ÇANKAYA UNIVERSITY
FACULTY OF ENGINEERING
COMPUTER ENGINEERING DEPARTMENT

Project Report
Version 2

CENG 408
Innovative System Design and Development II

<202121>
<LICENCE PLATE RECOGNITION SYSTEM>

Mehmet Furkan Turan

201726072

Arda Kayış

201711043

Doğukan Tutak

201711065

Burak Çetin

201711015

Muhammed Emin Atalık

201711003

Advisor: ***Prof. Dr. Ahmet Coşar***

Table of Contents

Abstract

License Plate Recognition System offers a solution that has been needed in recent years and necessary steps can be taken for the operation of an efficient smart transportation network. Due to the rapid increase in vehicles, it has become a necessity for traffic control management. The main purpose of the License Plate Recognition System is for traffic monitoring and security purposes. License plate recognition uses image processing techniques or OCR techniques and edge detection technology to detect characters on license plates. The model consists of three modules: vehicle detection module, license plate segmentation module and recognition module. License plate recognition system is used in various fields such as police forces, military areas, traffic control and security. It is used in areas such as keeping city entrances and exits under control with the License Plate Recognition System, city security, and the establishment of smart traffic systems. In the License Plate Recognition System project, we aim to detect and read the license plates of vehicles by using image processing and deep learning methods.

Key words:

Image Processing, Deep Learning, Licence Plate, Optical Character Recognition (OCR).

Özet:

Plaka Tanıma Sistemi, verimli bir akıllı ulaşım ağının işletilmesi için son yıllarda ihtiyaç duyulan ve gerekli adımların atılabileceği bir çözüm sunmaktadır. Araçlardaki hızlı artış nedeniyle trafik kontrol yönetimi için bir zorunluluk haline gelmiştir. Plaka Tanıma Sisteminin temel amacı trafik izleme ve güvenlik amaçlıdır. Plaka tanıma, plakalardaki karakterleri algılamak için görüntü işleme tekniklerini veya OCR tekniklerini ve kenar algılama teknolojisini kullanır. Model, araç algılama modülü, plaka segmentasyon modülü ve tanıma modülü olmak üzere üç modülden oluşmaktadır. Plaka tanıma sistemi, polis kuvvetleri, askeri alanlar, trafik kontrolü ve güvenlik gibi çeşitli alanlarda kullanılmaktadır. Plaka Tanıma Sistemi ile şehir giriş çıkışlarının kontrol altında tutulması, şehir güvenliği, akıllı trafik sistemlerinin kurulması gibi alanlarda kullanılmaktadır. Biz de Plaka Tanıma Sistemi projesinde görüntü işleme ve derin öğrenme yöntemlerini kullanarak araçların plakalarının tespitinin yapılması ve plakaların okunmasını sağlamayı amaçlıyoruz.

Anahtar Kelimeler:

Görüntü İşleme, Derin Öğrenme, Plaka, Optik Karakter Tanıma (OKT).

1. Introduction

License Plate Identification System is a method that is frequently used in the identification of vehicles, in parking lots, in urban areas and in places where security is important such as road cameras. A license plate identification system is used to uniquely identify a vehicle. License Plate Recognition system plays an important role in many applications such as electronic payment system (toll payment and parking fee payment), stolen car finding, traffic surveillance. For example, in parking, plates are used to calculate the parking time. When a vehicle enters the door, the license plate is automatically recognized and recorded. When leaving, the license plate is re-recognized and compared to the registered license plates. The time difference is used to calculate the parking fee. License Plate Recognition System is a useful and versatile system as it is automatic.

Various researchers have suggested various techniques for each step, and an individual technique has its positives and negatives. The license plate recognition method includes three main steps. This is the zone of removal of interest, removal of license plate numbers and character recognition. Below is the block diagram for the plate system in Figure 1.

