PROJECT REPORT ON

Parking Management and Security System

Submitted to the University of Calicut in partial fulfillment of the requirement for the award of the degree of

BACHELOR OF COMPUTER APPLICATION
OF
UNIVERSITY OF CALICUT

2020-2023

Done By

Alwin k j
Adithya Rajan
Muhammed Suhail A.P

Under the guidance of

Mrs. INDU MANOJ

DEPARTMENT OF COMPUTER APPLICATION

MAR OSTHATHEOS COLLEGE, PERUMPILAVU

THRISSUR DIST, KERALA

CERTIFICATE

This is to certify that the project work entitled "**Parking Management and Security System**" is a bona fide record of work done by

Alwin k j
Adithya Rajan
Muhammed Suhail A.P

Under the guidance of

Mrs. Indu Manoj

SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENT

FOR

THE AWARD OF THE DEGREE OF BACHELOR OF COMPUTER APPLICATION OF UNIVERSITY OF CALICUT

Head OF Department

Faculty Guide

Submitted for the Viva-voce Examination held on……..

Internal Examiner

External Examiner

DECLARATION

We hereby declare that this project work entitled "**Parking Management and Security System**" submitted to the University of Calicut in partial fulfillment of the requirements for the award of a degree in bachelor of computer application is a record of original work done by us, under the guidance of **Mrs. Indu Manoj** assistant professor of computer science, Mar Osthatheos College Perumpilavu.

Alwin k j

Adithya Rajan

Muhammed Suhail A.P

Place: Perumpilavu

Date:

ACKNOWLEDGEMENT

We wish to take this opportunity to express our sincere thanks to all Who contributed to this project and helped us at every turn.

First and foremost we like to convey our sincere thanks to **Prof. BABU C. P** Principal of Mar Osthatheos College, Perumpilavu for his help. We would like to express our sincere thanks to **Mrs. ANNU ROSE JOSE & Mrs. PRADEEPA PK** for their assistance and guidance throughout the project. we are indebted to the valuable discussions and helpful counseling.

we deeply express our gratitude to all the staff members of **MAR OSTHATHEOS COLLEGE**, Perumpilavu for the encouragement, we received throughout the project.

This acknowledgment would be incomplete without expressing our wholehearted thanks to our friends who motivated us during our tenure of study, and to our parents for the inspiration and encouragement throughout this project. Above all, we the Almighty for all have done for us ……

Alwin k.j

Adithya Rajan

Muhammed Suhail A.P

SYNOPSIS

Parking management and security system is a management and security software for parking areas and parking lots, its image processing technology is used to identify the license plates, motion detection to detect incoming and outgoing vehicles, using security cameras (CCTV cameras, surveillance cameras), or Ip cameras to source the input video feed, and a local system or device to store data collected.

PMSS plays a major role in the management of parking areas, and the surveillance of vehicles. since every vehicle has a unique number plate so it can be used to identify the number plate of the targeted vehicle. The classification is utilized for displaying available parking spaces for vehicles and recording the vehicle's traffic. The identification is also employed for managing parking facilities, monitoring and analysis of parking time, and security systems such as observation of stolen vehicles and monitoring of unauthorized vehicles entering the parking area.
PMS the system also can send notifications to users in real-time.

CONTENTS

INTRODUCTION

1. Introduction

In today's fast-paced world, the use of vehicles has become increasingly common and vehicles can be found almost everywhere. However, keeping track of vehicles and their records in places like apartments, schools, and private buildings can be a daunting task. To solve this problem, the Parking Management and Security System (PMSS) was introduced. The PMSS utilizes the Automatic Number Plate Recognition System (ANPR) and a Database Management System (DBMS) to efficiently manage and track vehicle information. This system can function independently and requires minimal human intervention. In this article, we will explore the various technologies and systems that make up the architecture of the PMSS.

1.1 Objectives

This project is developed to solve manual parking and its management worked by a security watchman and CCTV cameras. By using the PMS system the owner can fully automate the process such as monitoring vehicles, Recording individual videos & images recording, notifying the owner/admin for unauthorized entry, assigning parking lot numbers to vehicles and managing the parking, and recording data and analyzing later.

1.2 Applications

PMSS has many applications which are listed below □□

- ★ Management of parking areas
- ★ Surveillance of the parking area and surroundings(only if allowed in pmss)
- ★ Recording vehicles traffic details
- ★ Monitoring and analysis of parking time
- ★ Allocation of parking space
- ★ Security
- ★ Monitoring unauthorized vehicles entry
- ★ Observe activities in the parking area
- ★ Telegram notification to the Owner(user)
- ★ Broadcasting Parking lot number to the driver

SYSTEM STUDY AND ANALYSIS

2. System Study & Analysis

Number plate recognition is an essential task in the field of computer vision and image processing. This system aims to identify and localize the license plates of vehicles and extract the text information from them for further processing. The proposed system consists of four main stages, including number plate identification and localization, character segmentation, optical character recognition, and character recognition using K-nearest neighbors (KNN). The system utilizes OpenCV, an open-source computer vision library, for image and video processing.

2.1.1 System Analysis

The proposed system involves the following stages:

- **Number Plate Identification & Localization:**
  In this stage, the visual of the scene is enhanced with image processing and OpenCV to identify and localize the LP of vehicles. This involves detecting the region of interest (ROI) containing the license plate and cropping it for further processing.

- **Character Segmentation:**
  characters are segmented from the detected LP for retaining valuable information in the system so that further processing can take place. This stage involves segmenting each character from the LP

- **Optical Character Recognition (OCR):**
  In this stage, the text is transferred into encoded text information. This involves recognizing each segmented character and converting it into a digital format that can be processed by the computer.

- **Character Recognition using KNN:**
  The encoded text information is fed into the KNN algorithm to recognize characters. KNN is a machine learning algorithm that can be used for classification tasks, such as character recognition. The algorithm works by finding the k-nearest neighbors of a sample and classifying it based on the majority class of its neighbors.

2.1.2 System Design

The proposed system can be designed as follows:

- **Image Acquisition:**
  The system will acquire images of either the front or rear of vehicles with cameras (input sources such as IP cameras or surveillance cameras).

- **Pre-processing:**
  The acquired images will be pre-processed using OpenCV to enhance the visual of the scene and detect the region of interest (ROI) containing the license plate.

- **Character Segmentation:**
  Once the license plate is localized, each character will be segmented from the detected number plate and pre-processed to remove any noise or artifacts.

- **Feature Extraction:**
  In this stage, the segmented characters will be converted into digital format and extracted as features for the character recognition stage.

- **Character Recognition using KNN:**
  The extracted features will be fed into the KNN algorithm to recognize characters.

- **Post-processing:**
  In this stage, the recognized characters will be post-processed to form the complete license plate number.

2.1.3 Risk Assessment

The proposed system may face the following risks and constraints:

- **Image quality:**
  The system's performance may be affected by the quality of the acquired images, such as lighting conditions and camera angles.

- **Character recognition accuracy:**
  The accuracy of character recognition may be affected by factors such as noise, occlusion, and font variations.

- **Processing time:**
  The system's processing time may be a constraint, particularly in real-time applications.

Conclusion:

In conclusion, the proposed parking management and security system involves four main stages, including number plate identification and localization, character segmentation, optical character recognition, and character recognition using KNN. The system utilizes OpenCV for image and video processing and faces potential risks and constraints such as image quality, character recognition accuracy, and processing time. The proposed system has the potential to be used in various applications, such as traffic monitoring and law enforcement

2.2 Feasibility Study:

A feasibility study is conducted to assess the practicality and viability of the project. In the case of the parking management and security system, the following factors are considered:

- ★ Technical Feasibility
- ★ Economic Feasibility
- ★ Operational Feasibility
- ★ Legal and Ethical Feasibility

2.2.1 Technical Feasibility

The technical feasibility of the project is determined by assessing whether the technology and equipment required to develop and operate the system are available and can be integrated. The system requires input sources such as IP cameras or surveillance cameras, image processing tools such as OpenCV, and machine learning algorithms such as KNN. These technologies are readily available and can be integrated to develop the system.

2.2.2 Economic Feasibility

The economic feasibility of the project is assessed by determining the cost of developing and operating the system compared to the benefits it provides. The cost includes the cost of hardware, software, development, maintenance, and training. The benefits include increased efficiency, accuracy, and security. The system can provide significant benefits in terms of reducing the manual effort required for number plate recognition and improving security. However, the initial development cost and ongoing maintenance cost may be significant.

2.2.3 Operational Feasibility

The operational feasibility of the project is assessed by determining whether the system can be operated efficiently and effectively. The system requires trained personnel to operate and maintain it, and it must be integrated with existing surveillance systems. The system can be integrated with existing surveillance systems, and trained personnel can be hired to operate and maintain it.

2.2.4 Legal and Ethical Feasibility

The legal and ethical feasibility of the project is assessed by determining whether the system complies with the relevant laws and regulations and ethical standards. The system must comply with data protection and privacy laws, and the use of the system must be transparent and ethical. The system should be used for legal and legitimate purposes only, and the data collected should be protected.

Conclusion:

The feasibility study indicates that the parking management and security system is technically, economically, operationally, legally, and ethically feasible. The system can provide significant benefits in terms of reducing manual effort, improving security, and increasing efficiency and accuracy. However, the initial development cost and ongoing maintenance cost may be significant, and the system must comply with relevant laws and ethical standards.

2.3 Software Requirement Specification

The software requirement specification outlines the functional and non-functional requirements of the parking management and security system. The system aims to acquire images of either the front or the rear of vehicles with cameras and then use image processing to identify license plates. The system consists of four main stages: Number Plate Identification & Localization, Character Segmentation, OCR (Optical Character Recognition), KNN (K-nearest neighbors) character recognition, PSQL Database management system, and Telegram bot services.

2.3.1 Functional Requirements

- The system shall acquire images of vehicles with cameras.
- The system shall use image processing techniques to identify and locate license plates.
- The system shall segment the characters from the detected number plate.
- The system shall use OCR to transfer text into encoded text information.
- The system shall use the KNN algorithm to recognize characters.
- The system shall provide accurate and reliable results for number plate recognition.
- The system shall be able to operate in real time.
- The system shall be able to handle various types of number plates.
- The system shall provide a user-friendly interface for operating and configuring the system.
- The system shall store the data related to recognized number plates in a database.
- The system sends notifications to the user via the telegram bot service
- The system broadcast the allotted parking lot to the driver

2.3.2 Non-functional Requirements:

- The system shall have a high level of accuracy in number plate recognition.
- The system shall have a low error rate in recognizing characters.
- The system shall have a fast response time for recognizing number plates.
- The system shall be able to handle a large volume of data.
- The system shall be secure and protect the data collected.
- The system shall be scalable and adaptable to changes in the operating environment.
- The system shall be compatible with different camera types and operating systems.
- The system shall be easy to install and maintain.
- The system shall have good performance and reliability.
- The system shall comply with relevant data protection and privacy laws.

2.3.3 Constraints:

- The system must be compatible with existing surveillance systems.
- The system must be cost-effective to develop and operate.
- The system must comply with relevant laws and ethical standards.
- The system must have a high level of accuracy and reliability.

2.3.4 Assumptions:

- The cameras used in the system will have sufficient resolution and quality to capture clear images of license plates.
- The system will be operated by trained personnel who are familiar with the software and hardware used.
- The system will be used for legal and legitimate purposes only and will not violate any laws or ethical standards.

Conclusion:

The software requirement specification outlines the functional and non-functional requirements of the parking management and security system. The system shall acquire images of vehicles with cameras, use image processing techniques to identify and locate license plates, segment characters, use OCR to transfer text, and recognize characters using the KNN algorithm. The system shall have a high level of accuracy, and be fast, reliable, and secure. The system shall comply with relevant laws and ethical standards and be cost-effective to develop and operate. The system must be compatible with existing surveillance systems and be operated by trained personnel. The system will be used for legal and legitimate purposes only.

SOFTWARE CONFIGURATION

3.1 System Requirement

Processor	intel i5 or AMD Ryzen 5
RAM	16 GB
Storage	500 GB hard disk space
Graphics Card	2GB VRAM minimum
Operating System	Windows 10 or higher operating system
Network Requirement	LAN/PAN/Local-Hotspot/Local Network HUB *(PMSS must be in the same network)
Camera(Input source)	IP-Camera or any other surveillance camera with IP-Cam support in same network
Environments	Python 3.7 or above
Android Device	Android Mobile phone or Smart TV of Android version 7.0 or above

3.2 Hardware Requirements for PMS System:

- **Cameras**: The system requires cameras to capture images of vehicles. The cameras can be IP cameras or surveillance cameras, and they must be compatible with the system software. The camera should have a high resolution to ensure that the license plate numbers are clear and legible.

- **Computer**: The system requires a computer to run the software. The computer should have a minimum of 8GB RAM and 500GB hard disk space to store the data collected. The processor should be an Intel Core i5 or higher, and the operating system should be Windows 10 or higher.

- **Graphics Card**: The system requires a graphics card to process images and videos. The graphics card should be NVIDIA or AMD, with a minimum of 2GB VRAM.

- **Storage**: The system requires storage to store the data collected. The storage can be internal or external hard drives, and the capacity should be sufficient to store the data collected.

- **Network**: The system requires a network connection to transfer data from the cameras to the computer. The network should be stable and have high bandwidth to ensure that the data transfer is fast and reliable.

- **Power Supply**: The system requires a stable power supply to ensure that it operates without interruptions. The power supply should be reliable and have a backup in case of power failures.

- **Mounting**: The system requires mounting equipment for the cameras. The mounting should be stable and secure to ensure that the cameras capture clear and stable images.

- **Android Device:** The System requires an android smart Tv or mobile phone to broadcast the parking lot number to the vehicle or its driver

LITERATURE REVIEW

4.1 Literature Review of OpenCV

OpenCV (Open Source Computer Vision Library) is an open-source computer vision and machine learning software library. OpenCV was built to provide a common infrastructure for computer vision applications and to accelerate the use of machine perception in commercial products. Being an Apache 2 licensed product, OpenCV makes it easy for businesses to utilize and modify the code.OpenCV is a popular open-source computer vision library that can be used to perform various image and video processing tasks

Key Features

- Real-time face detection and recognition: This project uses OpenCV to detect faces in real-time video streams and then recognize those faces by matching them against a database of known faces.

- Lane detection for self-driving cars: This project uses OpenCV to detect and track the lanes on the road, which is an essential part of building a self-driving car.

- Object detection and tracking: This project uses OpenCV to detect and track objects in real-time video streams, which is useful for surveillance and security applications.

- Handwritten digit recognition: This project uses OpenCV to recognize handwritten digits using machine learning techniques.

- Pedestrian detection: This project uses OpenCV to detect pedestrians in real-time video streams, which is useful for building pedestrian safety systems.

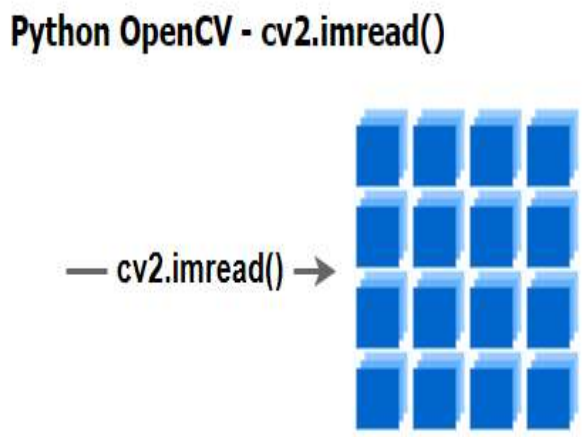
- Optical character recognition (OCR): This project uses OpenCV to recognize characters in images and convert them into text, which is useful for document scanning and data entry tasks.

4.1.1 OpenCV packages

➢ OpenCV cv2. imread()

It is used to read images into a numpy array using the OpenCV library. The array contains pixel-level data. And as per the requirement, we can modify the data of the image at a pixel level by updating the array values.

To read an image in Python using OpenCV, we use cv2.imread() function. imread() returns a 2D or 3D matrix based on the number of color channels present in the image. For a binary or grayscale image, a 2D array is sufficient. But for a colored image, we use a 3D array.



The syntax of cv2.imread() function is given below.

"cv2.imread(/path/to/image, flag)"

For more details, https://bit.ly/3KunDPl

➢ OpenCV cv2. imshow()

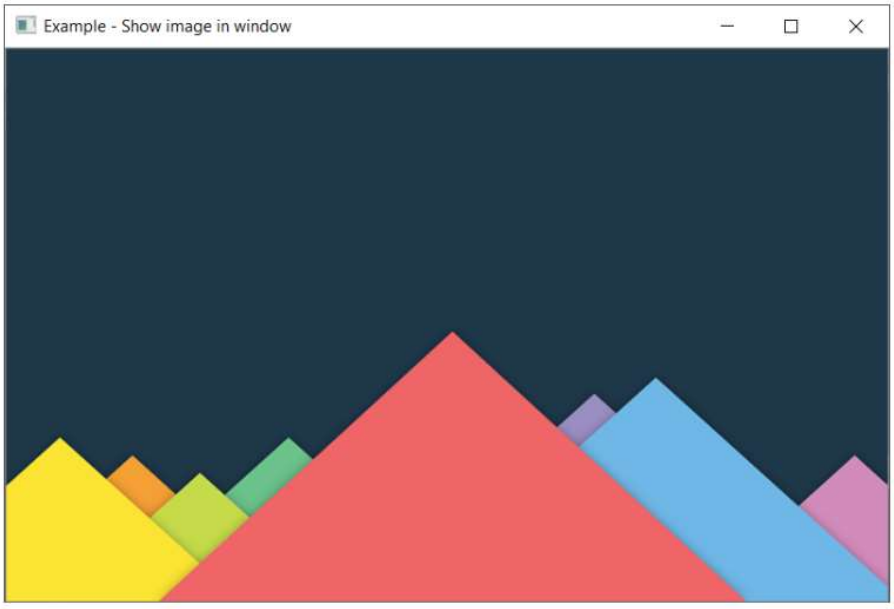
With cv2.imshow we can display an image to the user during the execution of our Python OpenCV application. To display an image using opencv cv2 library, we can use cv2.imshow() function.

The syntax of imshow() function is given below.

