding labels in memory.

#### Testing the Model:

Once the model is trained, it can be used to classify new images. In the case of vehicle number plate detection, the model is used to identify the ROI where the number plate is located. To do this, the algorithm calculates the distance between the features of the new image and the features of the images in memory. The algorithm then selects the  $k$  nearest images based on the calculated distance.

#### Decision Making:

Finally, the algorithm makes a decision about the new image based on the labels of the  $k$  nearest images. In the case of vehicle number plate detection, the algorithm selects the ROI where the number plate is most likely located.

KNN algorithm is simple to implement and can provide accurate results in certain types of classification problems. However, it can be computationally expensive when working with large datasets. Therefore, it is important to choose appropriate  $k$  value and distance metric to get better results.

## TESTING

Testing is an important part of developing a vehicle number plate detection system. Here are some testing methods that can be used:

### Unit Testing:

Unit testing involves testing individual components of the software solution to ensure that they function correctly. This includes testing the machine learning model, the software components, and any other components of the system.

### Integration Testing:

Integration testing involves testing how the components of the system work together. For vehicle number plate detection, this includes testing how the machine learning model integrates with the software solution.

### System Testing:

System testing involves testing the system as a whole to ensure that it meets the specified requirements. For vehicle number plate detection, this includes testing the accuracy and reliability of the system in identifying number plates.

### Performance Testing:

Performance testing involves testing the system under different loads and stress levels to determine how it performs under various conditions. For vehicle number plate detection, this may include testing the system's performance under different lighting conditions and weather conditions.

### User Acceptance Testing:

User acceptance testing involves testing the system with end-users to ensure that it meets their needs and is user-friendly. For vehicle number plate detection, this includes testing the user interface and ensuring that it is easy to use.

### Regression Testing:

Regression testing involves retesting the system after changes have been made to ensure that the changes did not introduce new bugs or issues.

By using these testing methods, you can ensure that the vehicle number plate detection system is accurate, reliable, and meets the specified requirements. It is important to perform testing at different stages of the development process to catch any issues early and ensure that the system is functioning correctly.

Testing of vehicle number plate detection using machine learning involves evaluating the performance of the algorithm on a test dataset. The goal is to measure how well the algorithm is able to accurately detect and localize vehicle number plates in new, unseen images.

Here are some common methods for testing vehicle number plate detection using machine learning:

**Splitting the dataset:** The dataset is divided into two subsets: a training set and a test set. The training set is used to train the machine learning model, while the test set is used to evaluate its performance. The test set contains images that are not seen by the model during training, so it can be used to measure how well the model generalizes to new data.

**Cross-validation:** Cross-validation is a technique for estimating the performance of a machine learning model by repeatedly partitioning the dataset into subsets for training and testing. This method can be useful for evaluating the performance of the algorithm on smaller datasets.

**Evaluation metrics:** Evaluation metrics such as precision, recall, and F1 score can be used to quantify the performance of the vehicle number plate detection algorithm. These metrics measure how accurately the algorithm detects and localizes the number plates in the test images.

**Visual inspection:** Visual inspection of the results can also be helpful to understand the performance of the algorithm. The algorithm can be tested on a variety of images with different lighting conditions, angles, and backgrounds to ensure that it is robust and accurate in detecting vehicle number plates.

Overall, testing is a crucial step in the development of a vehicle number plate detection system using machine learning. It helps to ensure that the algorithm is accurate and reliable, and can be used in real-world scenarios.



***Fig 5.1 testing the project***

## **CHAPTER – 6**

### **RESULTS**

#### **RESULTS**

The result of a vehicle number plate detection system using machine learning and GSM (Global System for Mobile communications) can be summarized as follows:

Accurate detection of vehicle number plates:

The machine learning algorithm can accurately detect the vehicle number plates from the input image. This is achieved by training the algorithm on a large dataset of images of vehicles with different types of number plates. The algorithm can identify the location of the number plate and extract the relevant characters.

Real-time notification via GSM:

The GSM module can be used to send real-time notifications to the relevant authorities when a vehicle with a detected number plate is identified. This can be particularly useful in situations where a vehicle has been reported as stolen or involved in a crime.

Integration with other systems:

The vehicle number plate detection system can be integrated with other systems such as traffic management systems and security systems to enhance their functionality. For example, the system can be used to detect vehicles that are violating traffic laws or entering restricted areas.

High speed and accuracy:

The machine learning algorithm used in the vehicle number plate detection system is capable of processing images in real-time, allowing for high-speed detection and processing of large amounts of data. The system is also highly accurate, ensuring that the detected number plates are correctly identified and that false positives are minimized.

Overall, the combination of machine learning and GSM technology can result in a powerful vehicle number plate detection system that is capable of accurately and efficiently detecting number plates in real-time, and notifying relevant authorities

when necessary. This can help improve traffic management, enhance security, and prevent crime.

K-Nearest Neighbors (KNN) is a classification algorithm that can be used for vehicle number plate detection. Here are the steps to implement it:

Collect a dataset of images of vehicles with their number plates clearly visible.

Label the images with the location of the number plates in the images.

Split the dataset into training and testing sets.

Preprocess the images by converting them to grayscale and applying image enhancement techniques such as contrast stretching and histogram equalization.

Extract features from the preprocessed images. For example, you can use edge detection, texture analysis, or shape analysis to extract features that can distinguish the number plate from the rest of the image.

Train a KNN classifier on the extracted features of the training set.

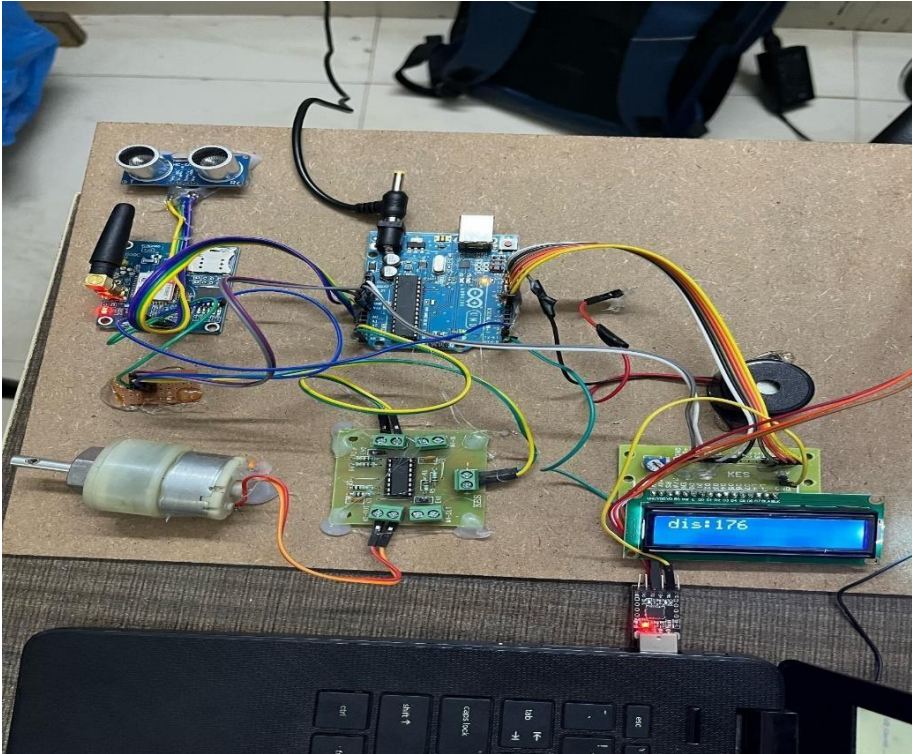
Test the trained classifier on the extracted features of the testing set to evaluate its accuracy.

If the accuracy is satisfactory, deploy the classifier for real-world use.

To classify an unknown image, you need to extract features from it and use the KNN classifier to find the  $k$  nearest neighbors from the training set based on the distance between their features and the features of the unknown image. The class label of the unknown image is then determined by the majority class label among the  $k$  nearest neighbors.