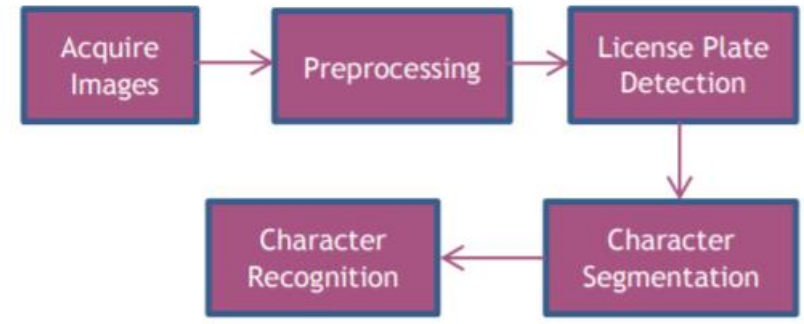


Figure 1 Block Diagram of License Plate Recognition

License plate recognition is provided through the ANPR instrument program. Images captured using cameras show the printout of the license plate. The basic job of ANPR is to read and open the license plate. ANPR is also called ALPR (Automatic License Plate Recognition). The software of this system creates a record of all license plates which it interprets with all relevant data such as date, time and GPS location. It uses OCR technology to recognize the characters of number plates. License Plate Recognition is an important process of automated parking. It will be a feature expected by industry demand for higher commercial parking management projects in smart city areas.

It has also been used for security purposes in toll collection systems, traffic control, gas stations and many other exploration opportunities. Intelligent Transportation Systems play a leading role in facilitating smart cities due to its many applications such as highway surveillance, city logistics and traffic law enforcement, and more.

With the License Plate Recognition System, our goal is to determine the license plates of vehicles using image processing methods such as character and segmentation. We aim to determine the characteristics such as the type, type, size and color of the vehicles by using the determined plates. In addition, we aim to use the deep learning method using different vehicle images and datasets of the TensorFlow library in order to be accurate and realistic when defining the license plates in the License Plate Identification System.

1.1 Problem Statement

In the License Plate Recognition System, our aim is to recognize and determine the license plates by using the deep learning method with the camera, with methods that will increase the accuracy of the result. Thanks to the deep learning method, it is aimed to make license plate recognition faster by shortening the license plate recognition time.

Controlling limited access to restricted locations, automating parking systems and payments, monitoring toll tax payments, controlling state border pass and security measures in countries, traffic management, and law enforcement are all applications that use a license plate recognition system.

Since it is aimed to use deep learning, image optimization, camera adjustment and image processing method, data processing speeds and related delay times can be seen as potential problems. Since license plate recognition will be made with different vehicle images, the teaching time may vary. There can also be issues with the camera, such as lags.

1.2 Background or Related Work

Several studies and research projects have already been completed. Using the dataset produced by them, creates a deep learning model to recognize the number plate. They employed the Keras deep learning package in conjunction with the TensorFlow framework. They used a variety of image processing techniques because the photographs were captured against a live scene. Following these steps, the CNN model is trained on the images.

In a machine learning method is used to construct an automatic number plate recognition system. They used an infrared camera to gather data, which was then pre-processed with contrast enhancement and noise reduction.

They then used the Region of Interest to locate the number plate in the image (RoI). After that, contour tracing is used to extract the image's key features. Then, to find the edges of the characters on the number plate, edge detection is used. Last but not least, segmentation is used to separate the characters. Artificial Neural Networks use pattern matching to distinguish individual characters (ANN). Vehicle image capture, number plate detection, character segmentation, and character identification are the basic phases in ANPR.

Plate size, plate placement, plate background, and screw must all be considered while detecting number plates. Character recognition follows, which is commonly accomplished via Artificial Neural Networks, template matching, or Optical Character Recognition (OCR) approaches. [1]

1.3 Solution Statement

With more specific algorithms and methods, the license plate recognition system can be made in accordance with speed and time. Thus, problems such as camera optimization can be solved.

1.4 Contribution

We aim to make the license plate recognition system that we will build in a similar way to the previous one, but by improving it. While doing the project, we aim to do it by considering features such as deep learning and performance.