"cv2.imshow(window_name, image)"

where window_name is the title of the window in which the image numpy array will be shown. If a window is not created already, a new window will be created to fit the image.

Output of OpenCV imshow()



For more details, https://bit.ly/3It2aop

➢ OpenCV cv2.VideoCapture

With cv2.videoCapture We can record a video from a Camera in Python using OpenCV Library. cv2.videoCapture can use not only an inbuilt webcam but also Ipcamera.

The syntax cv2.VideoCapture function is given below.

" cap = cv2.VideoCapture(0) "

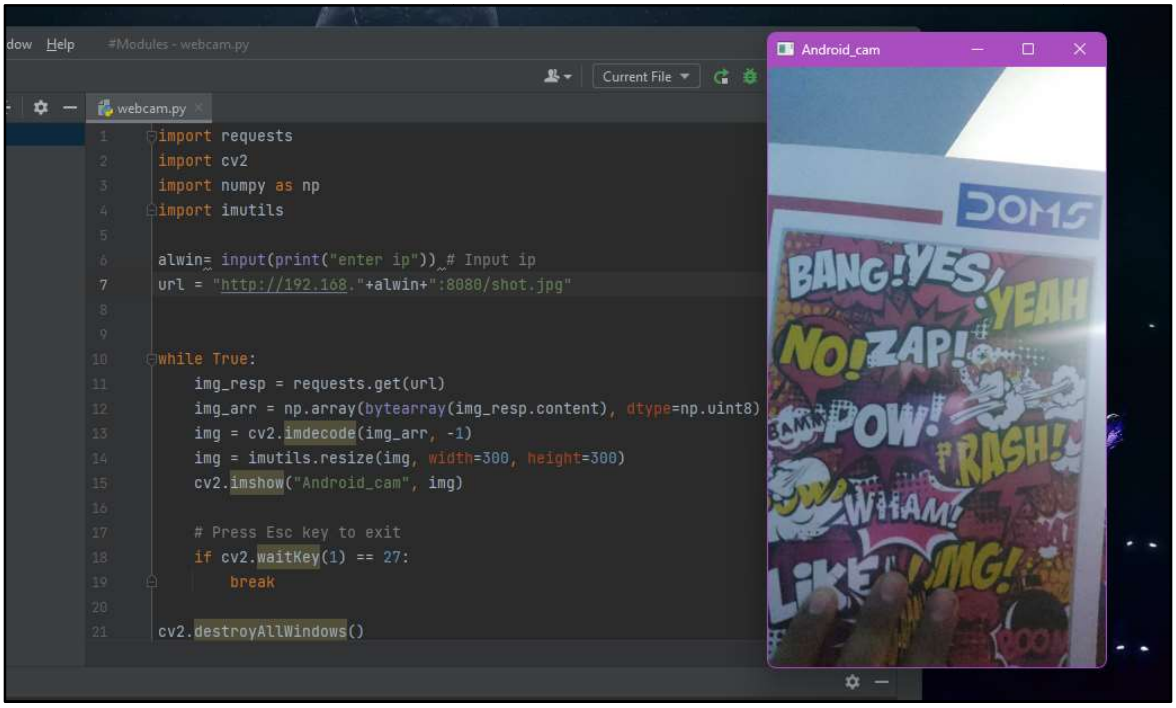
Output of cv2.videoCapture :

For more details: https://bit.ly/3S1looh

4.1.2 OpenCV IP camera Support

IPcamera can run on python by using OpenCV and imutils(a package on python or other programing languages that helps to support image or video input and its processing)with these two components we can source our input device as an ip camera or a surveillance camera(by default the input source will be default web camera).

Preview of IP camera streaming



For more details https://bit.ly/3XzNdVY

4.2 KNN Algorithm

The k-nearest neighbors algorithm, also known as KNN or k-NN, is a non-parametric, supervised learning classifier, which uses proximity to make classifications or predictions about the grouping of an individual data point. While it can be used for either regression or classification problems, it is typically used as a classification algorithm, working off the assumption that similar points can be found near one another.

KNN is commonly used in image recognition tasks. In the context of number plate recognition, KNN can be used for character recognition and plate recognition. Given an image of a character, KNN can classify the character by comparing it to other characters in a training dataset.

Key Features

- Non-parametric: KNN is a non-parametric algorithm, meaning it does not make any assumptions about the underlying distribution of the data. This makes it a versatile algorithm that can work well in a wide range of situations.

- Instance-based learning: KNN is an instance-based learning algorithm, meaning it stores the training instances and makes predictions based on their similarity to new instances.

- Lazy learning: KNN is a lazy learning algorithm, meaning it defers the actual computation of predictions until a new instance needs to be classified. This can be an advantage in some situations because it allows for incremental training and the addition of new data points without the need for retraining the model.

- K-parameter: The value of K, which represents the number of neighbors to consider when making a prediction, is a key parameter in KNN. The optimal value of K depends on the specific problem and can be determined through hyperparameter tuning.

- Distance metric: KNN uses a distance metric to measure the similarity between instances. The most common distance metric used in KNN is Euclidean distance, but other metrics can also be used depending on the problem at hand.

4.3 Machine Learning

It is the process of organizing, analyzing, and modeling data. Regression and classification are used for predictive modeling. In simple words training the machine to learn. First, of we need some data to train the machine after this there is a need to cross-validate the learning and at the end, we need to test whether the results are near to the requirement or not. If the results are near to the requirements it means that our machine is learning well otherwise we need to train the machine again and cross-validate again and this process will repeat till our requirements meet.

For learning machines, there are a number of techniques and many algorithms that we can use.

Here are two types of machine learning techniques:

- Supervised machine learning
- Unsupervised machine learning

4.3.1 Supervised machine learning

Supervised learning (machine learning) takes a known set of input data and known responses to the data and seeks to build a predictor model that generates reasonable predictions for the response to new data.

Suppose you want to predict if someone will have a heart attack within a year. You have a set of data on previous people, including age, weight, height, blood pressure, etc. You know if the previous people had heart attacks within a year of their data measurements So the problem is combining all the existing data into a model that can predict whether a new person will have a heart attack within a year.

Supervised learning splits into two broad categories:

1. **Classification** for responses that can have just a few known values, such as 'true' or 'false'. Classification algorithms apply to nominal, not ordinal response values.
2. **Regression** for responses that are a real number, such as miles per gallon for a particular car.

4.3.2 Unsupervised machine learning

It is a type of machine learning where the model is trained on a dataset without any supervision or labels. Unlike supervised learning, where the model is trained on labeled data to predict outcomes, unsupervised learning is used to find hidden patterns or structures in data.

In unsupervised learning, the model is given a set of data points and is expected to find patterns or groupings within the data. The most common techniques used in unsupervised learning are clustering and dimensionality reduction.

Unsupervised learning is used in a variety of applications, including anomaly detection, market segmentation, and image recognition. An important advantage of unsupervised learning is that it can be used to analyze large datasets without the need for labeled data, which can be time-consuming and expensive to obtain

SYSTEM DESIGN

5.1 Design and Methods used in the proposed system

5.1.1 Proposed system modeling

System modeling is the process of developing abstract models of a system, with each model presenting a different view or perspective of that system. System modeling has generally come to mean representing the system using some kind of graphical notation, which is now almost always based on notations in the Unified Modeling Language (UML). Models are used during the requirements engineering process to help derive the requirements for a system, during the design process to describe the system to engineers implementing the system, and after implementation to document the system's structure and operation,

We may develop different models to represent the system from different perspectives. For example

1. An external perspective is where we model the context or environment of the system.

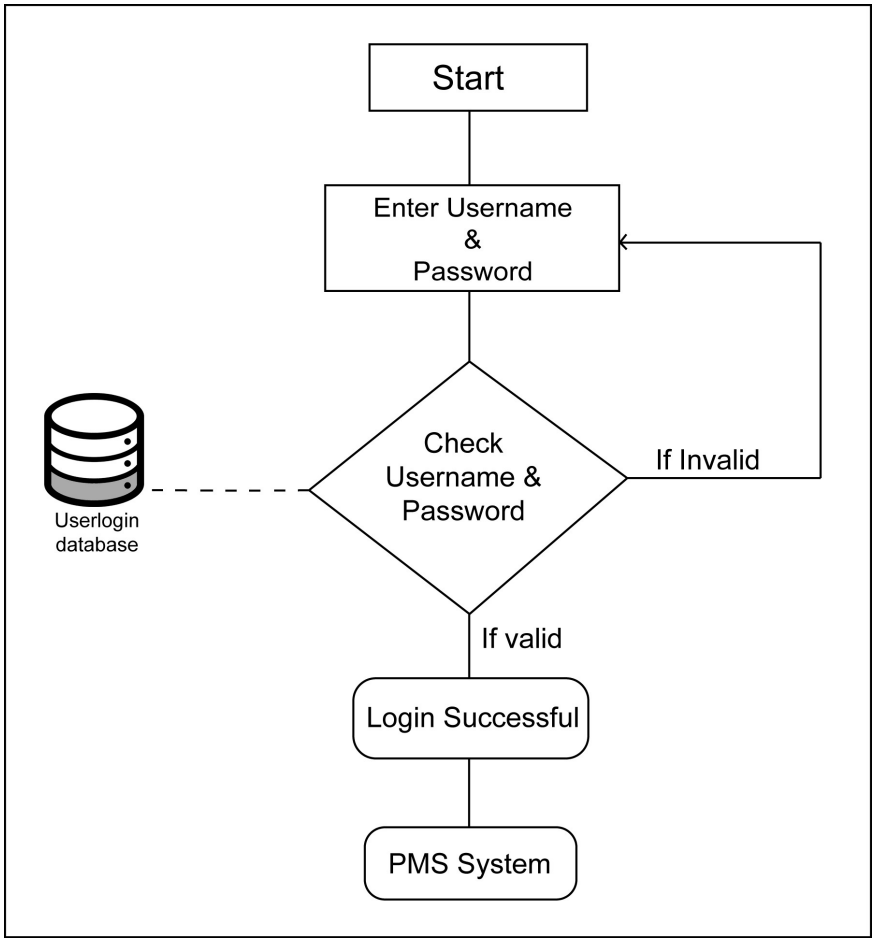
2. An interaction perspective where we model the interactions between a system and its environment or between the components of a system.

3. A structural perspective, where we model the organization of a system or the structure of the data that is processed by the system.

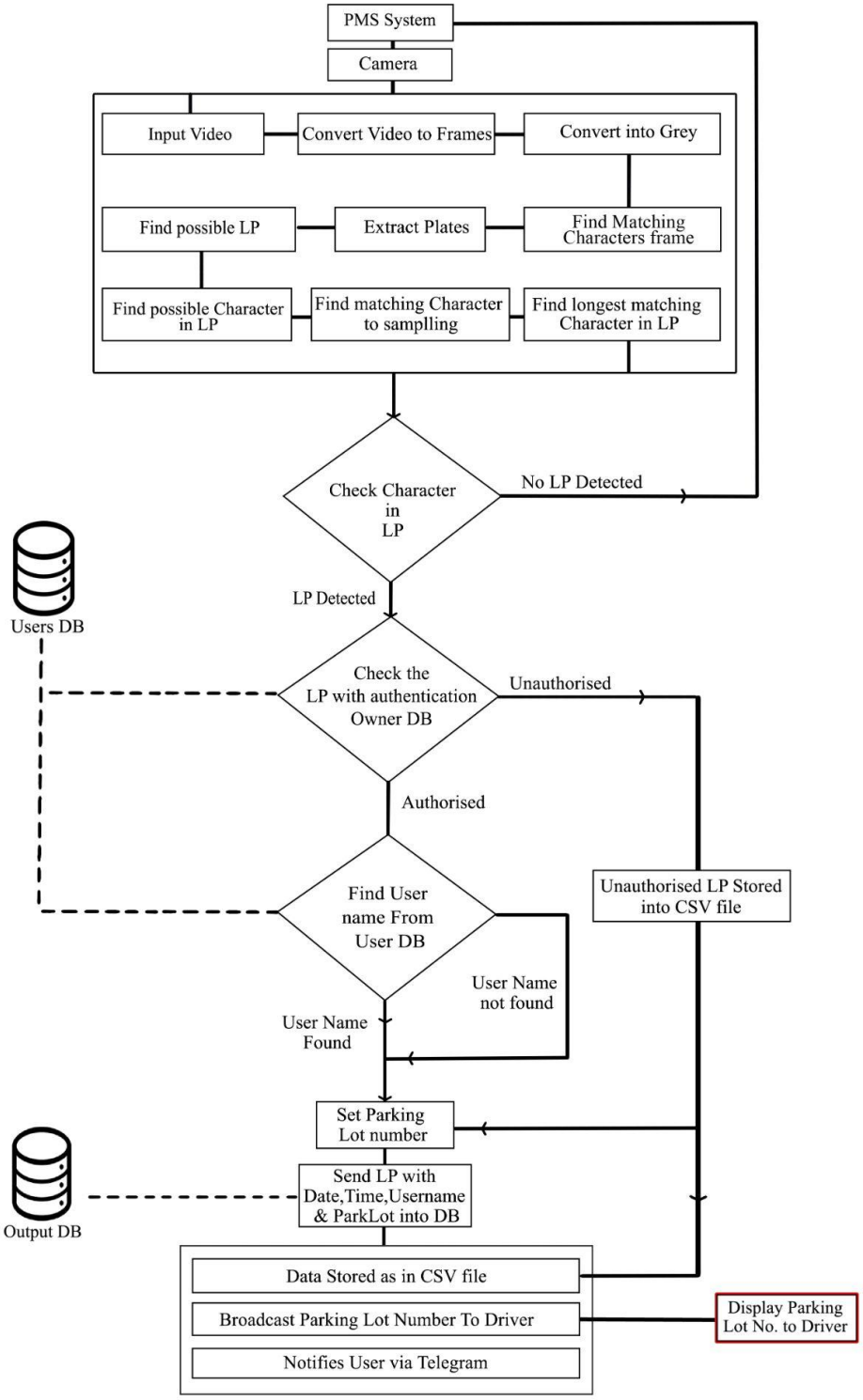
4. A behavioral perspective, where we model the dynamic behavior of the system and how it responds to events.

5.2 Different models describing the proposed system

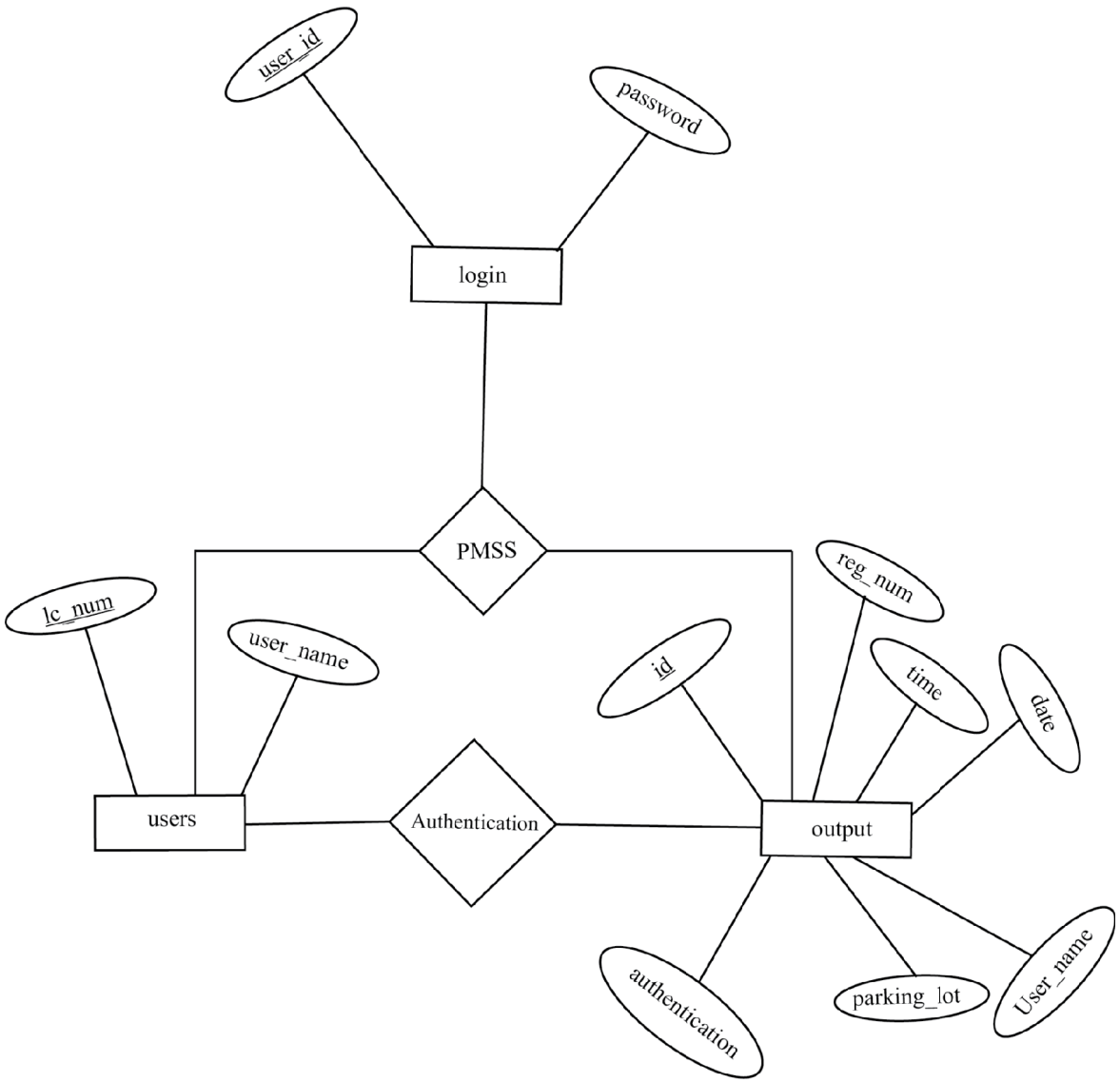
5.2.1 Block Diagram of PMS system User Login