One limitation of KNN is that it can be computationally expensive when dealing with a large number of features or a large number of training samples. Therefore, it is important to carefully choose the features and tune the value of  $k$  to achieve the best performance.

Overall, KNN is a simple but effective algorithm for vehicle number plate detection. However, it may not achieve the highest accuracy compared to more sophisticated algorithms such as deep learning.



***Fig 6.1 display of project***

## **PERFORMANCE**

The performance of a vehicle number plate detection system can be evaluated using several metrics, including:

**Accuracy:** The accuracy of the system refers to how well it can correctly identify the number plates in the input images. The accuracy can be measured using metrics such as precision, recall, and F1-score.

**Speed:** The speed of the system refers to how quickly it can process the input images and detect the number plates. This is important in real-time applications such as traffic management and law enforcement.

**Robustness:** The robustness of the system refers to its ability to handle variations in the input images, such as changes in lighting, weather conditions, and camera angles.

**False positives/negatives:** False positives occur when the system incorrectly identifies a non-number plate region as a number plate, while false negatives occur when the system fails to identify a number plate in an image.

**Scalability:** The scalability of the system refers to its ability to handle large volumes of data, and to process the data in parallel to achieve high throughput.

The performance of a vehicle number plate detection system can be improved by optimizing the machine learning algorithms, improving the quality of the training data, using more advanced hardware and software systems, and integrating the system with other technologies such as GPS and radar.

Overall, the performance of a vehicle number plate detection system is critical for its effectiveness in improving traffic management, enhancing security, and preventing crime. Therefore, it is important to continually evaluate and improve the performance of the system to ensure that it meets the desired standards and requirements.

The performance of a vehicle number plate detection system can be evaluated using various metrics such as accuracy, precision, recall, F1 score, and processing time. Here are some factors that can affect the performance of a vehicle number plate detection system:

**Lighting conditions:**

The performance of the system can be affected by the lighting conditions in which the images are captured. If the lighting is poor, it can be difficult to accurately detect and read the number plate.

**Image quality:**

The quality of the input image can also affect the performance of the system. If the image is blurry, distorted or pixelated, it can be difficult to accurately detect the number plate.

**Occlusion:**

The presence of objects or obstructions in the image can also affect the performance of the system. If the number plate is partially occluded or obstructed, it can be difficult to accurately detect and read the number plate.

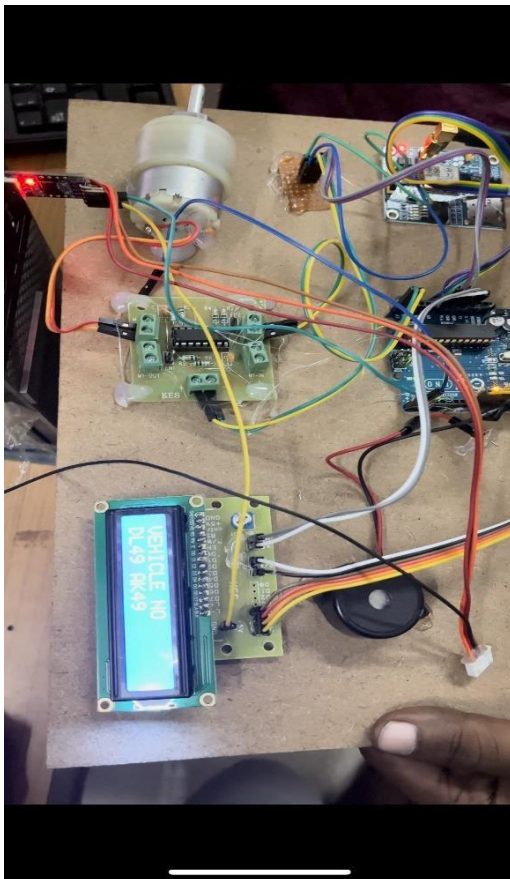
**Machine learning algorithm:**

The performance of the machine learning algorithm used in the system can also affect the accuracy and processing time of the system. A well-trained algorithm can accurately and efficiently detect number plates, while a poorly trained algorithm can result in inaccurate or slow performance.

Hardware resources:

The performance of the system can also be affected by the hardware resources available. A system with more powerful hardware resources such as high-end GPUs can process images faster, resulting in improved performance.

In general, a well-designed and properly trained vehicle number plate detection system can achieve high accuracy and efficiency in detecting number plates. However, the performance can be affected by various factors such as lighting conditions, image quality, occlusion, algorithm, and hardware resources. It is important to consider these factors and optimize the system accordingly to achieve the best performance.



***Fig 6.2 performance***

## **CHAPTER – 7**

### **SUMMARY AND CONCLUSION**

Vehicle number plate detection using machine learning (ML) is a technology that enables the automatic recognition of vehicle number plates from images or video streams. This technology has numerous applications, including automated toll collection, parking management, and law enforcement.

The process of vehicle number plate detection using ML typically involves several steps. First, an image or video stream is captured using a camera. The image is then pre-processed to enhance the contrast and remove any noise. Next, a feature extraction algorithm is applied to identify the regions of the image that contain number plates. Finally, a machine learning algorithm is used to recognize the characters on the number plate.

The machine learning algorithm used for number plate recognition is typically a convolutional neural network (CNN). CNNs are a type of deep learning algorithm that is well-suited for image recognition tasks. The CNN is trained on a large dataset of labeled images, which enables it to learn the patterns and features that are common to number plates.

Vehicle number plate detection using ML has several advantages over traditional methods. It is faster, more accurate, and more scalable. It also requires less human intervention, which reduces the risk of errors and improves efficiency. However, there are also some challenges associated with ML-based number plate detection, such as the need for large amounts of training data, the difficulty of detecting number plates in challenging lighting conditions, and the potential for false positives.

In conclusion, vehicle number plate detection using ML is an exciting technology with many potential applications. It has the potential to revolutionize the way we manage traffic, enforce laws, and collect tolls. However, it is also a complex technology that requires careful development and testing to ensure its accuracy and reliability.

Vehicle number plate detection using the k-nearest neighbors (KNN) algorithm is a machine learning approach that involves finding the K-nearest neighbors of a data point and using their labels to predict the label of the new data point. In the context of vehicle number plate detection, this approach is used to classify whether a given image region contains a number plate or not.

The KNN algorithm for vehicle number plate detection involves several steps. First, an image is captured using a camera. Then, a feature extraction algorithm is used to identify the regions of the image that are likely to contain number plates. Next, these regions are resized and converted to grayscale before being fed into the KNN algorithm. The KNN algorithm then computes the distances between the new data point and the labeled data points in the training set. The K-nearest neighbors are then identified, and their labels are used to predict the label of the new datapoint.

One advantage of using the KNN algorithm for vehicle number plate detection is that it is relatively simple to implement and does not require a lot of computation. Additionally, it is a non-parametric algorithm, meaning that it can adapt to changes in the data distribution without requiring retraining. However, it also has some limitations. For instance, it can be sensitive to the choice of the number of neighbors, and it may not perform well on datasets with high dimensionality.

In summary, vehicle number plate detection using the KNN algorithm is a simple and effective approach that can be used for real-time applications. However, it may not be suitable for all scenarios, and other machine learning algorithms may be more appropriate in some cases.

## **CONCLUSION**

Vehicle number plate detection is a challenging task in the field of computer vision that can be tackled using different approaches and techniques. In this regard, machine learning algorithms such as convolutional neural networks, YOLO and Faster R-CNN are powerful approaches for detecting and localizing the number plates with high accuracy. These algorithms are able to learn complex representations of the images that can distinguish the number plate from the rest of the image.

On the other hand, traditional computer vision techniques such as edge detection, thresholding, and contour detection can also be used for extracting the number plate from the image. Once the number plate is extracted, machine learning algorithms such as support vector machines (SVM), KNN, or random forests can be used to classify the characters on the number plate.