2. Literature Search

2.1 History of Licence Plate Recognition System

In light of its widespread use in recent years for a variety of purposes such as traffic studies, access control, and parking, many people believe it was invented in this millennium. The Police Scientific Development Branch devised ANPR in 1976 in the United Kingdom, and early devices were composed for use in 1979. [2]

Earlier prototypes were premature, with low precision readings, and only worked in restricted laboratory circumstances, making real-world use practically impossible. A few years later, at Wokingham, England, moderate improvements were made and functioning models were implemented. As a result, business contracts and apps for early adopter use were successfully developed. While still primitive, with an accuracy rate of less than 60%, the technology was cutting-edge at the time.

After several decades, the technology has advanced significantly, and previous constraints caused by vehicle speed, light fluctuation, angular skew, character segmentation, and recognition have been overcome using today's algorithm technology. Furthermore, previously exorbitant prices have given way to more reasonably priced implementations, allowing the app to thrive in a variety of businesses. [3]

2.2 Methods of Licence Plate Recognition System

2.2.1 Licence Plate Detection

In video analysis of the number plate image, computer vision and character recognition, as well as algorithms for license plate recognition, play a vital role. As a result, they are the foundation of any ANPR system. A camera, a frame grabber, a computer, and custom created software for image processing, analysis, and recognition make up the system for autonomous car license plate recognition. Over the last few years, vehicle identification has been a hot topic of research. Several studies have been conducted in order to determine the type of vehicle, such as a car, truck, scooter, or motorbike.

A machine-understandable definition is required in order to detect a license plate. A license plate can be defined as "a rectangular area of a vehicle with a high density of horizontal and vertical edges," based on its characteristics. [7].

Many algorithms have been proposed to tackle the license plate detection challenge based on these features, some of which are anchored in standard computer vision techniques and others in deep learning. Figure 2 shows a classification of the license plate identification algorithms now in use.

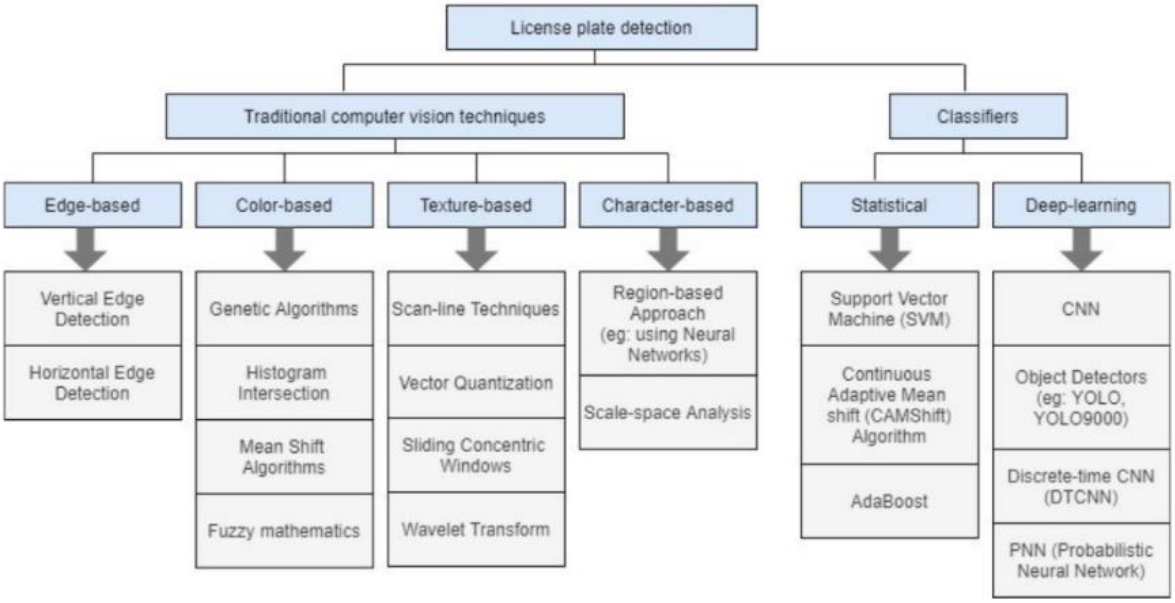


Figure 2 Categorization of Related License Plate Detection Techniques

2.2.1.1 Edge Based Methods

Edge detection is the process of finding edges in an image and is an important step in understanding visual features. Edges are regarded to provide important information and have distinguishing characteristics. It considerably reduces the size of the image to be analyzed and filters away extraneous data, leaving only the structural features of an image for a business problem to be solved.