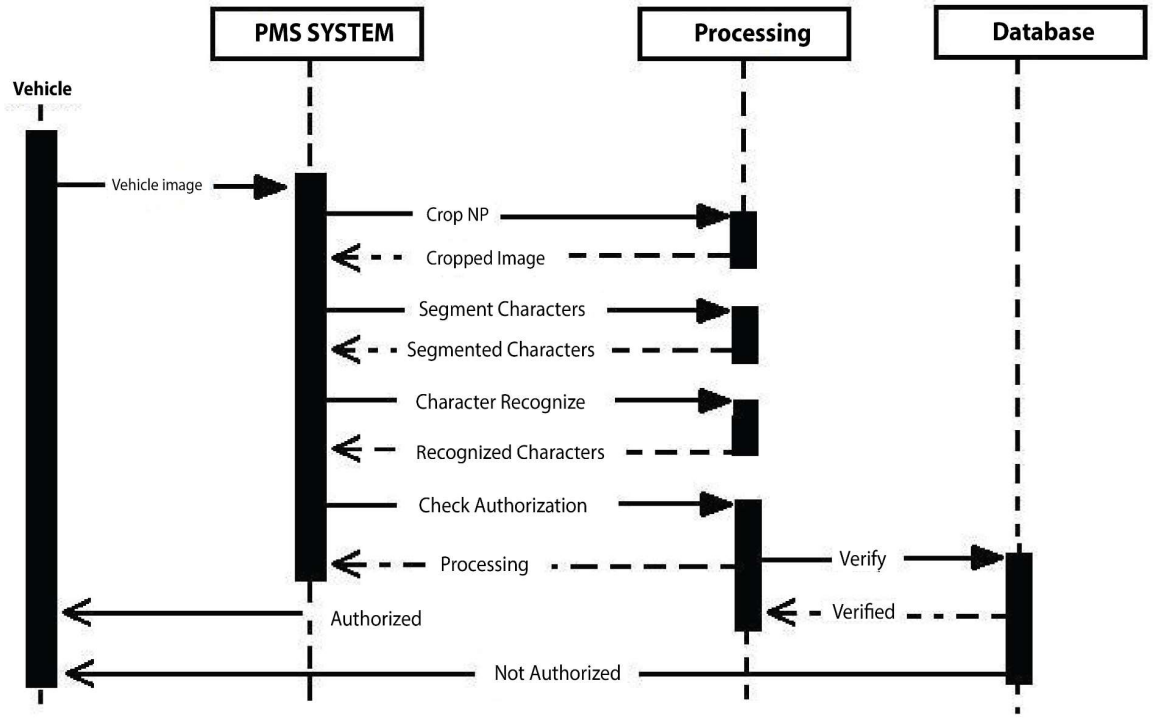
5.2.2 Block Diagram of PMS system



5.2.3 ER Diagram OF PMS system



5.2.4 Sequence Diagram of PMS system



5.3 Design DataBase

Database Design is a collection of processes that facilitate the designing, development, implementation, and maintenance of enterprise data management systems. Properly designed databases are easy to maintain, improve data consistency, and are cost-effective in terms of disk storage space. The database designer decides how the data elements correlate and what data must be stored. The main objectives of database designing are to produce logical and physical design models of the proposed database system.

The logical model concentrates on the data requirements and the data to be stored independent of physical considerations. It does not concern itself with how the data will be stored or where it will be stored physically.
The physical data design model involves translating the logical design of the database onto physical media using hardware resources and software systems such as database management systems (DBMS).

5.3.1 Database Table Design

In the Parking management and security system there are 4 databases

They are

- **Login: Stores user login and password**
- **Users: Stores vehicle owner names and vehicle registration**
- **Output: Store PMS system's Output Data**

□ LOGIN Table design

SL_NO	NAME	TYPE	CONSTRAINTS
1	user_id	Varchar(10)	Primary Key
2	Password	Varchar(10)	not null

□ Users Table design

SL_NO	NAME	TYPE	CONSTRAINTS
1	lc_number	Varchar	Primary Key
2	user_name	Varchar	not null

□Output Table design

SL_NO	NAME	TYPE	CONSTRAINTS
1	id	integer	Primary Key
2	reg_num	Varchar	not null
3	time	Time	not null
4	date	Date	not null
5	authentication	Varchar	not null
6	user_name	Varchar	not null
7	parking_lot	Varchar	not null

SYSTEM IMPLEMENTATION

6.1 Training PMS system

First Training the system to detect character and number plates using the KNN algorithm and OpenCV libraries

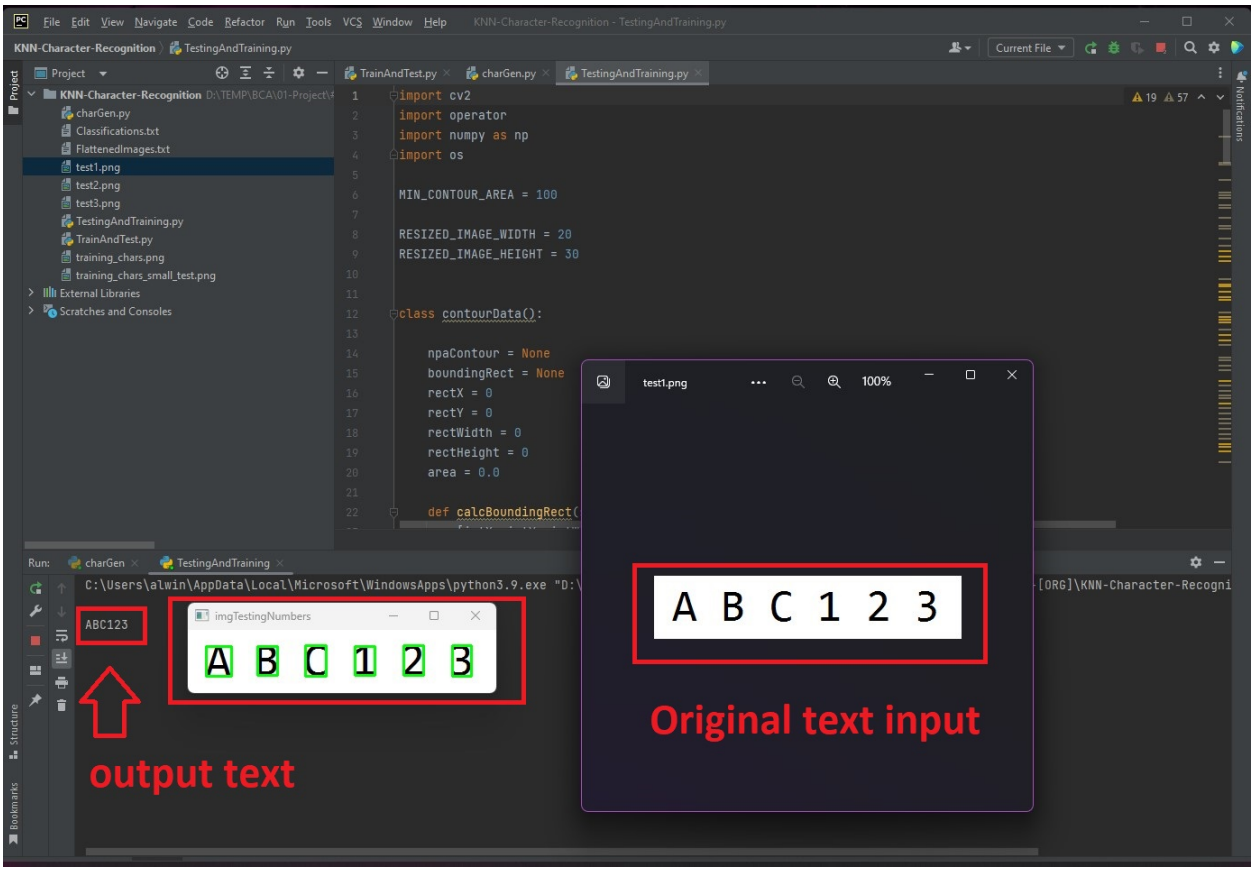


Figure 6.1 Training of neural network

The above Figure output is trained with a set of alphanumeric datasheets

6.1.1 Dataset or Data sample Images used in PMS system

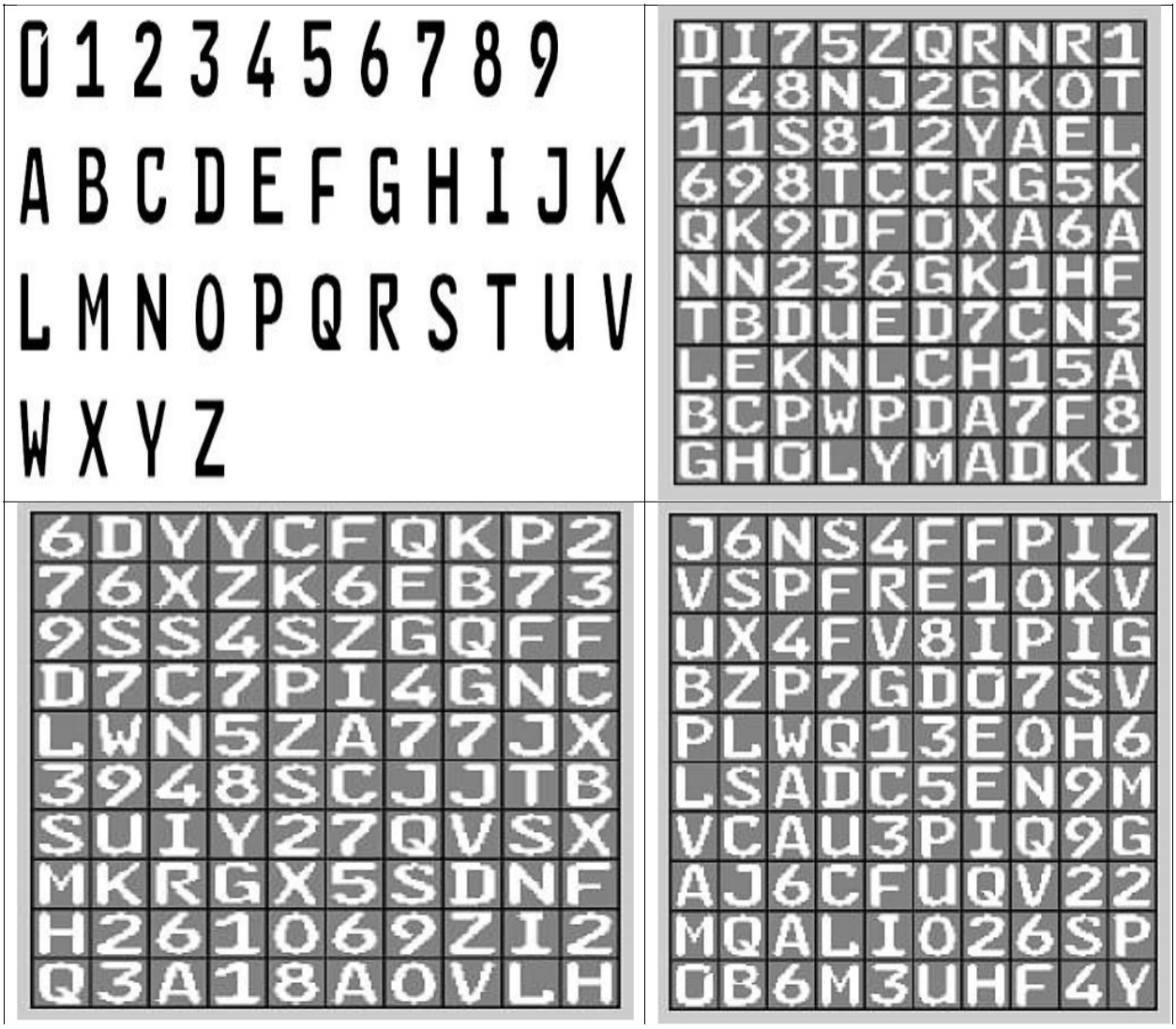


Figure 6.2 Dataset images

6.1.2 Dataset images for training KNN

short datasheet for training "ABC123"

Figure 6.3 KNN training materials

6.2 GUI Of PMS System

User Interface proposed by PMS system

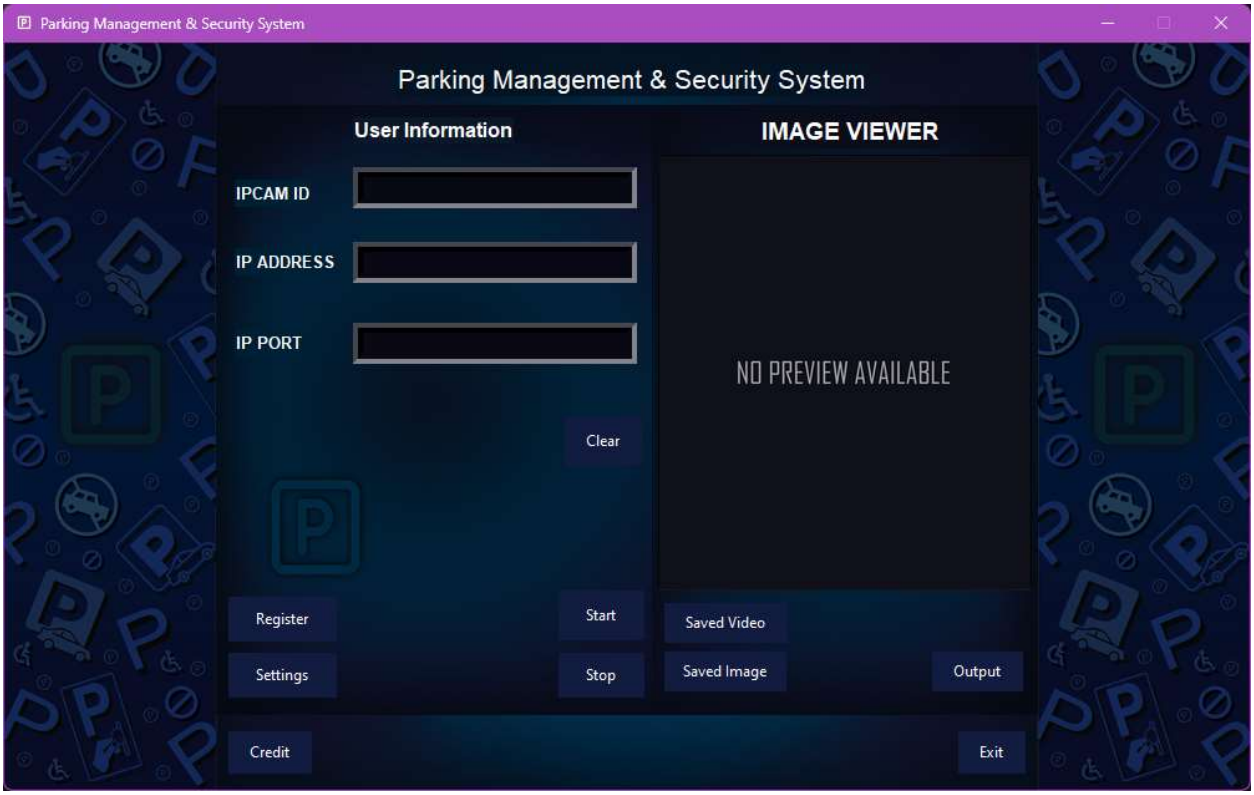


Figure 6.1 GUI of Proposed system

6.2.1 User information panel

By filling the user information tab of the PMS system will add the proposed camera and other details of the camera to the system for further processing.

- IPCAM-ID : Camera name (Must be unique)
- IP-ADDRESS : IP address of the camera
- IP-PORT : Port number of the camera

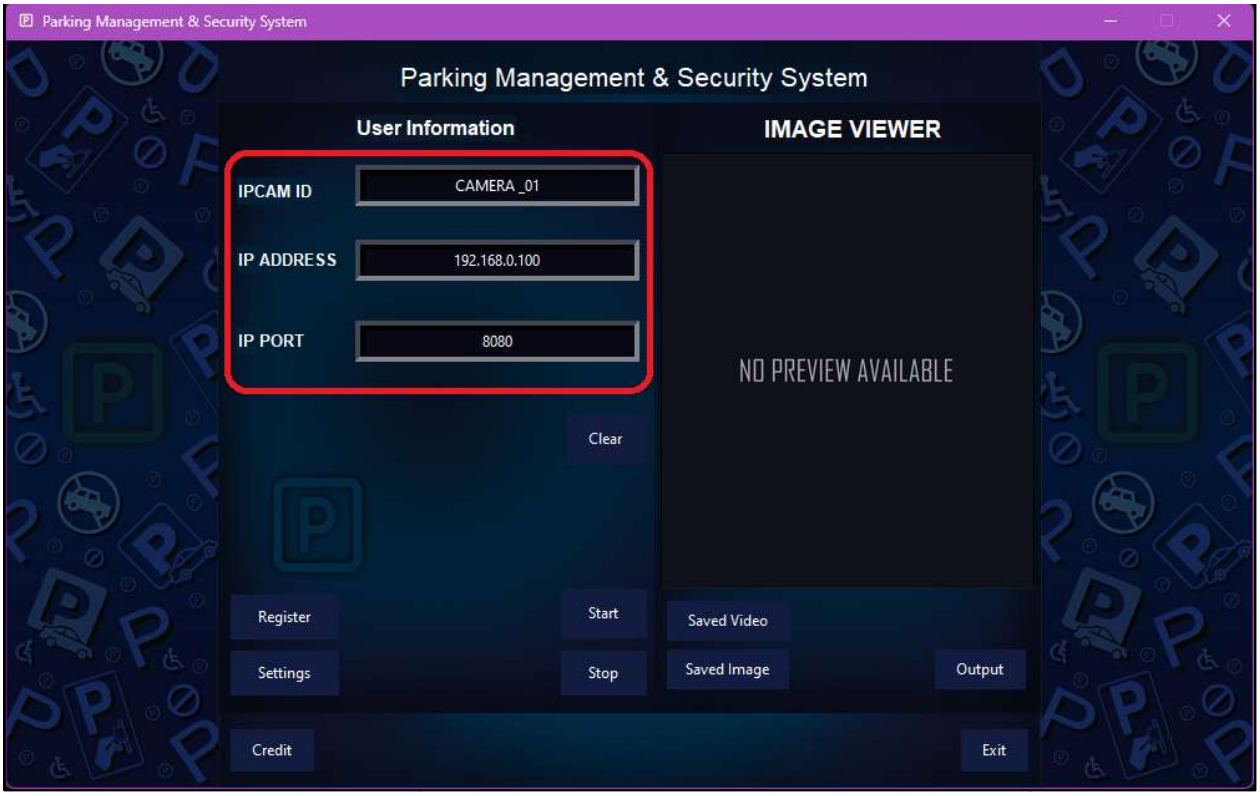


Figure 6.2 GUI user information panel

6.2.2 Start Button

Start Button is used to start PMS System, When the start button is pressed the data in the user information panel data is passed to the PMS system which checks whether the information is correct or not, then the PMS system starts its vehicle detection and management