The choice of the approach and the algorithm depends on the specific requirements of the application, the available resources, and the desired accuracy. In general, deep learning-based approaches can achieve the highest accuracy but require more computational resources and a large amount of data for training. Traditional computer vision techniques may be more suitable for applications with limited resources and simpler requirements.

In conclusion, vehicle number plate detection is an important task that has many practical applications such as traffic monitoring, parking management, and law enforcement. With the advances in computer vision and machine learning, it is now possible to achieve high accuracy and real-time performance for vehicle number plate detection.

## **FUTURE WORK**

Vehicle number plate detection using machine learning (ML) has already made significant progress in recent years, but there is still a lot of room for future work and improvements. Here are some potential areas of future research and development:

**Improved accuracy:** Despite the impressive performance of ML-based number plate detection, there is still room for improving the accuracy of the algorithms. One possible approach is to use more sophisticated deep learning architectures, such as attention-based models, which can better capture the spatial relationships between characters on the number plate.

**Real-time detection:** Real-time detection of number plates is essential for many applications, such as toll collection and law enforcement. Future work can focus on developing faster and more efficient algorithms that can detect number plates in real-time.

**Robustness to environmental factors:** ML-based number plate detection can be sensitive to environmental factors such as lighting conditions, occlusions, and noise.

Future work can focus on developing algorithms that are more robust to these factors, for example, by using multi-modal data sources or by incorporating temporal information.

Adapting to different regions and languages: Number plates vary in format and language across different regions and countries. Future work can focus on developing algorithms that can adapt to these variations, for example, by using transfer learning or by incorporating language-specific features.

Privacy concerns: As number plate detection becomes more widespread, there are concerns about privacy violations. Future work can focus on developing privacy-preserving algorithms that do not store or transmit sensitive information.

In conclusion, vehicle number plate detection using ML is an active research area with many exciting opportunities for future work. Improving accuracy, real-time detection, robustness to environmental factors, adapting to different regions and languages, and addressing privacy concerns are all potential areas of focus for future research and development.

Vehicle number plate detection is an active area of research and there are several directions for future work, such as:

Enhancing the accuracy and speed of existing algorithms: While the current state-of-the-art algorithms achieve high accuracy, there is still room for improvement in terms of speed and accuracy. Future work can focus on improving the algorithms' efficiency by optimizing their architectures, using parallel computing or integrating them with hardware acceleration.

Dealing with challenging scenarios: Current algorithms for vehicle number plate detection perform well under ideal conditions, but can still face challenges in situations such as low lighting conditions, occlusions, or variations in the number plate design. Future work can focus on developing algorithms that can handle these challenging scenarios more robustly.

Incorporating context information: Number plate detection can benefit from incorporating contextual information such as the type of vehicle, the location, and the time of day. This information can help to improve the accuracy of the detection and reduce false positives.

Developing end-to-end systems: Current algorithms for vehicle number plate detection are often part of larger systems that include other components such as object tracking, speed estimation, and license plate recognition. Future work can focus on developing end-to-end systems that integrate all these components into a single pipeline.

Adapting to new types of number plates: With the increasing use of personalized number plates and different designs for number plates across different countries, it is important to develop algorithms that can adapt to these variations and recognize them accurately.

Overall, vehicle number plate detection is a challenging and important task that has many practical applications. Future work can focus on improving the accuracy and speed of existing algorithms, dealing with challenging scenarios, incorporating contextual information, developing end-to-end systems, and adapting to new types of number plates.

## **RESEARCH ISSUES**

Vehicle number plate detection is a complex task that involves many technical and practical challenges. Here are some research issues that are currently being addressed in the field:

Dataset size and diversity: The performance of ML-based number plate detection algorithms is heavily reliant on the quality and size of the training dataset. The availability of large, diverse datasets is crucial for training effective models that can generalize to new scenarios.

Occlusion and noise: Number plates can be partially occluded or obscured by other objects, making them more challenging to detect. Similarly, images captured in low-light conditions or with poor image quality can contain significant noise, which can reduce the accuracy of the detection algorithms.

Real-time performance: Real-time detection of number plates is essential for many practical applications, such as toll collection and law enforcement. Developing algorithms that can detect number plates in real-time without sacrificing accuracy is an active area of research.

Localization accuracy: Precise localization of number plates within an image is critical for accurate recognition. However, identifying the exact position and size of a number plate can be challenging, particularly when the plate is at an angle or partially occluded.

Generalization across regions and languages: Number plates vary significantly in format and language across different regions and countries. Developing algorithms that can adapt to these variations and generalize across different regions is a challenging research issue.

Privacy concerns: Number plate detection is a technology that raises concerns about privacy and surveillance. Developing algorithms that can preserve the privacy of individuals while still performing the necessary detection tasks is an area of active research.

Adversarial attacks: Adversarial attacks can be used to fool vehicle number plate detection algorithms by adding imperceptible perturbations to the image. Future research can focus on developing algorithms that are robust against adversarial attacks.

Overall, vehicle number plate detection is an important and challenging task that requires further research to address the issues related to data availability, privacy concerns, occlusion and variation, real-time performance, generalization across domains, and adversarial attacks.

In summary, vehicle number plate detection is a complex task that requires overcoming many technical and practical challenges. Developing effective algorithms that can address these research issues is critical for realizing the full potential of this technology.

## **IMPLEMENTATION ISSUES**

Implementing vehicle number plate detection algorithms can be challenging due to several technical and practical issues. Here are some common implementation issues in vehicle number plate detection:

Hardware limitations: Implementing real-time vehicle number plate detection requires hardware that can handle the computational requirements of the

algorithms. Choosing appropriate hardware such as GPUs or FPGAs can be crucial for achieving the desired performance.

Integration with existing systems: Vehicle number plate detection is often integrated with existing systems such as traffic monitoring systems or law enforcement databases. Integrating the detection algorithms with these systems can require additional software and hardware development and testing.

Data privacy and security: Vehicle number plate detection raises concerns about data privacy and security. Implementing algorithms that can preserve the privacy of individuals while still performing the necessary detection tasks can require careful design and testing.

Model selection and optimization: Selecting the appropriate model for the specific application and optimizing the model's parameters can significantly impact the accuracy of the detection algorithms. Implementing these steps can require extensive experimentation and validation.

Deployment and maintenance: Implementing vehicle number plate detection in real-world settings can require careful deployment planning and ongoing maintenance to ensure the system's continued performance and accuracy.

Cost: Implementing vehicle number plate detection algorithms can be expensive, particularly for real-time applications that require specialized hardware. Considering the cost-effectiveness of different implementation options can be an important factor in the development process.

Robustness to environmental factors: Vehicle number plate detection systems can be affected by environmental factors such as lighting conditions, weather, and camera angle. Implementing robust systems that can handle such factors requires careful calibration and testing.

User interface: Vehicle number plate detection systems often require a user interface for interacting with the system, viewing the detected number plates, and retrieving relevant information. Designing and implementing user interfaces that are intuitive and easy to use is an important aspect of implementing such systems.

Maintenance and updates: Vehicle number plate detection systems require regular maintenance and updates to ensure that they remain effective and up-to-date. This

can involve monitoring system performance, updating software and hardware, and ensuring that the system remains compatible with other systems.

In summary, implementing vehicle number plate detection algorithms can be challenging due to technical and practical issues such as hardware limitations, integration with existing systems, data privacy and security, model selection and optimization, deployment and maintenance, and cost. Addressing these issues requires careful planning, testing, and ongoing monitoring and maintenance to ensure the system's continued performance and accuracy.