To find edges in an image, edge-based segmentation algorithms exploit a variety of discontinuities in grey level, color, texture, brightness, saturation, contrast, and other characteristics. Additional processing procedures must be undertaken to concatenate all of the edges into edge chains that better fit the image's borders to improve the findings even more. Edge detection techniques are divided into two categories.

Grey histograms and gradient-based approaches These methods employ basic edge detection operators such as the sobel operator, the canny operator, and Robert's variable, among others. These operators help detect edge discontinuities and, as a result, designate edge boundaries. The end goal is to achieve at least a partial segmentation using this method, in which we combine all of the local edges into a new binary picture that only contains edge chains that match the required existing objects or image sections. [8]

2.2.1.2 Colour - Based Methods

Colour-based approaches rely on the fact that a license plate's colour differs from the vehicle's background colour. In addition, the colour combination of the plate and its characters is only present in the plate portion of the image. To identify pixels in an input image against different illuminations, the Hue, Lightness, and Saturation (HLS) colour model can be utilized. Unlike the RGB model, which divides pixels into three categories, the HLS model divides pixels into several categories.

2.2.1.3 Texture Based Techniques

Tactile texture refers to the real feel of a surface, and visual texture refers to observing the shape or contents of an image. In image processing, the texture of an image may be defined as a function of the spatial variation of the brightness intensity of the pixels. Texture is the most common phrase used to describe objects or concepts in a picture. [9]

In texture-based techniques, the presence of characters on the license plate is employed as the foundation for plate detection. Due to the substantial colour difference between the plate and its characters, a regular colour transition appears on the license plate. As a result, when the image is grey-scaled, there is a clear inequality between the characters and the plate backdrop. As a result, a distinctive pixel intensity distribution is formed around the plate region. Furthermore, the plate region has a high edge density due to the colour shift. [10]

2.2.1.4 Character-Based Methods

License plate detection also involves examining an image for the presence of characters and locating them. These approaches are classified as character-based methods because they consider the region containing characters to be a suitable plate region. To categorize those extreme locations, a neural network classifier is utilized, and if any linear spatial configuration is identified, it is assumed to be the potential region containing the license plate. This approach is said to be resistant to a variety of lighting situations and views, with a stated detection accuracy of 95%. [11]

2.2.1.5 Deep-Learning Techniques

Most statistical methods have been substituted by deep learning neural networks due to their excellent accuracy in object detection, according to current developments in computer vision technologies. Many studies in license plate detection have used various forms of neural networks as a result of this fact. [12]

2.2.2 Licence Plate Recognition

This stage, which is the second in a multi-step automatic license plate recognition pipeline, is in charge of "reading" the license plate once the detection stage has located it. This is a type of optical character recognition that takes into account specific characteristics of the license plate.

Many countries, for example, have tight regulations concerning the font and color of license plates, which are normally chosen to be easy to see. The license plates, on the other hand, have certain peculiarities [13]. For example, when the photograph is shot outside, the system designers must account for variables such as ambient light, uneven brightness, and weather effects. Even if you have a regular license plate, it is possible that it will be damaged or rotated. Figure 3 depicts a typical ALPR system's recognition pipeline, along with the strategies that can be used at each stage. However, depending on the methodology used, some of the chores may be skipped.