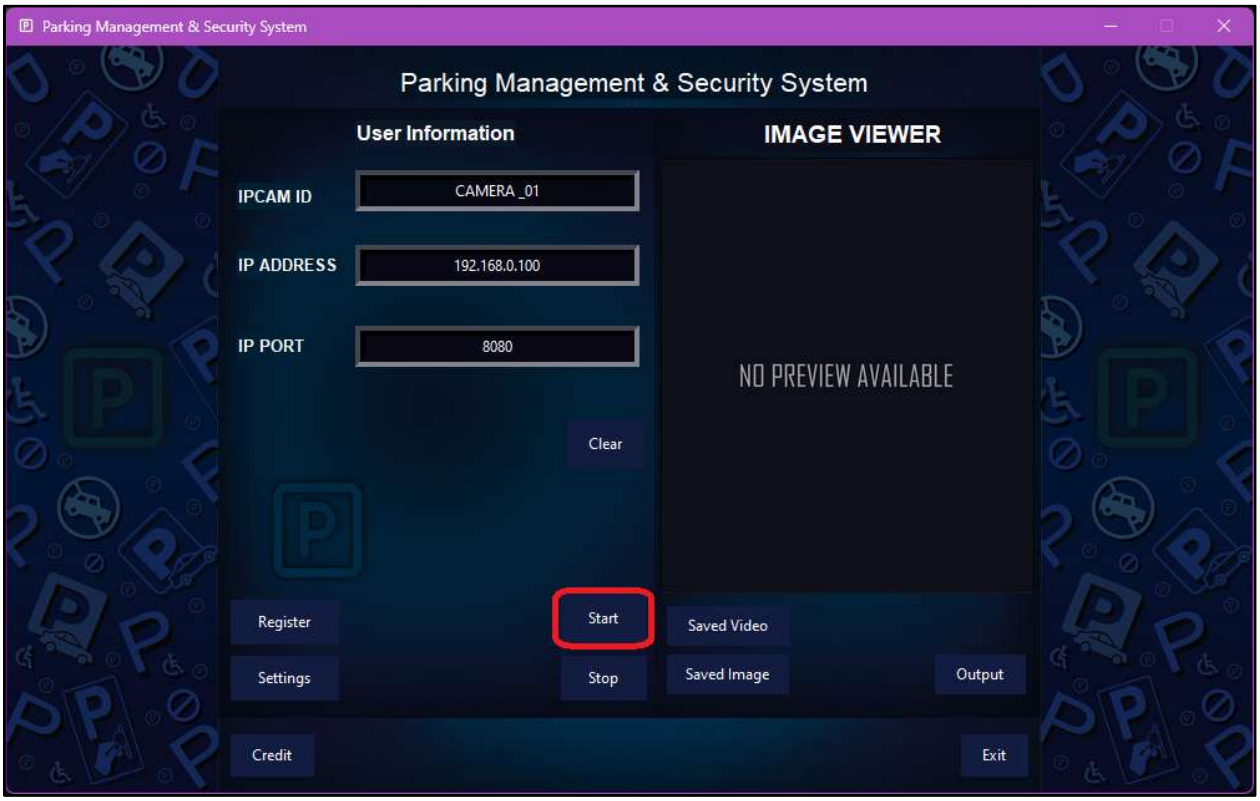


Figure 6.2 Start Button

6.2.3 Stop Button

Stop Button used to stop the running PMS system. When the stop button is pressed the GUI sends a signal to the PMS system to pause or stop its vehicle detection and other services without closing the entire window

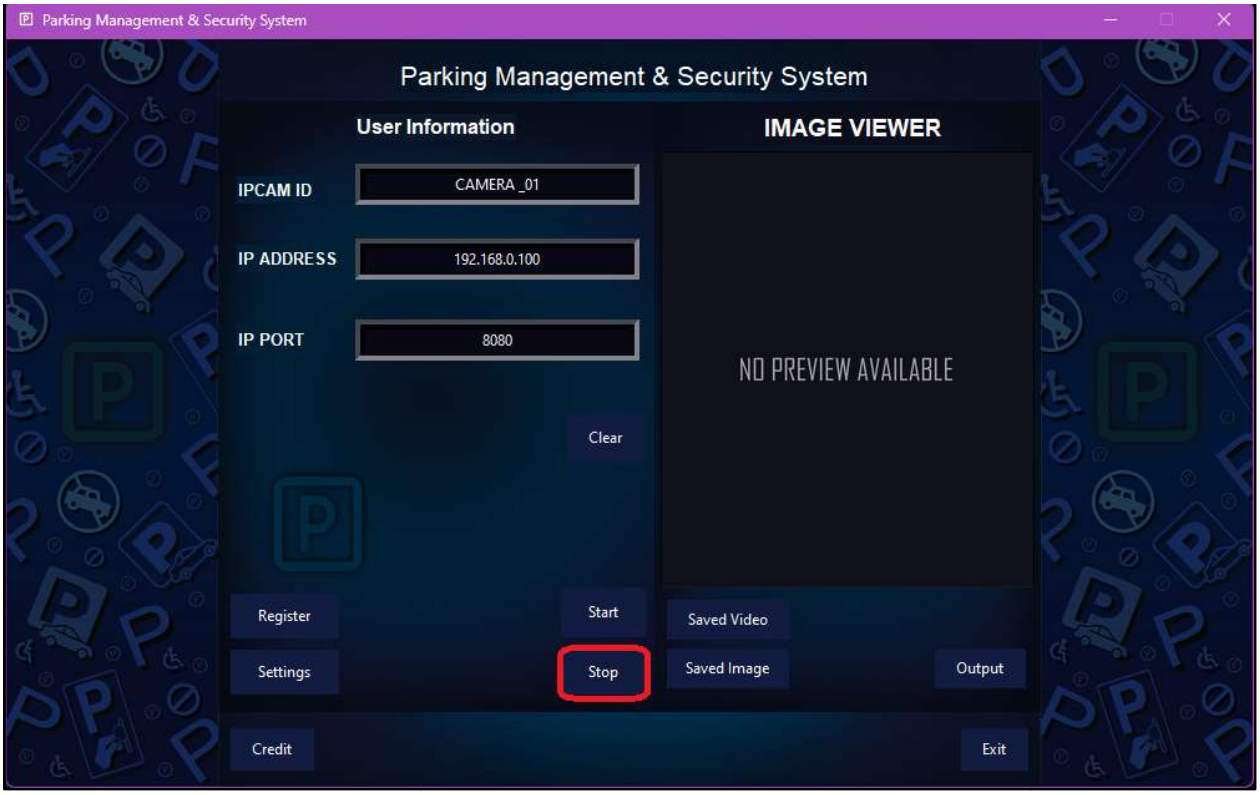


Figure 6.3 Stop Button

6.2.4 Settings Button

Settings button is used mainly for adjusting or tweaking or modifying the camera settings. When the settings button is pressed the PMS system redirects the user to the settings panel of the camera

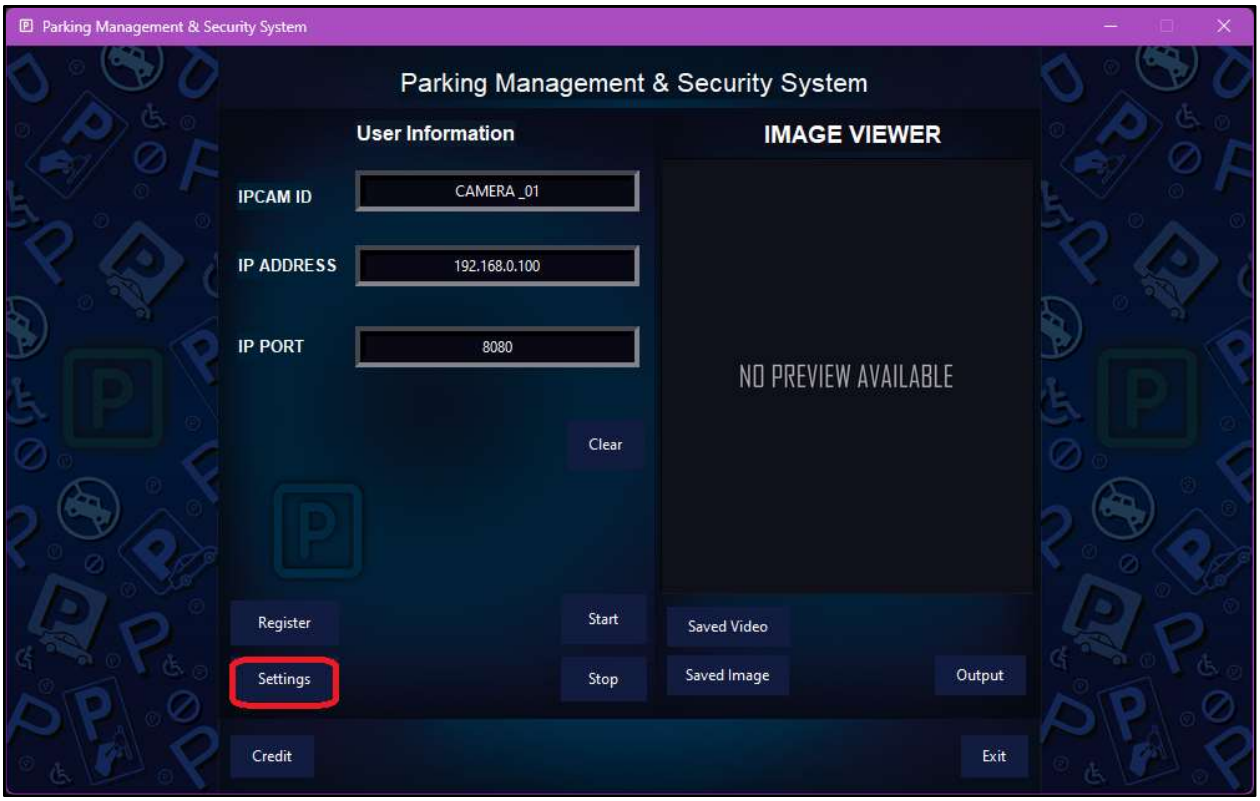


Figure 6.4 Settings button

6.2.5 Camera Settings

Camera Settings is used to modify and tweak the camera for user choice and preferences. It also contains camera extra features such as

- Motion Detection area
- Focus Mode
- Stream quality
- Sensor graph
- Night Vision settings
- Stream Quality and resolution of the output stream, etc.

Figure 6.5 Camera settings

6.2.6 Image preview panel

Image preview panel is used to preview the stored vehicle images and number plate. When the Saved Images button is pressed the image preview panel loads its stored images. it shows the latest image of the vehicles detected by the PMS system

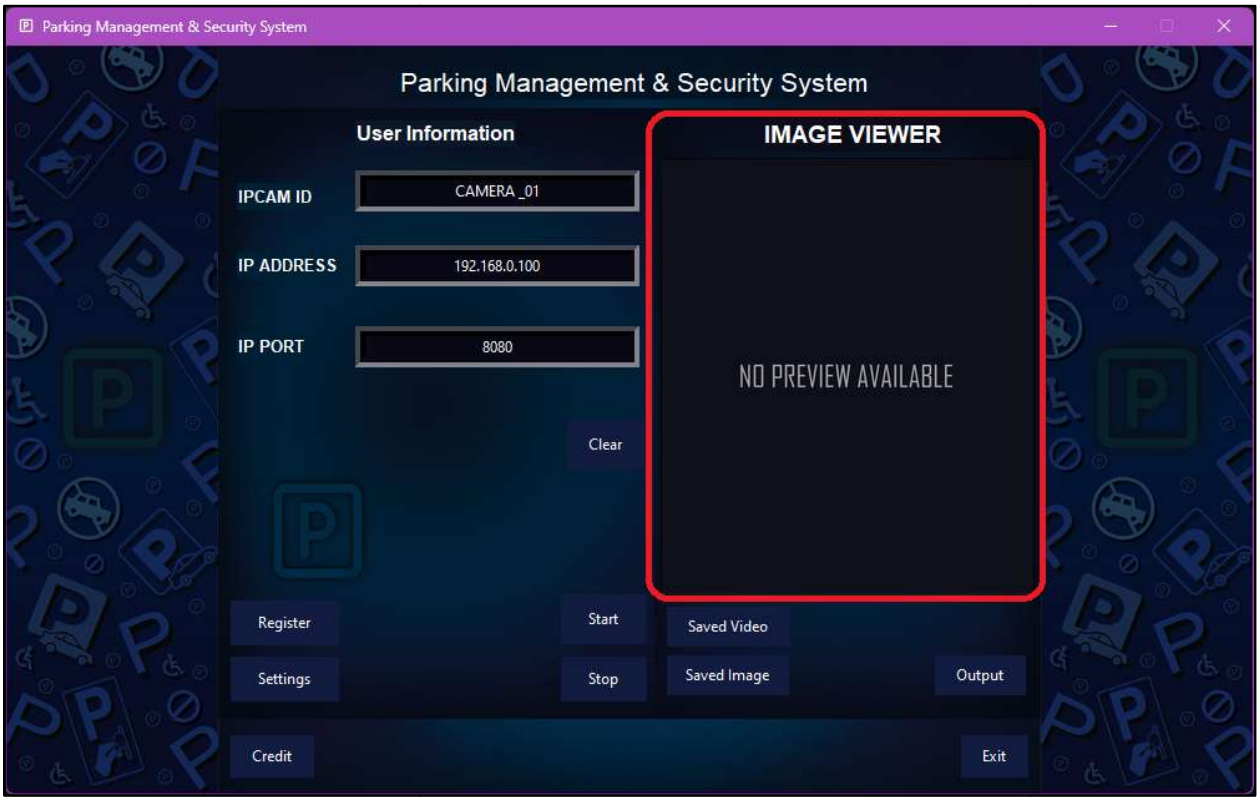


Figure 6.6 Image Viewer panel

6.2.7 Saved Image Button

Saved Image Button is used to load the saved images of vehicles detected by the PMS system into the image preview panel.

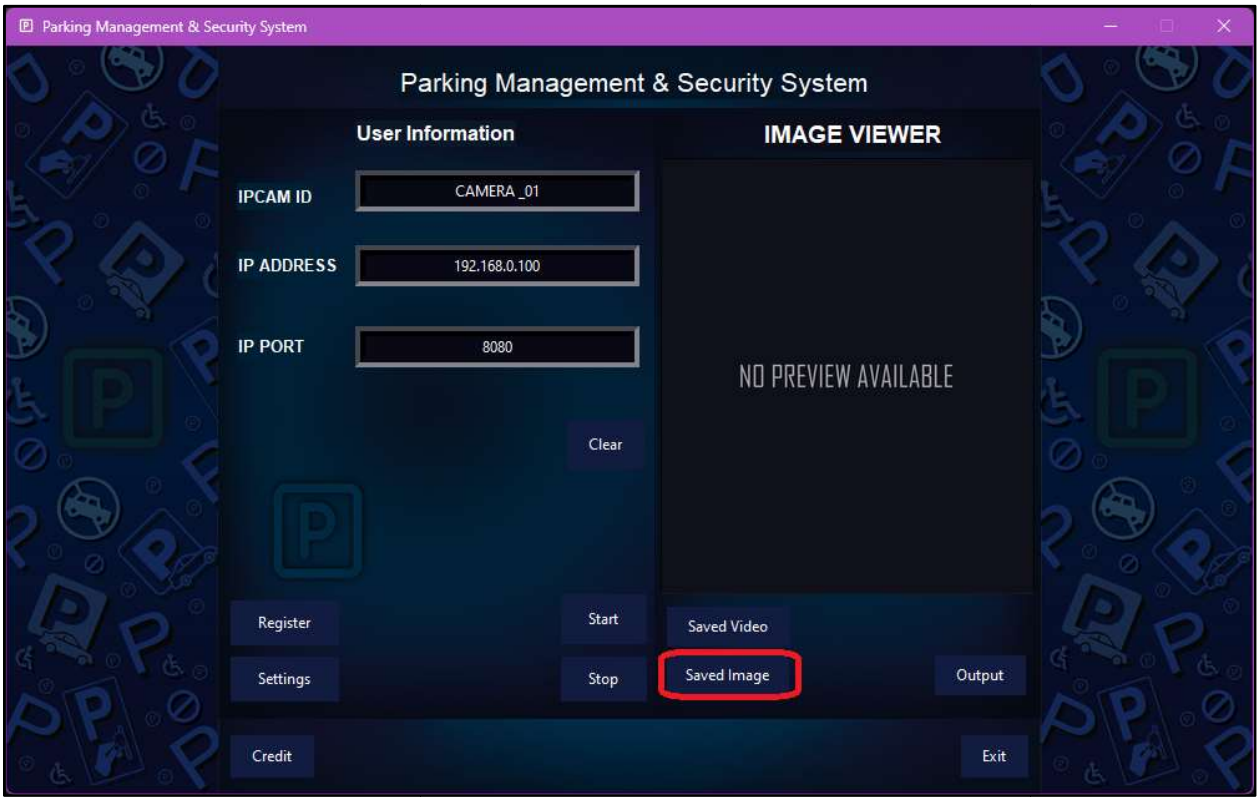


Figure 6.7 Saved image button

6.2.8 Saved VideoButton

Saved Video Button is used to load the saved Videos of vehicles detected by the PMS system. The video player is opened to play the video.