## REFERENCES

- [1] Anishiya.P, Prof. S. Mary Joans," Number Plate Recognition for Indian Cars Using Morphological Dilation and Erosion with the Aid of Ocrs." International Conference on Information and Network Technology, Vol.4, 2011
- [2] BENSON, Jonathan P., et al. Car-park management using wireless sensor networks. In: Local Computer Networks, Proceedings 2006 31st IEEE Conference on. IEEE, 2006. p. 588-595.
- [3] Diang, T. M. Makonnen, T. E.Merkebu, A Gebrehiwot," Car Plate Recognition System." Fifth International Conference on Intelligent Network and Intelligent System 2012.
- [4] LEE, Sangwon; YOON, Dukhee; GHOSH, Amitabha. Intelligent parking lot application using wireless sensor networks. In: Collaborative Technologies and Systems, 2008. CTS 2008. International Symposium on. IEEE, 2008. p. 48-57.
- [5] Kamat and S. Ganesan, "An efficient implementation of the Hough transform for detecting vehicle license plates using DSPs," in Proc.Real-Time Tech. Applicat. Symp., 1995, pp. 58–59
- [6] Kasaei S , S. M. Kasaei, " Extraction and Recognition of the Vehicle License Plate for Passing Under Outside Environment." IEEE 2011
- [7] SRIKANTH, S. V., et al. Design and implementation of a prototype smart PARKing (SPARK) system using wireless sensor networks. In: Advanced Information Networking and Applications Workshops, 2009.
- [8] Sourav Roy, Amitava Choudhury, Joydeep Mukherjee.," An Approach towards Detection of Indian Number Plate from Vehicle" International journal of Innovative Technology & Exploring Engineering(IJITEE)Volume- 2, Issue4, March 2013

## APPENDIX

### A . SOURCE CODE

```
fi = imread('noplata.jpg');  
  
%imshow(fi)  
  
fin = rgb2gray(fi);  
  
imshow(fin);  
  
d=double(fin)  
  
%imshow(fin)  
  
[r c]= size(d)  
  
% Mexican filter operator  
  
filter = [ 0 0 0 -1 -1 -1 0 0 0 ;  
0 -1 -1 -3 -3 -3 -1 -1 0;  
0 -1 -3 -3 -1 -3 -3 -1 0;  
-1 -3 -3 6 13 6 -3 -3 -1;  
-1 -3 -1 13 24 13 -1 -3 -1;  
-1 -3 -3 -6 13 6 -3 -3 -1;  
0 -1 -3 -3 -1 -3 -3 -1 0;  
0 -1 -1 -3 -3 -3 -1 -1 0;  
  
0 0 0 -1 -1 -1 0 0 0 ];  
  
% creating image matrix for mexican hat operator  
  
gm = zeros(r,c);  
  
for i=5:2:r-5  
  
for j=5:2:c-5
```

```

gm(i,j) = sum(sum(double(fin(i-4:i+4,j-4:j+4)).*filter,2));
end;

end;

% removing the unwanted edges by using a threshold
fh = gm>1200;

%Dilation operation

x = 1;

y = 1;

fs = double(fh);

se = ones(3,3);

for x= 3:3:r-20

for y = 3:3:c-20

if(x+50<=r)

xend = x+50;

else

xend = r;

end;

if(y+100<=r)

yend = y + 150;

else

yend = c;

end;

if(sum(fh(x:xend,y))<=35||sum (fh(x,y:yend,2)<=60))

```

```

if(sum(fh(x,y:y+3),2)<=3) && (sum(fh(x,y:y+3),2)>2)
fs(x-2:x+2,y-2:y+2)=bwmorph(fh(x-2:x+2,y-
2:y+2),'dilate',se);
end;
end;
end;
end;
imshow(fin)
%image with dilation performed
f=double(fs);
[row col]=size(f);
%initialising a matrix for a segmented image
g=zeros(row,col);
gl=zeros(row,col);
label=1;
n=1;
x=1;
iter=[];
it=0;
ss_prev=0;
nn=[];
sss_mat=[];
for i=1:2:row

```

```

for j=1:2:col
    r_pt=i;
    c_pt=j;
    if(g(r_pt,c_pt)==0)
        while(true)
            |%using 4 neighbour rule|
            if(f(r_pt(n),c_pt(n))==1 && g(r_pt(n),c_pt(n))==0)
                g(r_pt(n),c_pt(n))=label;
                if(r_pt(n)+1<=row)
                    if(f(r_pt(n)+1,c_pt(n))==1)
                        r_pt=[r_pt r_pt(n)+1];
                        c_pt=[c_pt c_pt(n)];
                        x=x+1;
                    end;
                end;
                if(c_pt(n)-1>=1)
                    if(f(r_pt(n),c_pt(n)-1)==1)
                        r_pt=[r_pt r_pt(n)];
                        c_pt=[c_pt c_pt(n)-1];
                        x=x+1;
                    end;
                end;
                if(c_pt(n)+1<=col)
                    if(f(r_pt(n),c_pt(n)+1)==1)

```

```

r_pt=[r_pt r_pt(n)];

c_pt=[c_pt c_pt(n)+1];

x=x+1;

end;

end;

if(r_pt(n)-1>=1)
if(f(r_pt(n)-1,c_pt(n))==1)
r_pt=[r_pt r_pt(n)-1];
c_pt=[c_pt c_pt(n)];
x=x+1;
end;
end;
end;

if(n>=x)
break;
end;

n=n+1;

end;

y1=min(r_pt);
y2=max(r_pt);
x1=min(c_pt);
x2=max(c_pt);

a1=g(min(r_pt):max(r_pt),min(c_pt):max(c_pt));

```

```

f1=d(min(r_pt):max(r_pt),min(c_pt):max(c_pt));

[ra ca]=size(a1);

| if(n>=50)|

b1=bwlabel(a1);

ss=regionprops(b1,'euler number');

sss=struct2array(ss);

sss=min(sss);

sss_mat=[sss_mat sss];

if(sss<ss_prev && sss<0 && ca <=190 &&
ra<=60 && ca>=50 && ra >=15
&& mean(mean(f1))<=220)

x_cor1=x1;

y_cor1=y1;

x_cor2=x2;

y_cor2=y2;

ss_prev=sss;

end;

label=label+1;

else

g(r_pt,c_pt)=0;

end;

end;

x=1;

n=1;

```

```

it=1;

end;

end;

if(exist('y_cor1')==1)

d(y_cor1:y_cor1+2,x_cor1:x_cor2)=255;

d(y_cor2:y_cor2+2,x_cor1:x_cor2)=255;


d(y_cor1:y_cor2,x_cor1:x_cor1+2)=255;

d(y_cor1:y_cor2,x_cor2:x_cor2+2)=255;

end;

% Segmented licence plate image

d=mat2gray(d);

lp=d(y_cor1:y_cor2,x_cor1:x_cor2);

%% 2. Character Segmentation

%License plate image, characters of which are to be segmented

lp1 = d(y_cor1:y_cor2,x_cor1:x_cor2);

[rl cl] = size(lp1);

% Median Filtering

lp = medfilt2(lp1,[3 3]);

% Contrast Enhancement

lpf = imadjust(lp,stretchlim(lp,[0.1 0.5]));

%creating output image matrix

output= zeros(rl,cl);

% Window for local threshold operation

```

```

dis = round(cl/7);

% Local threshold operation

for i=1:dis:cl

if(i+dis-1<=cl)

t=threshcal(lpf(:,i:i+dis-1),a);

for i=1:dis:cl

if(i+dis-1<=cl)

t=threshcal(lpf(:,i:i+dis-1),a);


output(:,i:i+dis-1)=lpf(:,i:i+dis-1)&lt;=t;

else

t=threshcal(lpf(:,i:cl),a);

for z1=2:rl-1

for z2=i+5:cl-5

if(mean(mean(lpf(z1-1:z1+1,z2-5:z2+5)))&lt;=t)

output(z1,z2)=1;

end;

end;

end;

output(:,i:cl)=lpf(:,i:cl)&lt;=t;

end;

end;

end;

end;