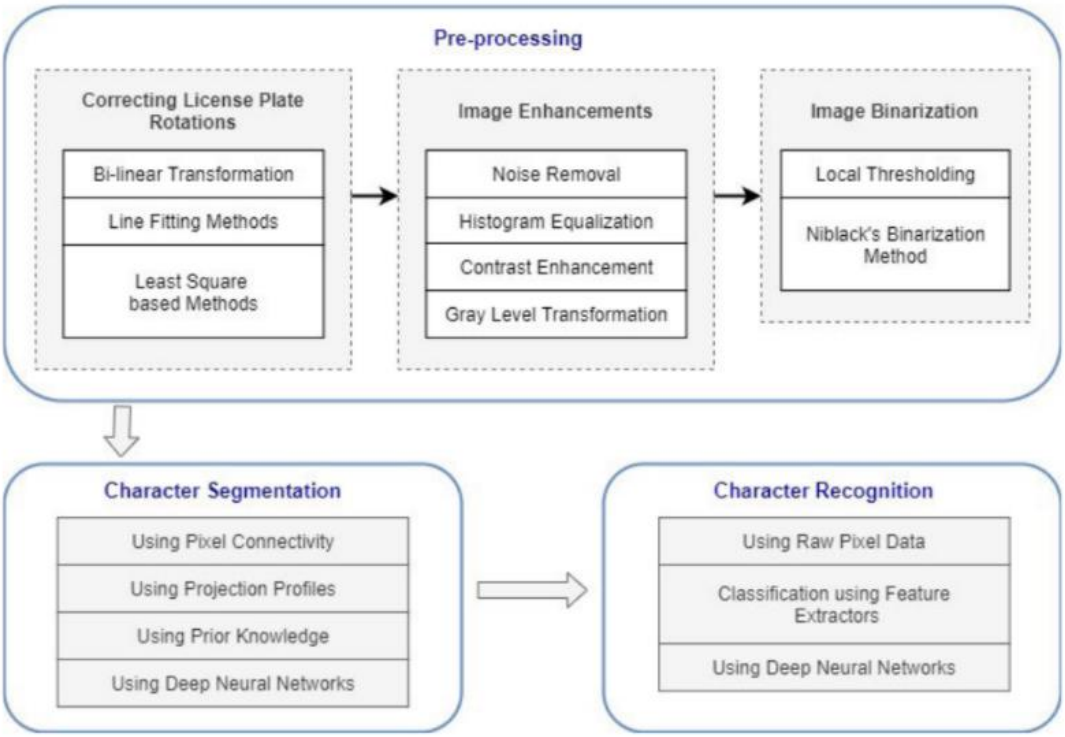


Figure 3 Pipeline for License Plate Recognition and Related Techniques

2.2.2.1 Pre-processing Techniques

To manage particular obstacles in license plate identification, certain pre-processing operations are performed before character segmentation and recognition. Bilinear transformations [14], least square-based methods [15], and line fitting methods [16] are examples of rotation techniques that have been employed in related investigations. The image is binarized before segmentation in several traditional machine vision-based character recognition systems. In comparison to grey-scale or color images, the method makes it easier to differentiate the pixels that belong to the characters in the image.

However, the binarization threshold must be calculated correctly to avoid character blending or merging with the license plate frame in the binary image, which makes segmentation difficult [17]. Image improvement techniques such as noise reduction, histogram equalization, contrast enhancement, and grey level transformation can be used to define the threshold value. Even with these improvements, getting a single threshold might be difficult.

2.2.2.2 Character Segmentation

Characters are segmented before classification in various optical character recognition systems. The feature of having distinct colours for the backdrop and characters is taken into account by license plate character segmentation algorithms. Binarization of the image facilitates this separation by assigning opposite "colours" to the foreground (character) and background pixels.

Character Segmentation Using Pixel Connectivity:

For character segmentation, pixel connection is a straightforward method. The connected pixels are tagged here, and if they have the same label for an item of a certain size or aspect ratio, they are extracted as a character. One issue with pixel connectivity-based approaches is that they fail when characters are damaged or when binarization threshold selection causes characters to be connected.

Pixel connectivity-based techniques, on the other hand, are resistant to rotating license plates and are reasonably easy to deploy. The license plate identification pipeline is further simplified by the absence of the necessity for pre-processing to correct for license plate rotation.