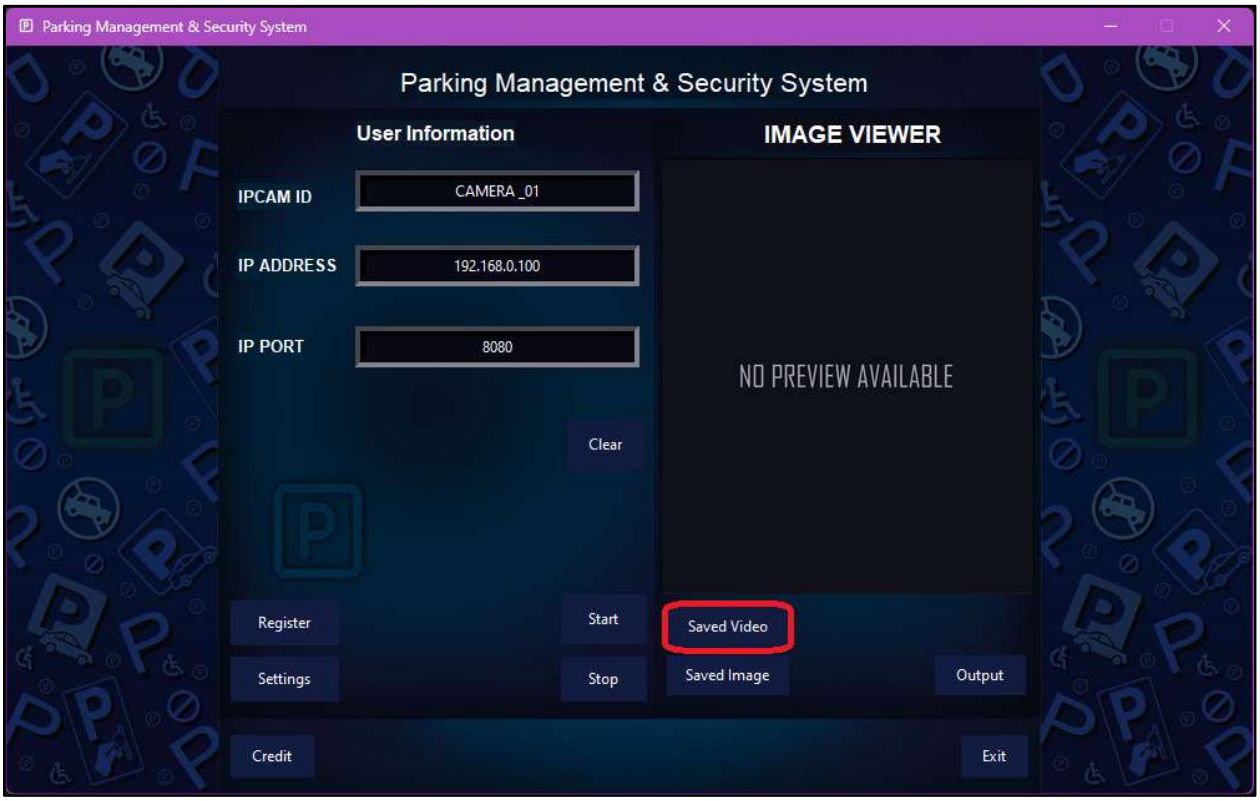


Figure 6.8 Saved Video button

6.2.9 Output Button

Output Button is used to access the data saved by the PMS system. The PMS system stores the data in a CSV(.csv file) file and a Database(.db file). The data saved on the CSV file can access by the user by pressing the output Button

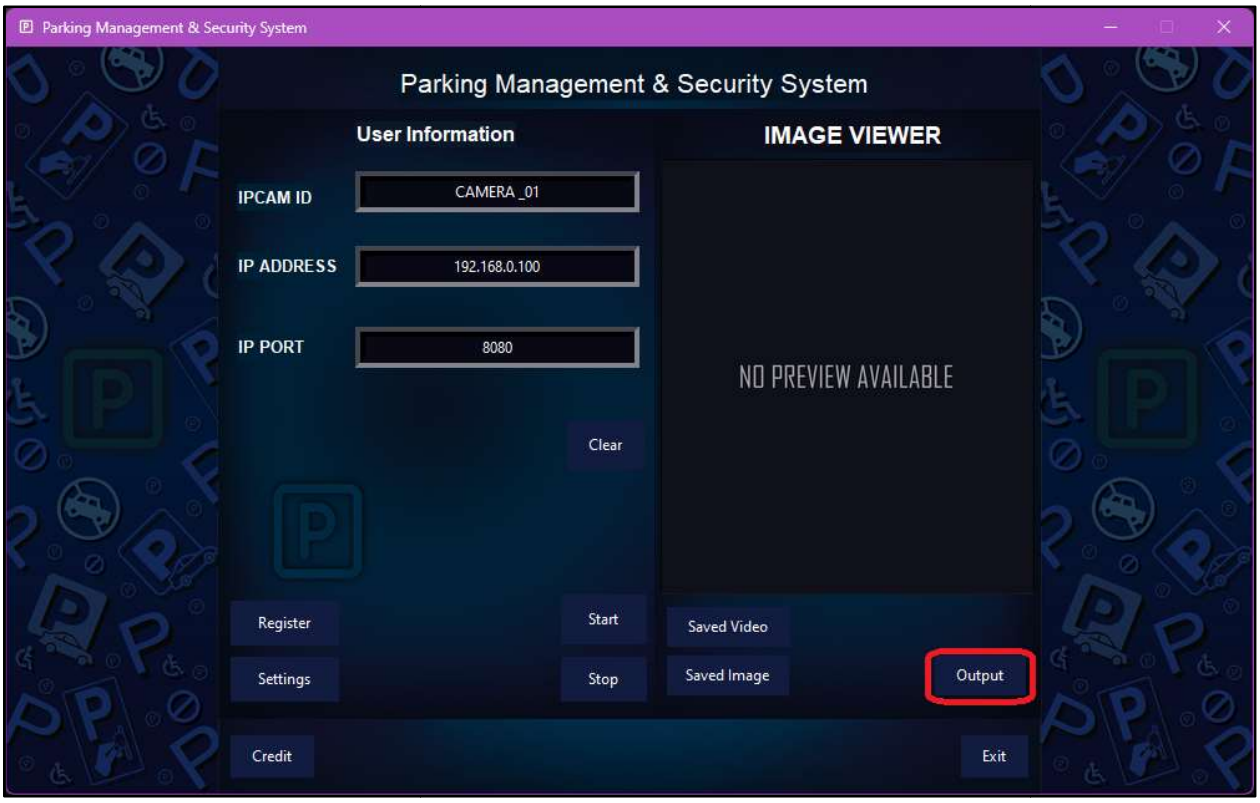


Figure 6.9 Output Button

6.2.10 Exit Button

Exit Button is used to terminate or close the PMS system, when the exit button has been pressed all process is stopped and the PMS system is closed. data saved on the PMS system is saved before so no data loss occurs

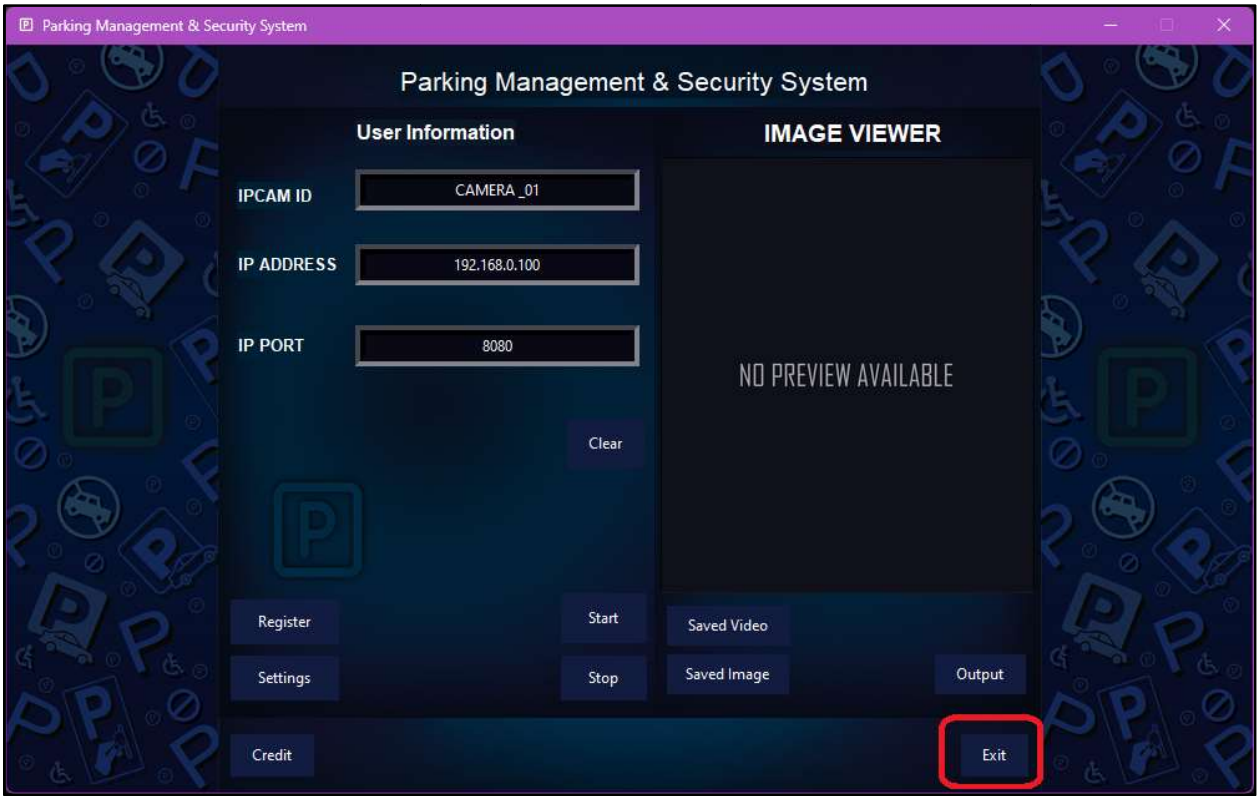


Figure 6.10 Exit Button

6.2.11 Register Button

Register Button is used to open the Register window. Which saves the User/Owner data such as Name/Role and Vehicle registration

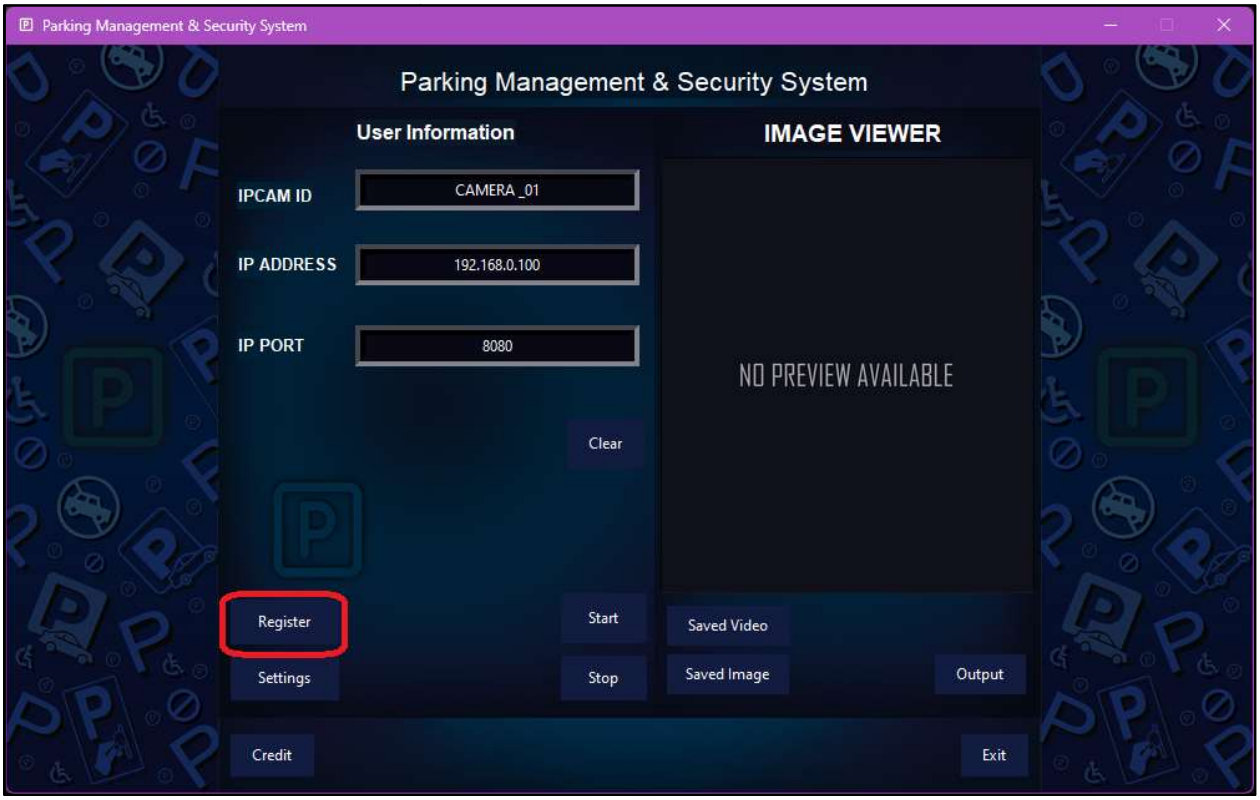


Figure 6.11 Register Button

6.2.12 Register Window

Register Windows is used to Add and Remove Data from the "Users" Database using GUI support

- **ADD Data Button**: Used to Add Username & number plate to Users DB
- **Remove Data Button**: Used to Remove Username & Number Plate form Users DB

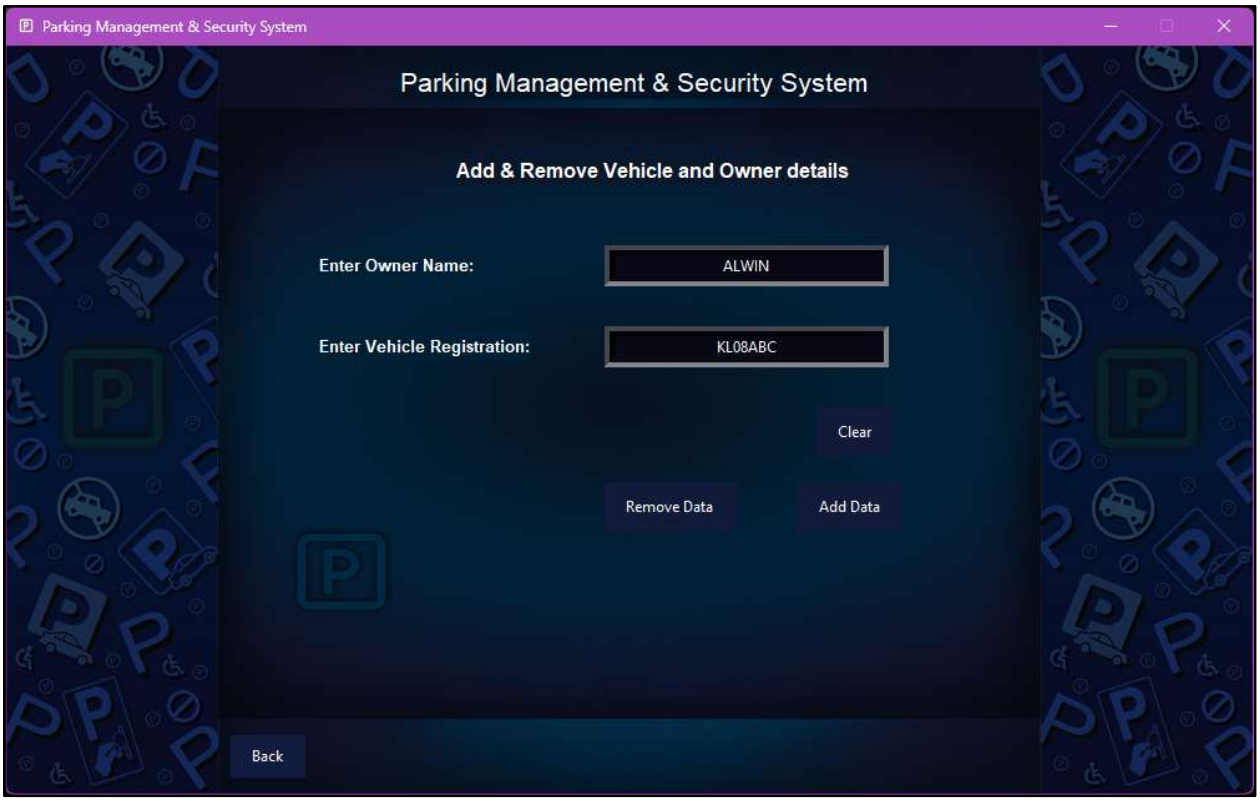


Figure 6.12 Add&Remove Vehicle and Owner details window

6.2.13 PMS SYSTEM Telegram BOT

PMS system sends notifications to users via telegram, the telegram bot PMSS_MSG. PMS system sends the notification to the user's mobile devices for alerts of the parking lot and its security notifications in real-time.