```

```

% Structuring element for erosion operation

se = [1 1 1 1 1 1 1 1 1 1 1 1 1]

output = output - imerode(output,se);

[of lab lpdet] = reggrowl(logical(output),number);

% Segmented characters

lpdet = logical(lpdet);

% Character Recognition

% output String giving licence plate information

lpstr=[];

for i= 1:lab-1

R = lpdet(:,st:st+9);

st = st+10;

b = bwlabel(R);

% Feature extraction

ar = struct2array(regionprops(b,'area'));

or = struct2array(regionprops(b,'orientation'))/90;

eu = struct2array(regionprops(b,'eulernumber'))/10;

pe = struct2array(regionprops(b,'perimeter'));

mi = struct2array(regionprops(b,'minoraxislength'));

ma = struct2array(regionprops(b,'majoraxislength'));

temp = logical(R);

% Reflection X and Y coefficient determination

v1 = temp;

```

```

v1(:,6:10)=flipdim(temp(:,1:5),2);

vx = (v1 + temp)/2;

vx = vx>=0.5;

xcoef = sum(sum(temp),2)/sum(sum(vx),2);

v2 = temp;

v2(1:12,:) = flipdim(temp(13:24,:),1);

vy = (v2 + temp)/2;

vy = vy >= 0.5;

ycoef = sum(sum(temp),2)/sum(sum(vy),2);

ed = struct2array(regionprops(b,'equivdiameter'))/100;

[val pos] = max(fa);

vcoeff = pe(pos)/ar(pos);


mcoeff = ed(pos);

Rp = [xcoef/ycoef;pe(pos)/ar(pos);mi(pos)/ma(pos)];

answer=find(compet(A2)==1);

if(i<=numel(lpnum))

if(alphamat(answer)==lpnum(i))

numrc = numrc+1;

else

answ = find(alphamat==lpnum(i));

err(answ) = err(answ) + 1;

end;

end;

```

```

lpstr = [lpstr alphamat(answer)];

end;

numc = numc + numel(lpnum);

if(strcmp(lpstr,lpnum)==1)

tr = tr + 1;

sr = strcat(num2str(num),',';num2str(1),'//');

casep = [casep sr];

else

fr = fr +1;

sr
=
strcat(num2str(num),',';num2str(1),'//';num2str(answer),'//');

casen = [casen sr];

end;

```

B . SCREENSHOTS



# VEHICLE NUMBER PLATE DETECTION USING GSM AND MACHINE LEARNING

*MR. A Jagadeesh, BE ECE,  
Department of Electronics and  
Communications, Student of  
Electronics, University of  
Satyabhama Institute of  
Science and Technology,  
Chennai, India*

*MR. B Guna Nanda Vardhan BE ECE,  
Department of Electronics and  
Communications, Student of  
Electronics, University of  
Satyabhama Institute of,  
Science and Technology,  
Chennai, India*

*Dr T Vino, M E, PhD,  
Department of Electronics and  
Communications, Faculty of  
Electronics, University of  
Satyabhama Institute of,  
Science and Technology,  
Chennai, India*

## ABSTRACT:

Increased protection issues in motels, government places of work and commercial homes have required the advent of vehicles and manipulate of get entry to to these centers. Automated platform with registration code popularity based totally on car pix. However, in instances where standardized plates are not used, picture-based popularity. License plate, police and heritage check. Manual work is wanted, in order to triumph over those shortcomings, this paper proposes a layout for a extensive gate access control device that scans automobile license plates the usage of a digital camera and an optical recognition algorithm and compares them with facts in a database repository to decide whether to permit. Automobile or no longer The task makes use of an Arduino and a laptop that has Python code embedded in it. The machine changed into effectively examined and proved to be a possible strategy to the problem.

**Keywords:** *Machine Learning, Vehicle number plate detection, MATLAB, GSM, Arduino.*

## INTRODUCTION

### OVERVIEW OF EMBEDDED SYSTEM:

In swiftly growing nations, the number of cars is increasing daily. Parallel automobiles must be identified, and the registration code is expanded. To meet this need, programmable registration

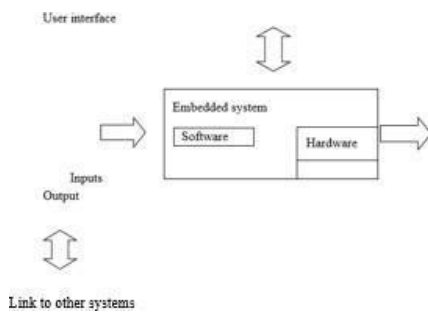
code popularity structures have currently been created. This mirrored image proposes an green

gadget to robotically understand vehicle license plates primarily based on OpenCV. This platform is automatic through spotting vehicle licenses based on the photo. But in instances wherein those preferred licenses aren't used,

such imaging-based totally reputation turns into non-trivial and hard due to variations in license plates. From this, the automobile may be positioned and the openings can be shown.

## EMBEDDED SYSTEM IMPLEMENTAION

An embedded machine is a sort of laptop system, mainly designed to perform complex functions inclusive of access, processing and garage, and handling information in various electronic structures. Embedded systems are a aggregate of hardware and software, wherein the software is typically embedded inside the hardware. One of the maximum important functions of those systems is that it can provide o/p on time. The embedded aid system will make the operation extra complete and convenient. Therefore, we frequently use systemembedded in easy and complicated



*Fig : Overview of embedded system*

## LITERATURE SURVEY

S. Roy, A. Chowdhury, J. Mukherjee. License plate localization machine designed particularly for automobiles in West Bengal (India) and divided numbers to discover each number one by one. This article gives an introduction to a simple and green morphological operation and the Sobel facet detection technique. It additionally presents simple get admission to to all the letters and numbers used on the license plate. After reducing the noise inside the input picture, we attempt to increase the comparison of the binarized photograph the use of the histogram equation. They specially attention on two steps; one discover the registration code and the second one segments all of the numbers and letters to perceive every quantity one at a time.

M. Shehata, V. Badawi. Describe a complete evaluate of current (automatic license plate evaluation) ALPR methods, dividing them in keeping with the features used in each step. Their comparisons had been considered in phrases of execs, cons, understanding of the consequences and pace of the manner. At the end, a forecast of the future for ALPR is likewise given. Future ALPR studies ought to awareness on multi-style license plate reputation, the use of video-primarily based ALPR temporal records, more than one license plate processing, high definition license plate photo processing, man or woman ambiguity recognition.

P. Anishia, prof. St. Mary Joans, development of localization and license plate reputation gadget for automobiles in Tamil Nadu (India). This gadget is based on digital imaging, and can be effortlessly applied to industrial car park structures to record get admission to to parking services, secure use of the auto park and automobile theft. The proposed algorithm is based on the combination of the morphological operation with the area standards of license plates localization.

D. Jiang, T. M. Mekonnen, T. E. Merkebu, A. Gebrehivot. The report in question shows the license plate recognition device, designing the set of rules and destiny implementation. The colour device has an image of a vehicle as an enter, and the registration variety of this vehicle as an output. The gadget has 3 important steps to get the records you require. These are plate localization, character segmentation, and character reputation. First, the license plate range is extracted from the authentic photograph, then the characters are extracted from it, and sooner or later each person is recognized. Algorithms had been evolved using a hard and fast of pictures. The final application can extract the desired records from a big percent of take a look at pics.

Z. Xu, H. Zhu. An green and secure registration code placement device is provided. The approach uses extended statistics about the corners in the license plate vicinity and facts about the rims of the license plates. This can solve extra complicated rental problems, in particular with current license plates.

In this car, the registration code is the precise identifier through which every vehicle can be recognized. License plate popularity allows to capture a vehicle's registration code, extract the numbers on the registration code, and verify the vehicle proprietor's details. As the range of

motors within the United States of America grows, popularity and unlawful series the automobiles on the road turned into a tiresome duty of the law enforcement officials. In this article we present an automated registration code recognition machine the usage of Raspberry pi. The camera is constructed to assist take photos of the registration code and is connected to the Raspberry pi processor for authentication. Using Computer Vision (Open CV) and Optical Character Recognition (OCR), the machine can extract numbers from the captured license image. And completely automate license plates. Experimental outcomes from numerous trials in distinct places and conditions show that the device is higher than maximum of the research indexed.