Character Segmentation Using Projection Profiles:

After image binarization, projection profiles methods take advantage of the fact that the character and background pixels in the license plate image have different colors. Vertical projections are usually utilized to detect the character's beginning and ending points, and then horizontal projects are used to extract the character. Project-based approaches, on the other hand, are susceptible to image quality and noise.

As a result, a de-noising stage must be added in the recognition pipeline's pre-processing stage. Although the projection-based approaches have lower resilient values for rotations than pixel connectivity-based methods, they are nonetheless robust to rotations and independent of character placements.

Character Segmentation Using Prior Knowledge:

For character segmentation, prior knowledge of the license plate is employed, such as the aspect ratio of the characters and the ratio of various colored pixels in the image. Prior knowledge-based approaches are usually simple to execute; nevertheless, these methods are frequently specific to the regions in which they were meant to work and do not apply in other situations.

Character Segmentation Using Deep Learning:

The use of CNN (Convolutional Neural Networks) for the task related with computer vision has become a current strategy for character segmentation. The CNN is given a localized license plate as input, and the bounding boxes of each character are produced as output. However, compared to standard computer vision-based algorithms, CNN execution consumes more time and resources depending on the dataset.

In addition, in later stages of various deep learning-based license plate recognition pipelines, explicit character segmentation was skipped in favour of implicit character segmentation, resulting in a reduction in the number of parameters and computing cost. [18]

2.2.2.3 Character Recognition

Many classification approaches necessitate fixed-size learning model inputs. The input segments are re-scaled before classification since the output from the segmentation stage varies in size. Because the amount of characters, their relative positions, and possible values are usually known, each segment is classified as one of the possible values. There are three scenarios in which this can be evaluated.

(A) Compare all the pixel values in the raw image data directly with the predefined templates, (B) extract features using various image processing and machine learning approaches before categorizing the segments, and (C) classify segments using deep learning techniques.

A) Template and Pattern Matching Techniques:

Given that license plates often have a consistent font and character size, using template matching techniques to identify characters is a popular choice. Binarized images are commonly used for template matching. A predefined template is constructed for each possible character, and each segment is matched with each template to find the most comparable template. Additional templates must be kept in order to support character rotations, which increases computation time and processing memory.

B) Character Recognition Using Feature Extractors:

In general, recognizing a character does not necessitate the use of all pixels. Feature extractors are thus used to separate basic features from images while lowering computing expenses. Some feature extraction algorithms can extract features that are resistant to image noise and rotation. In these methods, each segment is transformed into a feature vector, which is subsequently classified using a machine learning model. Eigenvector transformation, Gabor filter, and Kirsch edge detection are examples of feature extractor algorithms. The extracted features are classified using machine learning algorithms.

C) Character Recognition Using Deep Learning:

The benefit of adopting neural networks is that they can be fed raw pixel data and function as feature extractors and classifiers on their own. This goal has been achieved utilizing a variety of neural networks, including simple multilayer perceptrons, Probabilistic Neural Networks (PNN) [26], and discrete-time cellular networks.

However, CNN (Convolutional Neural Networks) have been employed in a number of recent research and have showed significant promise in a variety of computer vision applications. Another current approach is to use an object detection-based technology like YOLO directly (real-time vehicle detection using deep learning methods). [19] Although deep learning-based algorithms are more computationally expensive than alternative methods such as template matching and statistical feature extractors, they often yield superior accuracy.

3. Summary

3.1 Technology Used

Due to the wide library possibilities such as TensorFlow, OpenCV and Keras, we are considering the use of Python language in the system. During the development of the project software, we aim to use Python supported compilers such as Visual Studio, PyCharm and Spyder.

4. Software Requirements Specification

4.1 Introduction

4.1.1 Purpose

In the License Plate Recognition System, we aim to determine the characteristics such as the type, color and size of the vehicles by using image processing, image detection and recognition methods. In addition to these methods, we aim to ensure that the Plate Recognition System gives accurate results by using the deep learning method with libraries such as TensorFlow and Keras.