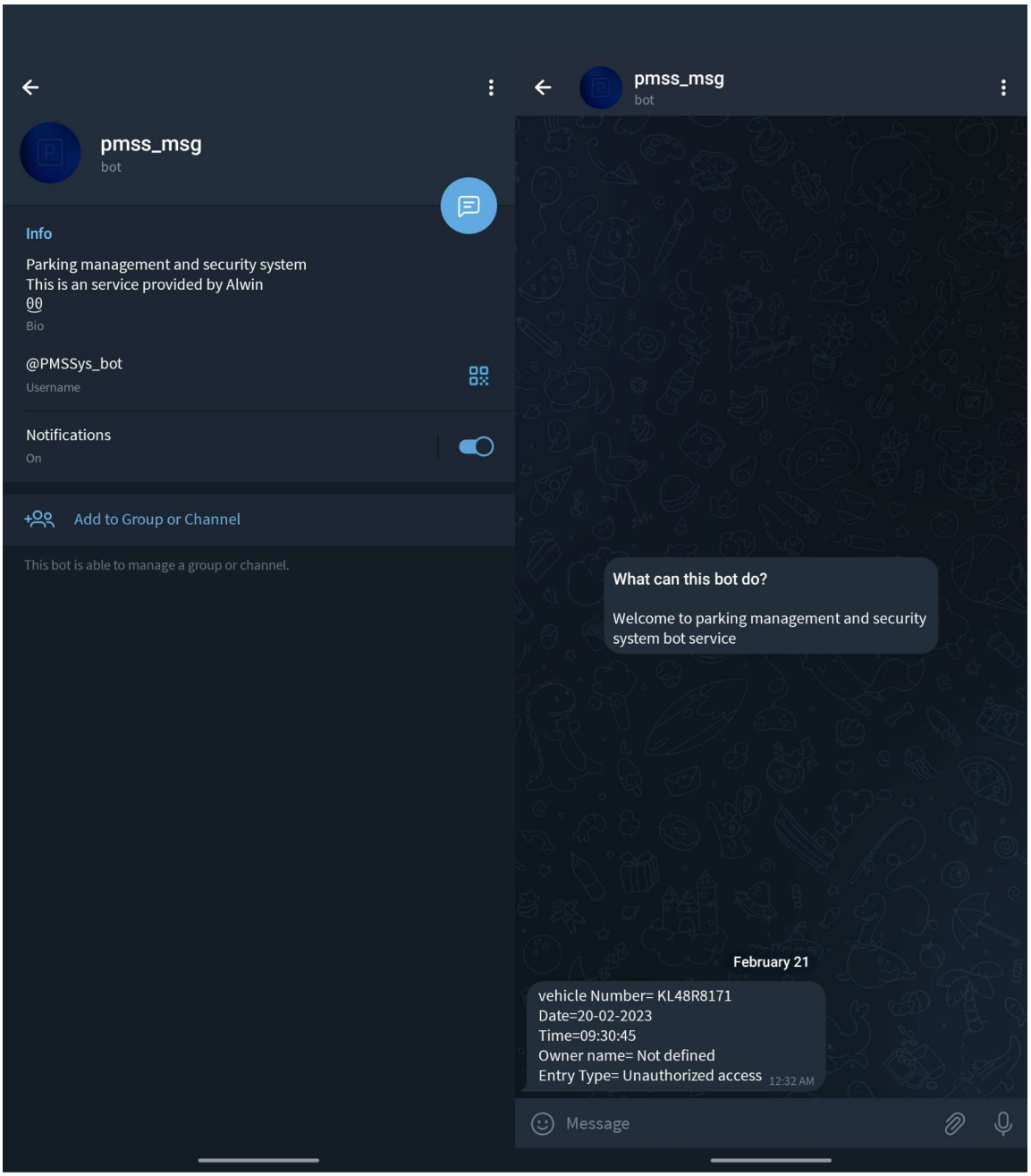


Figure 6.13 Telegram Bot

6.2.14 PMS Android BroadCast APPLICATION

PMS android broadcast App is used to broadcast the Parking lot number to Vechile's driver so he/she can park in the assigned parking lot

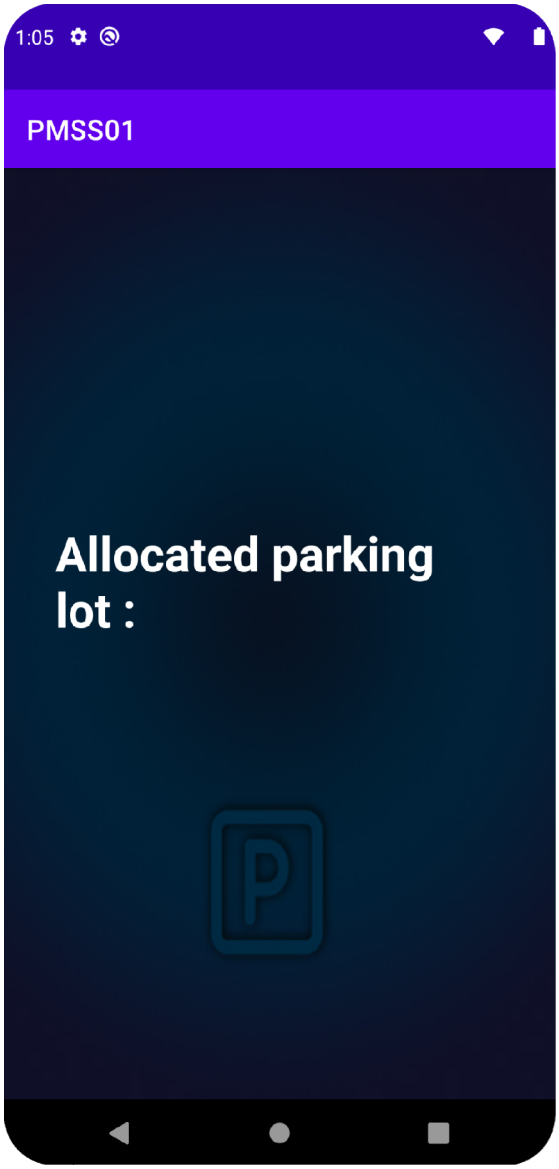
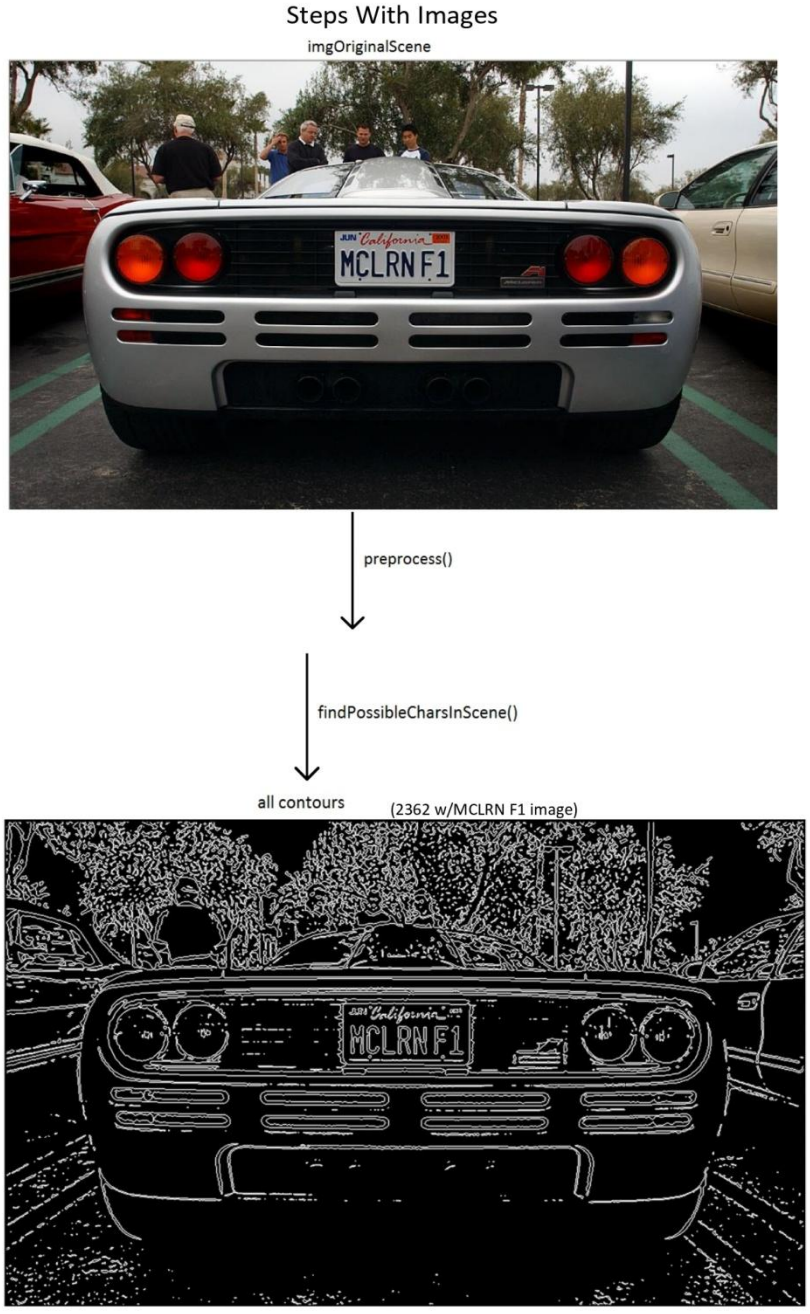


Figure 6.14 Android Broadcasting App

6.3 Vehicle Number plate and charter detection

Following Pictures show how the number plate and its character detected by PMS system in detailed steps

6.3.1 Number Plate Detection

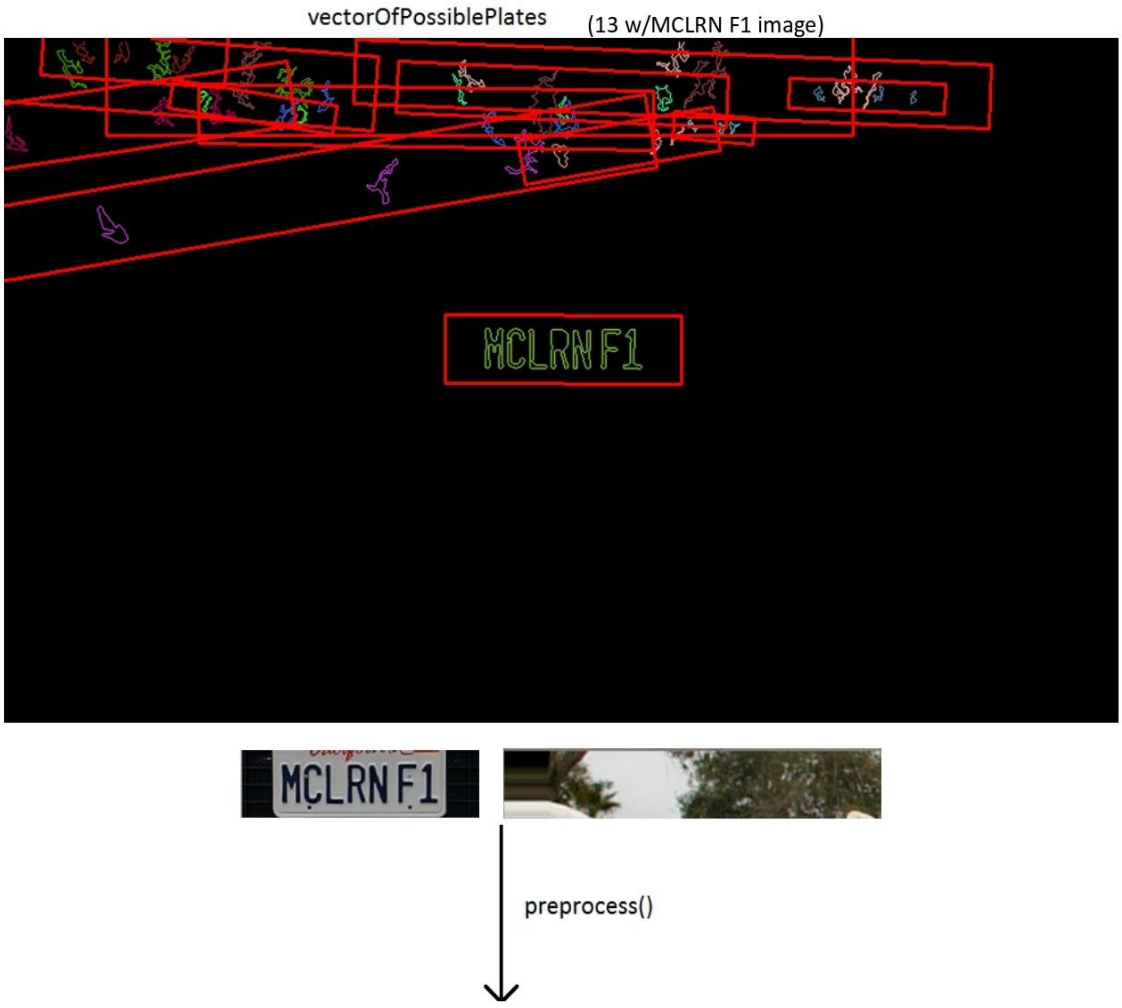


vectorOfPossibleCharsInScene (131 w/MCLRN F1 image)

findVectorOfVectorsOfMatchingChars()

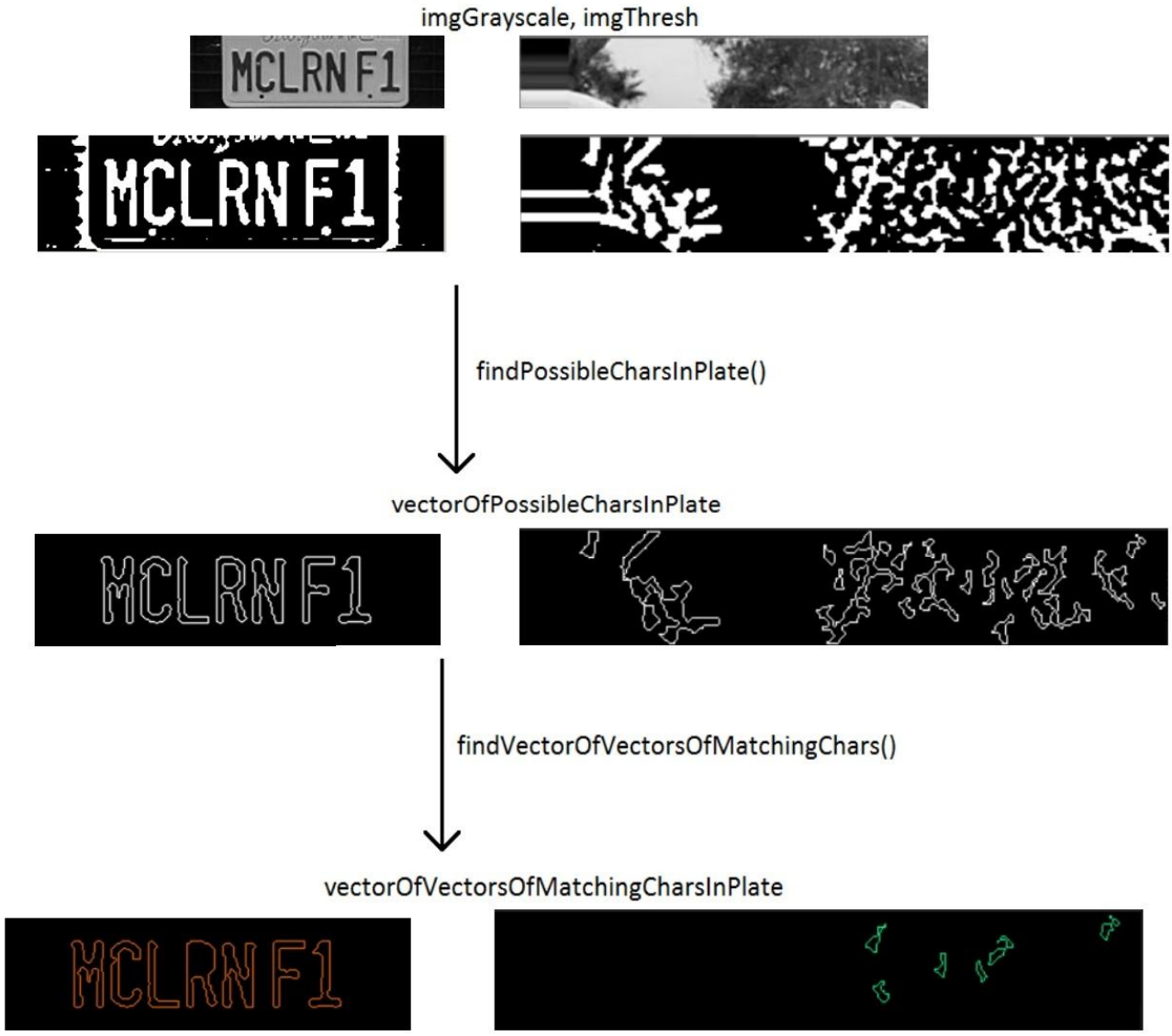
vectorOfVectorsOfMatchingCharsInScene (13 w/MCLRN F1 image)

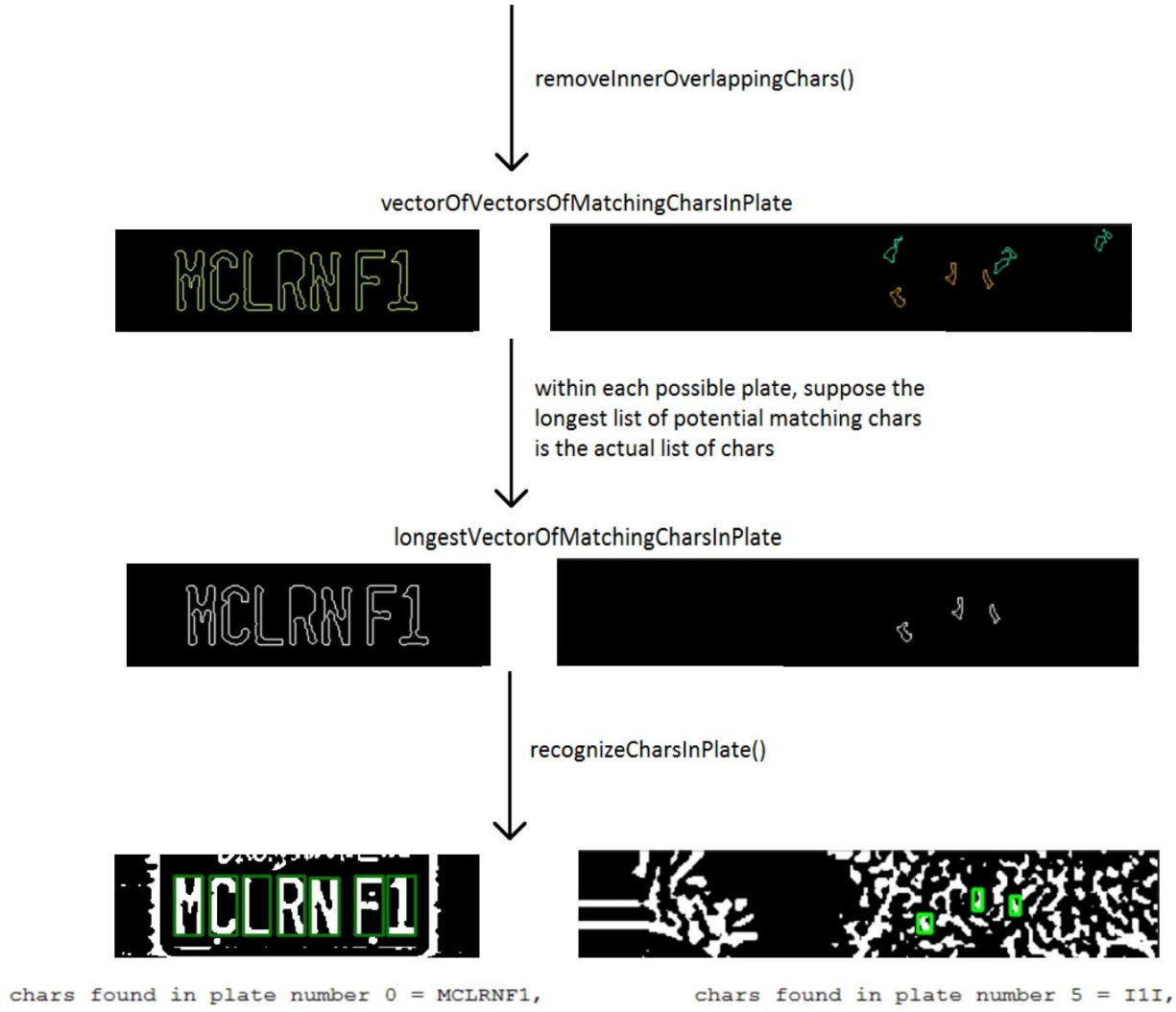
extractPlate()



With the help of the KNN Detect License plate, we found the number plate(possible number plate) location and the charterers on the LP, to decode the character string from the LP we use KNN- Detect character string python program that is already trained to detect and decode character string.