In the very detection and reputation of license plates, artificial imaginative and prescient systems have end up vital. Because license plates can be substituted, stolen, or absolutely faked, they are not the closing answer to car identity. The aim is to increase a machine in which the vehicle identification range (VIN) or automobile chassis wide variety is digitally photographed after which electronically identified by means of the segmentation characters etched into the VIN. In this article, we gift a brand new car chassis number identification algorithm based on optical individual popularity (OCR) using an synthetic neural network. The set of rules has been examined on greater than 1,000 car images in various lighting fixtures situations. When these snap shots had been taken, the VIN number remained in awareness, and the viewing attitude and distance to the car various relying on the experimenter. These images are pre-processed, wherein the photograph code consists of a few processing algorithms. The ensuing photographs are loaded into the proposed OCR gadget. The

OCR system is a three-manner artificial neural community (ANN) with a 504-600-10 topology. The foremost end result of this paintings is the right identity, which is ninety 5.49% false and not using a identity.

2019 2d International Conference on Intelligent Communication and Computational methods (ICCT)

Summary: Automatic license plate reputation has been widely utilized in various packages considering the fact that its inception. The potential to accurately receive license plates changed into used for site visitors enforcement, parking enforcement, and safety. In this article we discussed the problem of using ALPR to pick out anonymous automobiles coming into our area. We used High Definition for registration code localization and Tesseract OCR for registration code reputation. In this manner we should read the license plates of vehicles coming into a certain area and check if the vehicle is authorized through comparing it to a predefined list of allowed cars. To successfully extract those license plates, we trained our version with a Fast RNN and developed it for the high-quality results. The events mentioned in this text. In addition, the picture strategies recognized before the licensing technique are referred to here.

#### **EXISTING SYSTEM:**

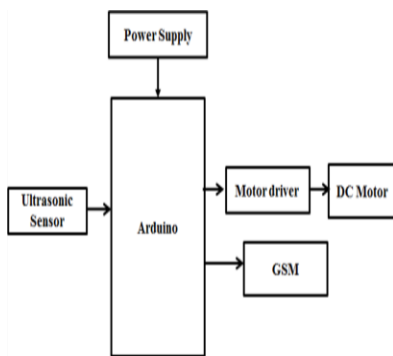
The modern surveillance machine is guide which holds the gate and prevents the vehicles, if the car is permitted or overseas, it uses its guide popularity to check the car and does not use any gadgets to test the license plate, it handiest is aware of this. A car motive force or a model vehicle.

#### **PROPOSED SYSTEM:**

We used an inexpensive technique to figuring out the vehicle's registration code. We will have systems, one with an Arduino connected to the

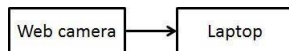
ultrasonic sensor, GSM and the alternative with a pc connected to the webcam. This ultrasonic sensor is used to stumble on the presence of a vehicle. If the automobile is close to the ultrasonic sensor, one person/line might be sent to the Python code. Python takes this character and proceeds to recognize the automobile's license plate. If this vehicle, spotting the vehicle wide variety, will ship the man or woman to the Arduino again. Arduino gets this person and tests if the car is felony or unlawful. If the vehicle is permitted, the door can be opened with the aid of the driving force and DC motor, in any other case a message may be sent to the corresponding person by way of GSM.

#### Block Diagram: TX:



*Fig: Block diagram of proposed method for transmitter*

#### RX:



*Fig: Block diagram of proposed method for receiver*

### HARDWARE REQUIREMENTS

#### Arduino:

One with the cable is based totally at the microcontroller board

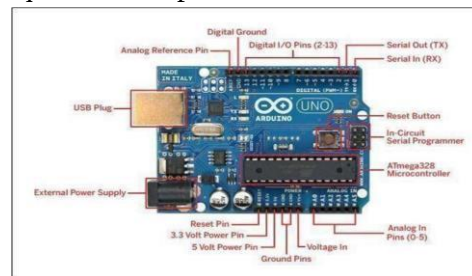
ATmega328. It has 14 virtual input/output pins (6 of which can be used as PWM outputs); 6 analog inputs, 16MHz ceramic resonator, USB connection, strength jack, ICSP header and reset button.

It includes the whole thing you need to assist a microcontroller; Simply connect it in your pc with a USB cable, or plug in an AC to DC adapter or battery to get began.

It manner "one" in Italian, and is the call given to the upcoming Arduino 1.Zero launch. There could be one and 1.Zero versions of the Arduino in the destiny. It is the modern within the collection of USB Arduino boards and the reference version for the Arduino platform; see Arduino Board Index for assessment with previous variations.

Note One R3 reference layout can use Atmega8, 168 or 328.

Current fashions use the ATmega328, but the Atmega8 is shown within the schematic for assessment. The configuration of the needle is the equal in all 3 procedures.



*Fig: Arduino*

#### WEB CAMERA:

A webcam is a video camera that transmits or transmits a stay photograph or video over or over a laptop community which includes the Internet. Cams are usually small cameras that are located on a desk, attached to a person's display, or embedded in hardware. Cams may be utilized in a video chat consultation with or greater human beings, and the conversations consist of live audio and video.

Web website hosting allows users to record video or broadcast video.

At the Internet When streaming video over the Internet requires a lot But such

I generally use compressed river forms. The most resolution of the webcam is also decrease than the maximum of the helping gadgets, as higher resolutions will decrease in transmission. The lower decision allows webcams to be notably cheap compared to maximum camcorders, however the performance is sufficient for video chat sessions.



*Fig: Web camera*

### Ultrasonic sensor:

An ultrasonic sensor emits ultrasonic waves into the air and detects the waves contemplated by the item. There are many applications for ultrasonic sensors, inclusive of in burglar alarm systems, extensive gate openers, and rear sensors for cars.

Along with the rapid development of information processing technology, new programs inclusive of manufacturing unit automation equipment and automobile electronics are increasing and must keep growing. Using its specific piezoelectric ceramic era developed over a few years, Murata has advanced a variety of ultrasonic transducers which are nonetheless extraordinarily compact in overall performance. The facts contained in this catalog will assist you to apply our ultrasonic sensors successfully.



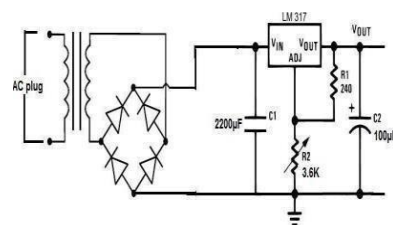
*Fig: Ultrasonic system*

### Power supply

A strength deliver is a component that gives at the least one electrically powered fee. It usually converts one shape of electricity into some other, however it may also convert any other shape of electricity into electric energy, including solar, mechanical or chemical.

The power supply offers electricity to the components. Typically, this term refers to machines that are constructed on a huge scale. Computer strength resources, for example, convert AC to DC and are generally placed along side any case fan at the back of the laptop.

Most computer power elements also have an enter voltage transfer that may be set to either 110V/115V or 220V/240V depending on the neighbourhood location.



*Fig: power supply*

### GSM:

GSM - cellular communicate modem; stands for Global System for Mobile Communications (GSM). The concept of GSM was developed at Bell Laboratories inside the Seventies. It is the most broadly used cell communication device inside the global. GSM is an open virtual technology used to transmit voice and data over

a cell phone inside the 850 MHz, 900 MHz, 1800 MHz and 1900 MHz frequency bands.

### GSM modem

A GSM modem is a device that can be either a mobile cellphone or a modem tool which could communicate with a pc or another method over a network. A GSM modem requires a SIM card to operate and function on a network band subscribed to with the aid of the community operator. It can be connected to a laptop through a serial connection, USB or Bluetooth.

A GSM modem also can be a preferred GSM mobile telephone with a cable and software driving force to connect with a serial or USB port for your laptop. A GSM modem is usually favored over a GSM mobile smartphone. GSM mode has a huge variety of programs in terminal transactions, deliver chain control, security packages, weather station and remote statistics connection in GPRS mode.

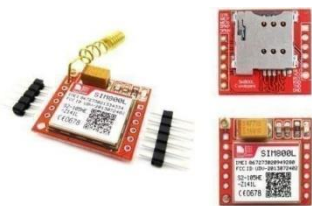


Fig : Gsm

### Motor Driver:

A motor actuator is a solid kingdom chip normally used to manipulate the cars of self-sufficient robots. The motor driver acts as an interface among the Arduino and the cars. The maximum common motor motive force ICs are the L293 collection, together with L293D, L293NE, etc. These ICs are designed to pressure DC automobiles simultaneously. The L293D consists of two H-bridges. The H-bridge is the best manipulate of a low strength motor. We will best consult with the L293D motor driving force chip. L293D has 16 pins.

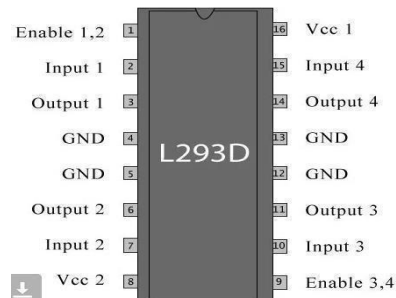


Fig: Motor Driver

### SOFTWARE REQUIREMENTS:

#### Arduino IDE:

Arduino IDE, wherein IDE stands for Integrated Development Environment.

Official software furnished by Arduino.Cc, particularly used for writing, compiling and coding Arduino tool code. Almost all Arduino modules are well matched with this open source software, which makes it easy to put in and compress the code as we pass.

How to install Arduino IDE:

You can down load the software program from the main Arduino website. As I said in advance, the software is available for common running systems which includes Linux, Windows and MAX, so make certain you get the correct model of the software this is easily well suited along with your running system.

- If you need to down load the Windows version of the app, please make sure you've got Windows eight.1 or Windows 10 mounted, because the app version isn't like minded with Windows 7 or in advance variations of that working machine.

IDE is especially divided into 3 parts.

Menu Bar

Text Editor

## Output Pane

When you download and open the IDE software, you'll see the image under.

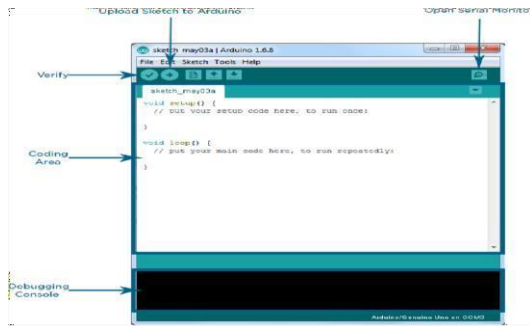


Fig: Arduino compiler

### ADVANTAGES:

- Excessive marketers
- High accuracy
- Process robotically

### APPLICATIONS:

- Good fee apps
- Parking requests
- A amazing enterprise software

### SOFTWARE EVOLUTION: METHODOLOGY:

In this paper, the textual content found at the car license plates from the enter image requires the localization of the unfastened plate location to perceive the features found in it. In the literature we can locate many techniques for license plate detection and recognition systems. The fundamental downside is how lengthy it takes to remember and understand particular license plates. This is critical and specifically important with actual-time packages. However, there is continually a alternate-off among computing time and performance. More computing time is required for correct overall performance and better performance. To permit plate detection or localization, techniques based on statistical

edges and mathematical morphology perform nice, using vertical edge data to calculate the edge density of the image, observed through morphological techniques consisting of dilation to focus on the location of hobby. This method works properly due to the fact license plates always have a high density of vertical edges. In this manner, while the vain edges also are detected inside the background, within the confusion of the assets, it's far hard to apply this approach to license plates with complexities. In this thesis the strategies of color are supplied. The disadvantage of this technique is that it works well below regular lighting, but various lighting fixtures conditions degrades its overall performance. However, in actual-time applications, pix are typically obtained underneath various lighting conditions. Also, the proposed technique is usa unique, as every united states will have a distinctive colour code for the vehicle's registration code.

The Connected Component Analysis (CCA) technique is used to stumble on the license plate place. CCA is beneficial for simplifying project detection because it splits a binary photograph into a couple of elements primarily based on their connectivity. Based on the challenge, it could decide on the selection of element detection connectivity the usage of 4-adjoining or eight-adjacent pixel connectivity. Spatial size is a degree of the spatial traits of connected components, such as location, orientation, element, and so forth., and filtering is finished to exclude outlying or needless additives. When component analysis is combined with spatial measurements and filters, registration code detection results are stepped forward. The computerized license plate recognition of motors has turn out to be very vital in our each day lifestyles due to the limitless boom within the number of cars and transportation systems,

which is not possible for people that allows you to handle and manage flawlessly, there are so many examples inclusive of the monitoring of big visitors, tracks. Theft of cars, parking management, policies violated for jogging a purple mild, limits and customs checkpoints. However, this is a totally difficult hassle because of the specific shapes of the plates, distinctive scales, rotations, and choppy lighting conditions within the imaging. This article gives an method that makes use of simple however effective morphological operations, filtering and looking for components connected to neighborhood Indian license plates. The algorithm become tested on 20 examples and confirmed that it can extract letters and numbers from vehicle registration code pics with 90% accuracy for four-wheeled vehicles from unique international locations in special climates. HD license plates also are marked, list them in terms of appearance.

#### **PRE-PROCESSING:**

The preceding photograph is the boundary of picture operations on the ultimate level of abstraction. These operations do now not growth the information of the photo, but lower it if the degree of entropy is the information. The first goal is to improve the records of the pictures which suppresses undesirable artifacts or complements positive capabilities of the snap shots that are essential for similarly processing and analysis. Image pre-processing makes use of photograph redundancy.

Neighboring elements corresponding to one real item have the equal brightness or near price. If the distorted pixel may be extracted from the photo, the rerun can be an average of the next photos. Image pre-processing strategies can be generated based totally on the scale of the pixel community that is used to calculate the brightness of the new pixel. This article will introduce some pixel brightness adjustments and

local preprocessing techniques carried out in Mat Lab.

#### **IMAGE CROPPING AND FILTERING**

The first step in picture preprocessing is picture cropping. Some of the unwanted elements of the picture can be removed and the area of hobby within the image will be the focus. This device affords the user with information about the size of the cropped picture. Mat Lab's picture crop characteristic performs this operation interactively, waiting for the user to draw a square crop with the mouse and it really works with strolling axes. The output photograph is of the identical type as the input photo. Convolution is a fundamental 2D photograph evaluation operation. The new fee given by way of the given pixel is the weighted average estimate of the pixel inside the place of the middle of the pixel  $k$ .

#### **INTENSITY ADJUSTEMENT AND HISTOGRAM EQUALIZATION**

The grayscale transformation  $T$  of the unique brightness  $p$  from the scale  $[p_0, p_k]$  to the brightness  $q$  from the brand new scale  $[q_0, q_k]$  is given by way of  $q = T(p)$ . It does now not rely on the placement of the pixel within the picture. Values underneath  $p_0$  and above  $p_k$  are cut off. A price below  $p_0$  is prescribed at  $q_0$ , and above  $p_k$  is prescribed at  $q_k$ . The alpha argument specifies the shape of the curve that describes the connection among the values within the enter and output photographs. If alpha is much less than 1, the pair is weighted in the direction of brighter output values. If alpha is greater than 1, the mapping is weighted in the direction of lower, darker output values. If the argument is unnoticed, its default price is 1. The portraits controls allow the user to growth and reduce brightness, evaluation, and alpha correction. Another proposed possibility to enhance the evaluation of the picture by means of converting

the values inside the depth of the photograph in order that the histogram of the output image is equal to the histogram equalization method. The description of the location is based on its statistical residences at gray level. The histogram suggests the frequency of the brightness cost in the picture. An photo with  $n$  grey levels is represented via one size of  $n$  elements.

### **BRIGHTNESS THRESHOLDING**

Threshold clarity is the level of command in attempting to find applicable records. A grayscale photo often incorporates most effective levels of meaningful statistics: the history discipline representing the gadgets of hobby, and the historical past area against which it stands inside the foreground.