6.3.2 Character String Detection from LP





The output is the only possible string that was detected using the KNN algorithm, it is detected by sorting all possible strings in a list and sorting out the longest string in the list and the output is printed out by the algorithm and then store into a variable in the program.

SYSTEM SECURITY

7.1 DataBase Security

Database security is the practice of protecting digital databases from unauthorized access, corruption, and theft of sensitive information. It involves the implementation of various security measures, such as access control, encryption, backups, and monitoring, to prevent data breaches and ensure data integrity. Database security is essential for organizations that store confidential and sensitive information, such as financial records, personal information, and intellectual property. Failure to implement adequate security measures can result in data loss, legal liabilities, reputational damage, and financial losses. Therefore, it is important for organizations to prioritize database security and adopt a comprehensive security strategy that addresses all potential vulnerabilities and threats.

7.2 Application Security

Application security is the process of making apps more secure by finding, fixing, and enhancing the security of apps. Much of this happens during the development phase, but it includes tools and methods to protect apps once they are deployed. This is becoming more important as hackers increasingly target applications with their attacks. Application security is getting a lot of attention. Hundreds of tools are available to secure various elements of your applications portfolio, from locking down coding changes to assessing inadvertent coding threats, evaluating encryption options, and auditing permissions and access rights.

7.3 proposed System Security

In the PMS System the application is secured with a username and password authentication process.Only with a username and password a user can log into the PMS system and run the application. The username and password are stored in the "logins" Database in the local system and the DBMS system is also locked by its own security system with username and password. A user can only login into the system when the user enters the user name and password that match within the login database.

SYSTEM TESTING

8.1 Testing

testing is an important process of evaluating and verifying software to ensure that it meets the specified requirements and works as intended. It involves running the software through a series of test cases to identify any defects, bugs, or errors, and to validate that it meets the functional and non-functional requirements.

This is important because it helps to ensure that software is reliable, efficient, and meets the user's expectations. It also helps to identify and fix defects before the software is released, which can save time and resources in the long run.

There are various types of software testing, including functional testing, performance testing, security testing, and usability testing, among others. Each type of testing has its own specific goals and methods, but all are designed to ensure that the software is of high quality and meets the needs of its users.

8.2 Unit testing

Testing individual modules or components of the software to ensure that they function correctly in isolation. In the case of the PMS system, the system contains many different units that are:

- Ip-camera /Video input source
- Number plate detection
- Character Detection
- Data management
- Graphical User Interface
- HTML Requests
- Telegram Bot
- Android BroadCasting

8.2.1 Camera/Video source unit testing

The video source is the input data that is used in the PMS system, without this source PMSS cant work properly

Video source connection is established via support of OpenCV library packages such as

- cv2.VideoCapture
- cv2.imshow

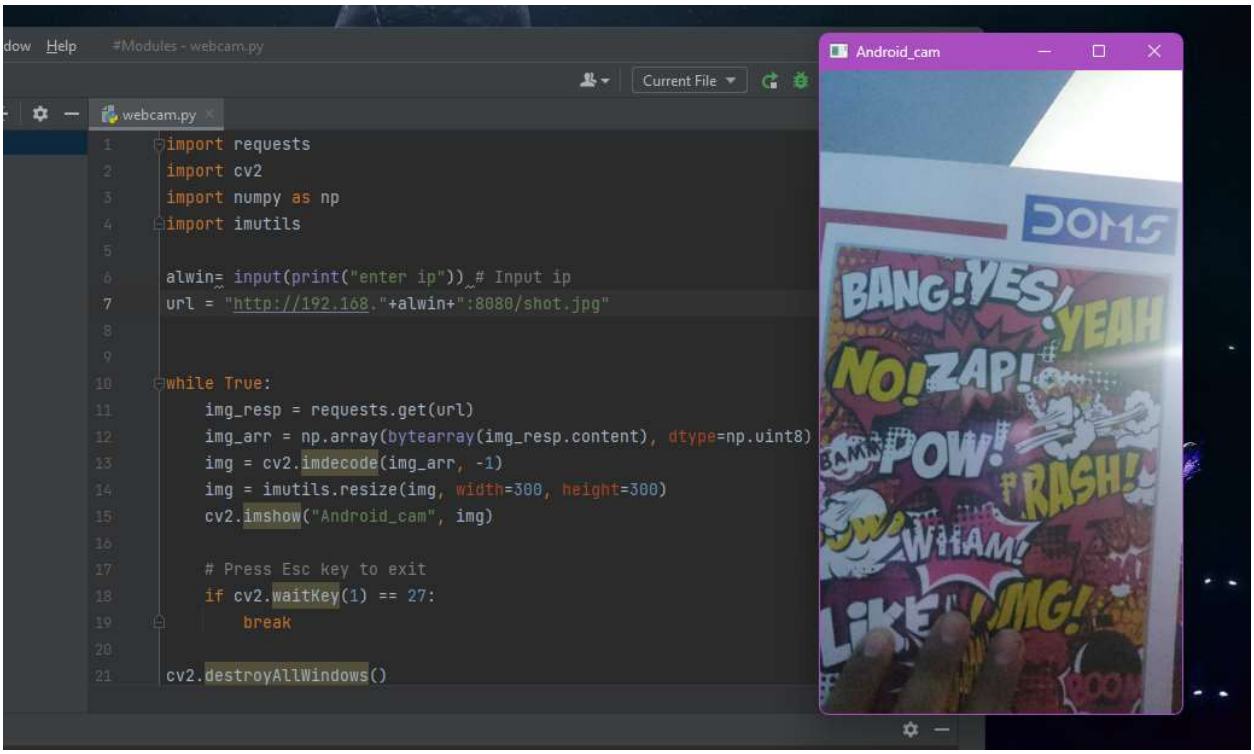


Figure 8.1 Video Source unit testing

Results

Input device (camera) successfully connected

8.2.2 Number plate detection testing

Number plate or license Plate(LP) detection is one of the hardest parts of the PMS system. The LP is backed up by the KNN algorithm to find the most possible LP that can find using the algorithm

Figure 8.2 Number Plate detection

Results:

Number plates were detected

8.2.3 Character string detection Testing

After detecting the possible number plates (LP) next job is to find the characters strings inside the detected number plates and store into a variable for future use. The character String is also detected by the KNN algorithm for detecting characters. Character detection is also a problem maker in the PMS system because the accuracy of detection may cause inaccurate data and sometimes print the wrong string of lines

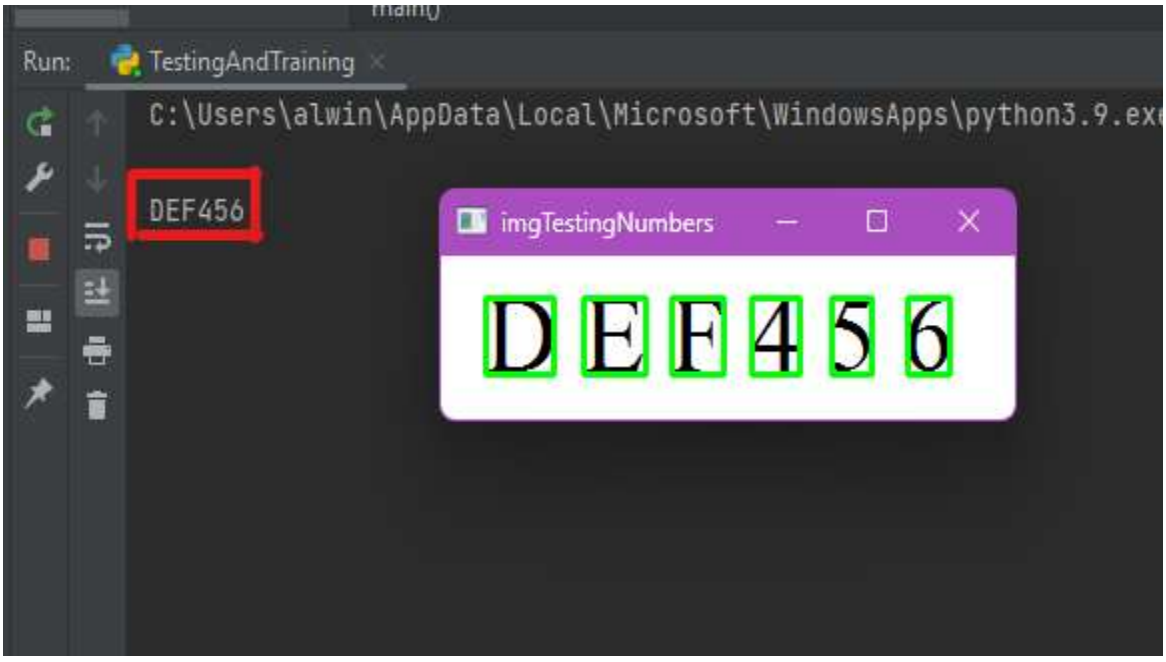


Fig 8.3 Character detection

Results

- The character strings were detected

8.2.4 Data management

File management is when PMS system stores data in excel or CSV files for detailed report and information. It stored data in excel format for easy management and viewing

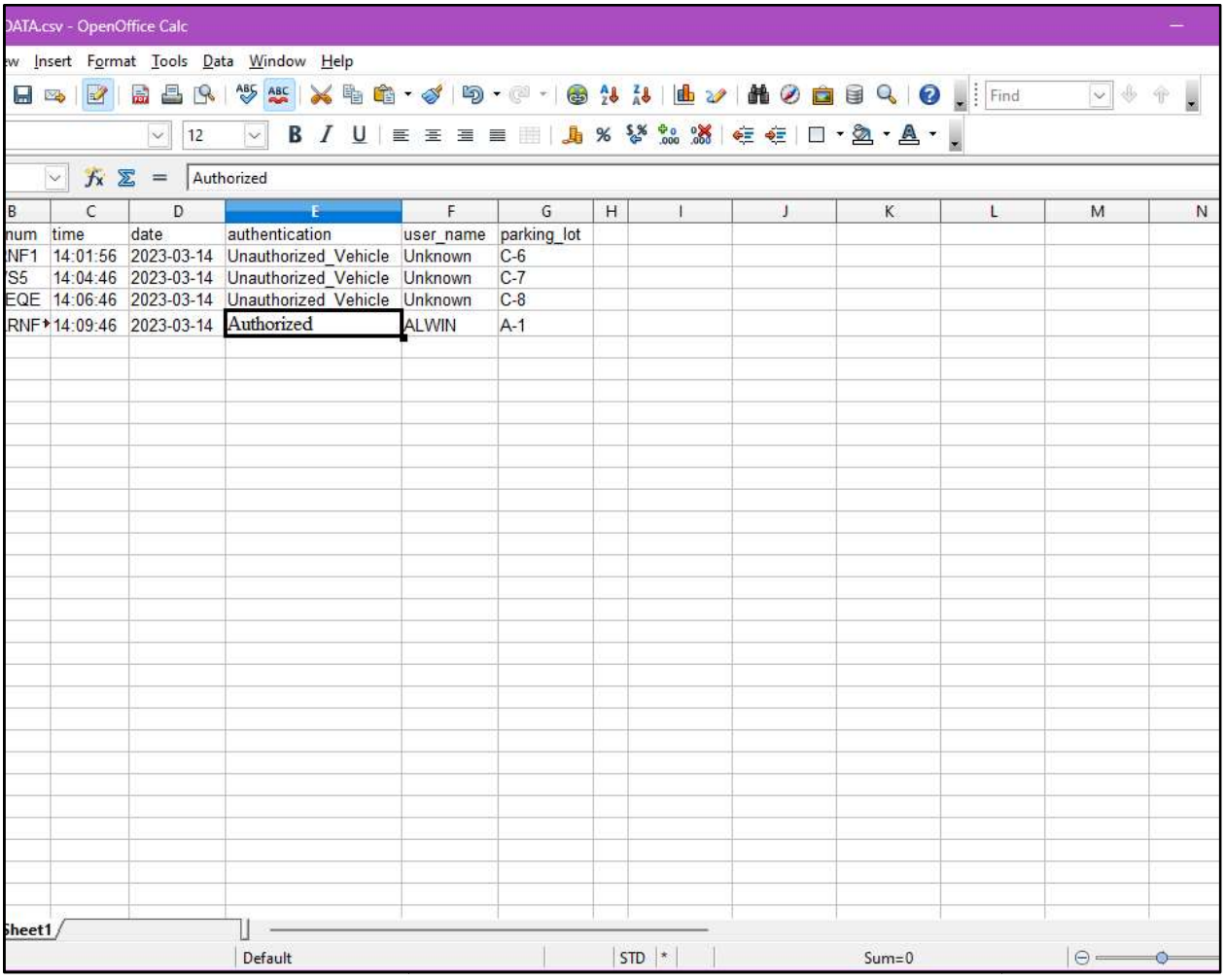


fig 8.4 Data Stored in a CSV file

Results

- Output data and other information are stored in a file system and working properly

8.2.5 Graphical User Interface Testing

Graphical User Interface(GUI)is used to interact between PMS system and user.PMS contains lots of widgets, buttons, forms, media players, etc...

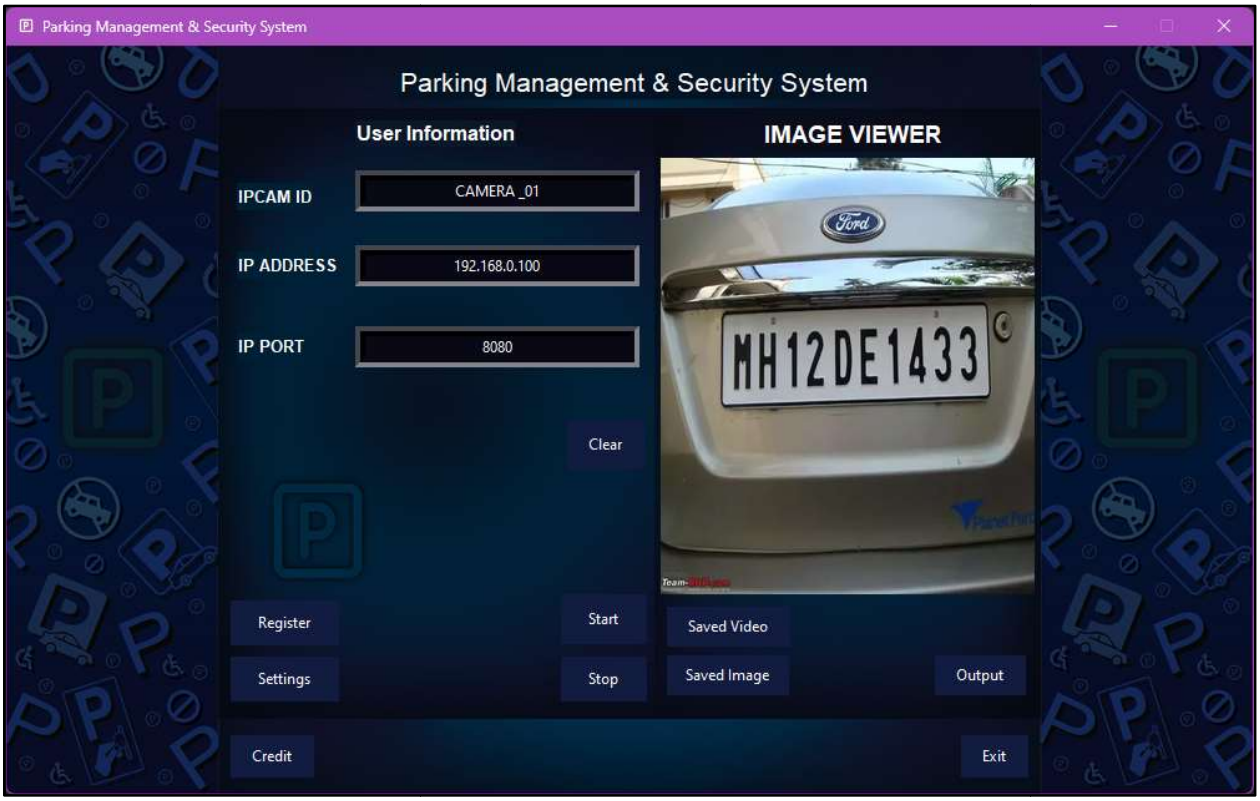


Fig 8.5 GUI

Results

- GUI elements such as widgets, Buttons, Forms, etc..work fine
- PMS system isn't working properly because the system will hang and reduce its performance when PMS system running constantly. looping isn't feasible
- PMS system works fine when looping elements are removed from the system

8.2.6 Telegram bot service testing

Telegram bot service is established for the user to notify vehicle's details such as vehicles number, Time, Date, Entry type, Owner name, etc. The bot is tested by sending messages to the user or the client without any errors and other problems

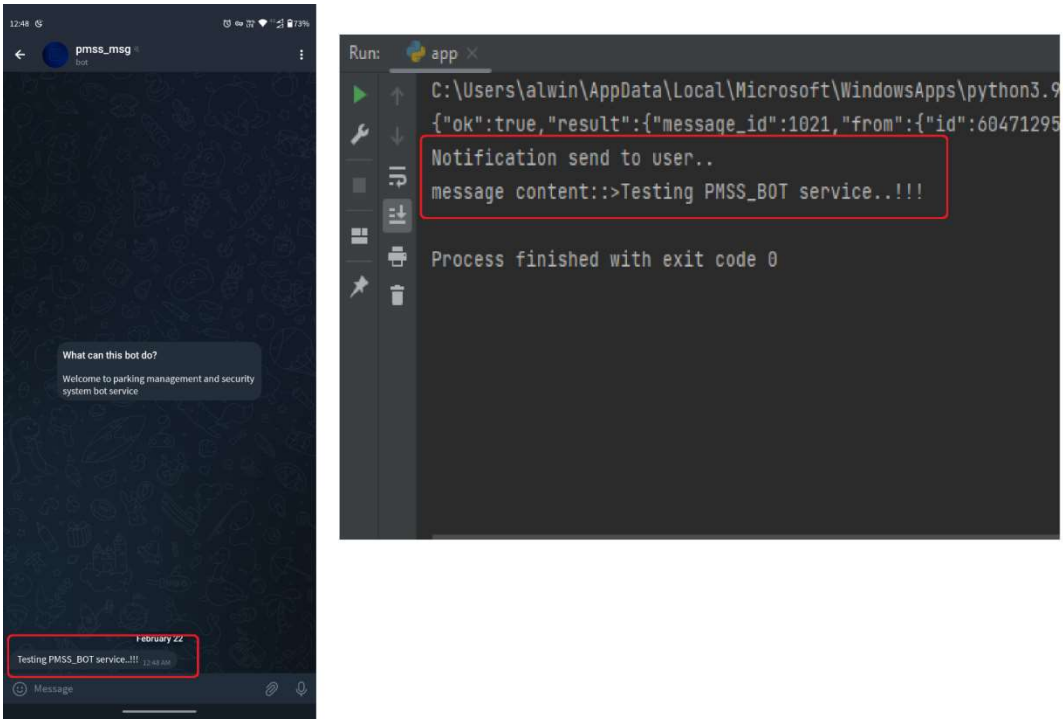


Fig 8.6 Telegram bot testing

Results

- Messages from the PMS system are successfully sent to the User
- PMSS_MSG bot service running without any problems

8.2.7 Android Broadcasting App testing

Android BroadCasting app is used to display Parking lot number to the driver.