### **CLEARRING AREAS OF A BINARY IMAGE**

If the deformation is the anticipated shape and length of the boundary and the complete area, while the object of the image is separated from its angle, it may be partly overcome. Typically, a small polygonal mask located close to and outside the vicinity is added to the bright area of the photo with a similar brightness place.

This mask can alternate the form of the photo items and offers separation of the images from each other and from the target photo. This operation is done interactively via adding vertices to the polygon. When the tip of the polygon is over the white place of the image, the saturation begins, which is displayed in black. The generated saturation mask is a boolean, and the enter image is a boolean matrix. When the usage of logical AND running with logical arguments, the output photograph is also received as a logical array. If the white areas on the front of the photo represent a black history, you may delete complete gadgets or elements of them. In the equal manner, unique use

requirements for the dimensions and form of the observed object can be applied.

### **DETECTING EDGES**

The side elements are in which the feature of the photograph depth modifications dramatically. Edge detectors are styles of local image preprocessing techniques used to hit upon changes within the brightness characteristic. The photograph characteristic depends on two variables, coordinates within the photo aircraft. The authors describe the margins as partial derivations. A alternate in the photo characteristic can be defined by means of a slope pointing in the path of maximum increase of the image characteristic. An facet vector is a variable with additives: value and path. The margin cost is the slope fee.

### **ILLUSTRATION**

Image pre-processing is performed in Mat Lab. The following discern suggests the results of numerous previous moves. The enter picture is a Doppler photograph of a blood vessel. First, you may crop and dispose of the essential elements. The photograph is of interest in the recognition area

### **FILTER FUNCTION:**

Image filtering is a process that refines the image and lets in sure information to be selectively highlighted. There are several techniques available and the satisfactory alternatives depend on the photograph and the way it is going to be used. Both analog and virtual imaging may require filtering to become with a useful and attractive result. This is a normal a part of the modifying manner that may be used to put together pictures for distribution. In the case of film images, while the photographer develops the pictures, it's far vital to use filtering to obtain the desired consequences. You can add filters to the magnifier to improve image excellent or to

broaden things like black and white prints from shade negatives. A charge tag can run numerous assessments to find the most suitable one.

Film photographers can use filters when modifying images, for instance, to increase comparison. A filter out can trade the wavelength of mild because it passes through the magnifier, thereby converting the exposure and the ensuing photo. There are lots of conventional cables for picture amplifiers and commercial cameras.

Digital filtering offers some superior image processing alternatives past the basic filters utilized in image development. One common utility of filtering in picture processing is blur removal. Pictures may be frightening due to the deformity of the record, the transferring elements inside the frame, and other matters. A price filtering algorithm may be used to selectively seize factors and clean the photograph. More state-of-the-art filters can restore in part corrupted snap shots, commonly, by means of using the statistics available to estimate the lacking photograph.

Another application of filtering in image processing is in photo processing, where technicians want to spotlight sure items of hobby in an picture. For example, astronomers can skip photographs via filters to slender down decided on data from particular wavelengths. This permits other data to stand out within the photograph. Filters also can put off noise, which include noise, from photos to lead them to cleanser and crisper, even supposing they do not scare the energy.

The software permits you to finish the most complicated photograph filtering techniques. Many come with items that people can use for simple capabilities like including gentle filters to snap shots, or customers can also design their very own filters through coding to create their

personal version for a selected need or task. This may additionally require superior programming talents and a complicated knowledge of images.

### **MATLAB IMPLEMENTATION:**

This bankruptcy describes the implementation of the license plate detection set of rules the use of MATLAB. MATLAB is a totally powerful software program tool for fixing troubles that require big calculations. It offers simple and fast implementation of algorithms compared to C and C++. MATLAB's most important function is that it contains a rich library of features for picture processing and statistics evaluation. This makes MATLAB a perfect device for faster implementation and validation of any algorithm before it's miles honestly implemented on actual hardware. Sometimes debugging bugs on real hardware is very laborious. MATLAB gives smooth get entry to to debug and attach mistakes in any algorithm. In addition, MATLAB has many functions inclusive of workspace, graph, imread, imhist, imshow, and many others. It consists of records for data analysis and processing, making it advanced to other programming languages consisting of C and C++. With the above benefits, the author of this venture has initially pursued an set of rules for license plates.

Discovery with MATLAB. The set of rules changed into to start with utilized in various integrated functions and applied numerous consumer-defined subroutines associated with picture processing. After the set of rules was advanced, it was tested with more than one input pictures which might be contained in car license plates. The enter images contained horizontal plates and additionally at some attitude to the horizontal axis. After Algorithm 17 turned into completely examined, the built-

in features in MATLAB have been changed with consumer-described functions.

Converting a coloration photograph to a grey photo;

The set of rules described right here is impartial of the color of the image and in particular is predicated on the grey level of the picture to procedure and extract the important records. Color combinations like crimson, inexperienced, and blue aren't used on this algorithm.

So if the enter picture is a 3-D color image represented in an array

MATLAB converts it to a 2D grayscale photograph earlier than similarly processing. Sample original input image and grayscale image



## CONCLUSION AND FUTURE SCOPE:

In this work, the methodologies and algorithms proposed in the literature on automobiles and license plates are reviewed. Due to the lack of

training of a plate number reputation system that meets our necessities, we're trying to installation a education machine to recognize the number plate. Pattern matching to the impact of license plates received from static pics and a median accuracy of 80.Eight% turned into done. This accuracy can be greatly progressed through setting the digicam to the great shot and using layers of neural networks. The implementation of the proposed gadget may be extended to apprehend the license plates of several cars in a single photograph frame the usage of a couple of genetic algorithms. Also, an urban version of this machine may be implemented via taking inputs from stay video and choosing the satisfactory car body for vehicle kind category and license recognition.

## REFERENCES

- Anishiya.P, Prof. S. Mary Joans," Number Plate Recognition for Indian
- Cars Using Morphological Dilation and Erosion with the Aid of Ocrs." International Conference on Information and Network Technology, Vol.4, 2011
- BENSON, Jonathan P., et al. Car-park management using wireless sensor networks. In: Local Computer Networks, Proceedings 2006 31st IEEE Conference on. IEEE, 2006. p. 588-595.
- Diang,T. M.Mekonnen, T. E.Merkebu,A Gebrehiwot," Car Plate Recognition System." Fifth International Conference on Intelligent Network and Intelligent System 2012.
- LEE, Sangwon; YOON, Dukhee; GHOSH, Amitabha. Intelligent parking lot application using wireless sensor networks. In: Collaborative Technologies and Systems, 2008. CTS 2008. International Symposium on. IEEE, 2008. p. 48-57.

- Kamat and S. Ganesan, “An efficient implementation of the Hough transform for detecting vehicle license plates using DSPs,” in Proc.Real- Time Tech. Applicat. Symp., 1995, pp. 58–59
- Kasaei S , S. M. Kasaei, " Extraction and Recognition of the Vehicle License Plate for Passing Under Outside Environment."
- IEEE 2011
- SRIKANTH, S. V., et al. Design and implementation of a prototype smart PARKing (SPARK) system using wireless sensor networks. In: Advanced Information Networking and Applications Workshops, 2009.
- Sourav Roy, Amitava Choudhury, Joydeep Mukherjee,,” An Approach towards Detection of Indian Number Plate from Vehicle” International journal of Innovative Technology & Exploring Engineering(IJITEE)Volume- 2, Issue4, March 2013
- Shan. Du, Mohamed. Shehata , Wael. Badawy ,,”Automatic License Plate Recognition(ALPR):A State-of –the-Art Review” IEEE Vol. 23, No.2, June 2013.
- TANG, Vanessa WS; ZHENG, Yuan; CAO, Jiannong. An intelligent car park management system based on wireless sensor networks. In: Pervasive Computing and Applications, 2006 1<sup>st</sup> International Symposium on. IEEE, 2006. p.