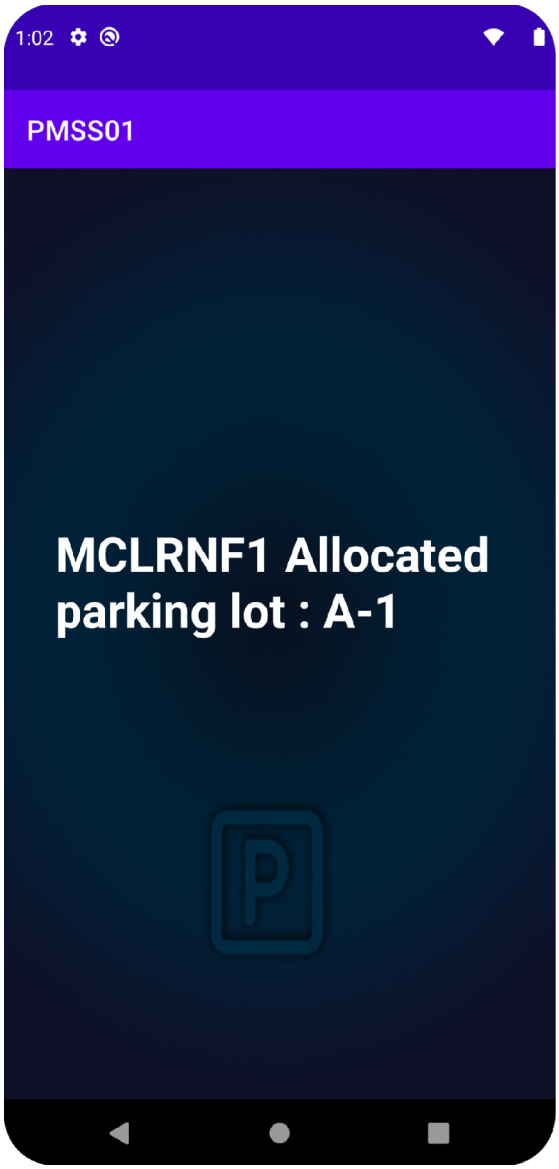


Figure 8.7 Android Broadcasting app Testing

Results

- **The android app works perfectly**

SYSTEM DEMO & PREVIEW

9.1 PMS system (python terminal)

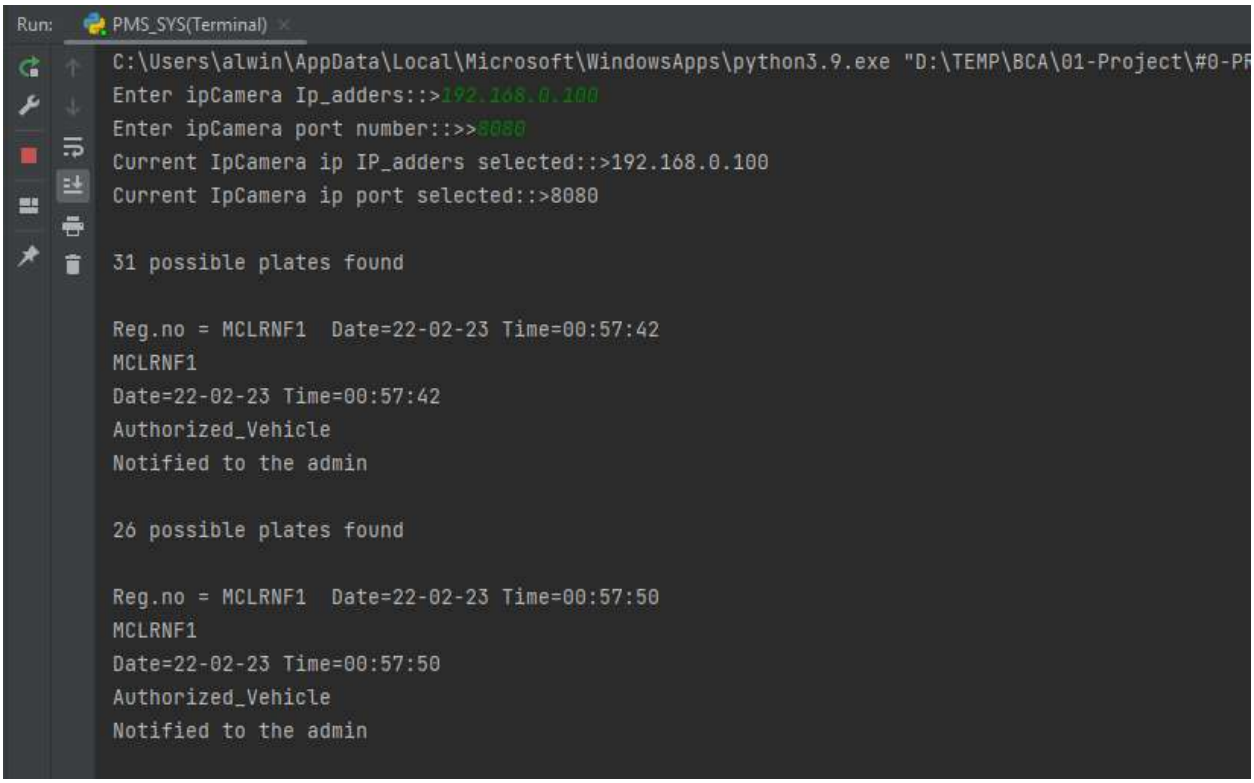


Fig 9.1 PMS system terminal working

9.2 PMS system (Graphical User Interface)

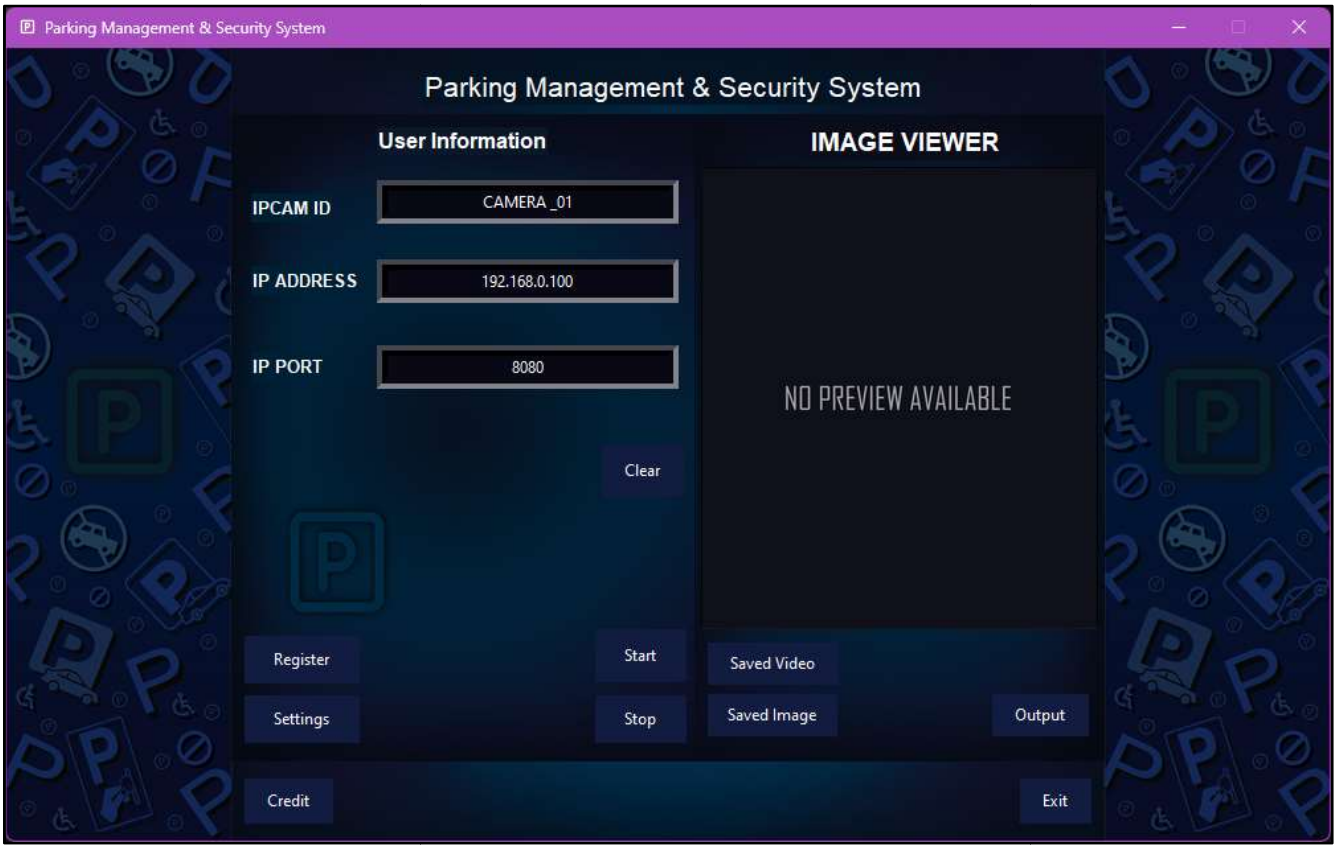


Fig 9.2 PMS system Graphical User Interface

6.3 PMS system Telegram bot preview

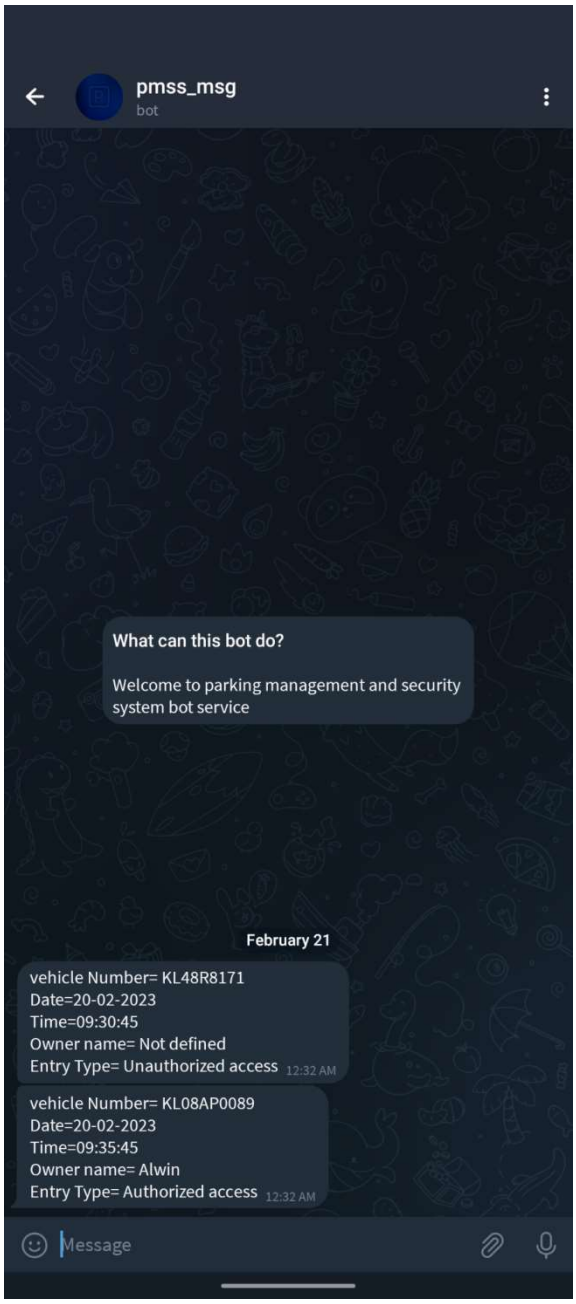


Figure 9.3 Telegram Bot

9.4 PMS System Broadcast app preview

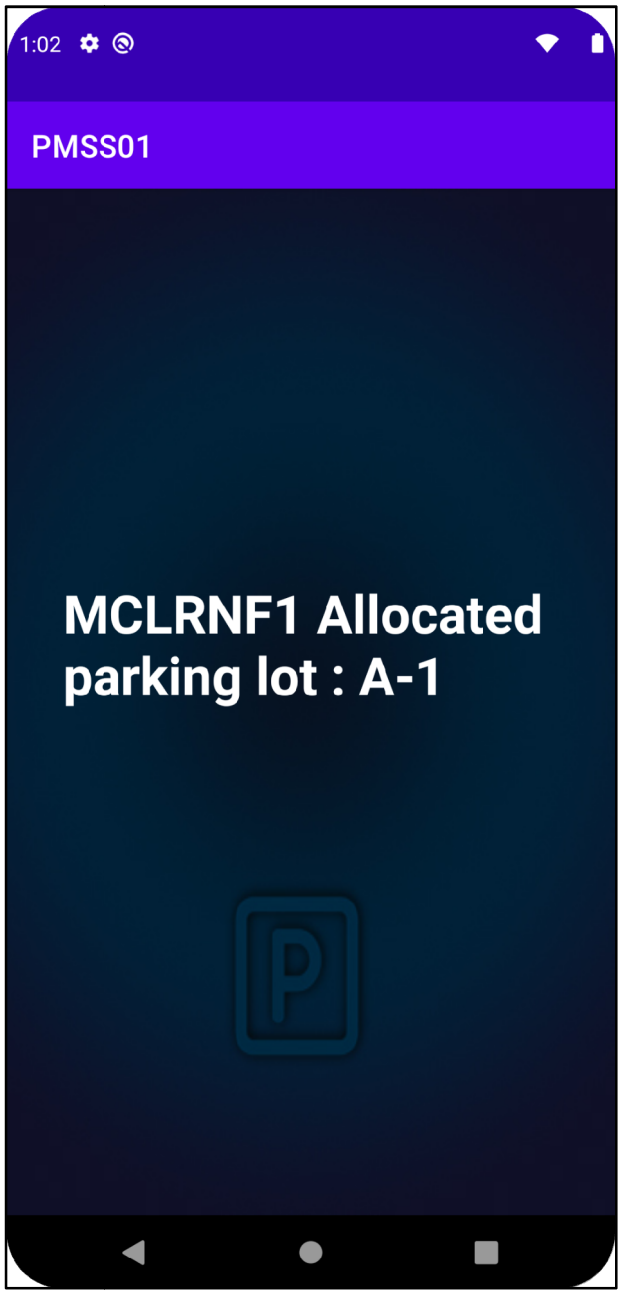


Figure 9.4 Broadcast app preview

CONCLUSION

10.1 Limitations and Future Work

10.1.1 Limitations

- ★ character detection on the number plate is sometimes inaccurate Because of the similarities of some characters, there may be some errors during recognition. The confused characters mainly are B and 8, E and F, D and O, S and 5, and Z and 2
- ★ PMS system need to install on the local network.
- ★ Image detection may fail sometimes due to less accurate images

10.1.2 Future Work

- ★ Android application will be developed
- ★ PMS system can be hosted in a server to increase quality and performance
- ★ Parking lots or slots will be added in the near future
- ★ Alarm System will be implemented
- ★ WhatsApp BOT will be introduced

10.2 Conclusions

In the PMS system, the live feed video from the camera is sent to PMS, where it detects vehicles and sends the frame with the vehicle's number plate. The OpenCV library is used to extract the vehicle number plate from the video source frame, and the license number of the vehicle is identified to obtain information about the vehicle owner.

In our proposed system, we use a technique where a picture of the vehicle's plate is taken. We then perform noise reduction to enhance the image quality, followed by segmentation and binarization. With the help of KNN character detection, we can identify the license plate (LP) of the vehicle. The LP is then sent back to the PMS system to be stored in its database and Excel file.

The PMS system cross-checks the LP with its database for the vehicle owner's name and authorization. The system then sends the details to the users' Telegram accounts using the Telegram bot service provided by PMS. The photos and videos of the vehicle are also stored locally in the PMS system for security purposes.

BIBLIOGRAPHY

Bibliography

- www.github.com
- www.stackoverflow.com
- www.opencv.org
- www.analyticsvidhya.com
- www.geeksforgeeks.org

* _ * _ * _ * _ * _ * _ * _ * _ * _ * _ * _ * _ * _ * _ * _ * _ * _ * _ *