4.1.2 Scope of Project

License Plate Identification System is a method that is frequently used in the identification of vehicles, in parking lots, in urban areas and in places where security is important such as road cameras. A license plate identification system is used to identify a vehicle.

The license plate recognition system is made by detecting the license plate region of the vehicles with the vehicle image and reading and separating the characters on the plate with image processing. By developing the license plate recognition system with deep learning and image processing, it will be tried to work towards detecting license plate recognition and the type of vehicle whose license plate is recognized.

With the License Plate Recognition System, our goal is to determine the license plates of vehicles using image processing methods such as character recognition and segmentation. We aim to determine the characteristics such as the type, size and color of the vehicles by using the determined plates. OCR (Optical Character Recognition) technology is used to recognize the characters of number plates. In addition, we aim to use the deep learning method using different vehicle images and datasets of some libraries in order to be accurate and realistic when defining the license plates in the License Plate Recognition System.

4.2 General Description

4.2.1 Glossary (Definitions, Acronyms, and Abbreviations)

Term	Definition
Camera	A device for recording visual images in the form of photographic, film or video signals.
Character Recognition	Character recognition is a method for computers to recognize written or printed characters like numbers or letters and convert them into a format that the computer can understand.
Character Segmentation	Character segmentation is a process that attempts to break down a picture of a series of characters into individual symbol subimages. It's one of the decision-making stages of an optical character recognition system (OCR).
Deep Learning	Deep learning is a machine learning and artificial intelligence (AI) technique that is modeled after how humans learn. In contrast to the usual, machine learning algorithms, which are linear, the complexity and abstraction of deep learning algorithms are constructed in a hierarchy.
Keras	Keras is an open-source software library for artificial neural networks that includes a Python interface. Keras acts as an interface for the TensorFlow.
Licence Plate	A license plate is a sign that is placed on the front and back of a car and displays the vehicle's license number. A rectangular, generally metal plate with a series of numbers, letters, or both that is issued by a government to identify a legally registered vehicle one of the signs with numbers on it at the front and back of a car.
Licence Plate Recognition System	A license plate recognition system is a form of technology, primarily software, that allows computers to read the registration number (license

	number) of automobiles from digital images automatically.
Optical Character Recognition (OCR)	The electronic or mechanical translation of images of typed, handwritten, or printed text into machine-encoded text, whether from a scanned document, a photo of a document, a scene photo, or subtitle text superimposed on an image, is known as optical character recognition or optical character reader.
OpenCV (Open Source Computer Vision Library)	OpenCV is a programming library geared mostly at real-time computer vision.
Python	Python has a standard library in the works, as well as a few AI libraries. It offers a simple control flow and data structures, as well as a straightforward syntax. It also allows for interpretative run-time without the use of traditional compiler languages. Python is very good for prototyping AI algorithms because of this.
Software Requirements Specification	A document that completely specifies all of the functions of a proposed system as well as the limitations that it must work under. For example, this document.
Users	The person using the License Plate Recognition System.

4.2.2 User Characteristics

Users are which Persons or institutions that use, operate and observe the License Plate Recognition System.

Licence Plate Recognition System is a system that enables to read the license plates of the vehicles by detecting the vehicle according to the color and type of the vehicles. While doing license plate reading, it detects the vehicle, detects the location of the license plate and performs license plate recognition by reading the characters on the plate with OCR.

Support system manager is the person who helps the License Plate Recognition System when there is a problem with the system, camera or license plate reading.

4.2.3 Product Perspective

In the License Plate Recognition System, the system works by training the license plate position in the photos of the vehicles at different angles and in different frames, using the deep learning method, so that the license plates of the vehicles can be determined easily. For License Plate Recognition System, we prefer Python language to be used in the system due to its wide library possibilities such as TensorFlow, OpenCV and Keras. During the development of the project software, we aim to use Python supported compilers such as Visual Studio, PyCharm and Spyder.

In the License Plate Recognition System, besides the deep learning method, it is also aimed to use image processing and machine learning techniques. The system should be trained and tested using the sources in the license plate recognition system available on the internet, different vehicle images and the TensorFlow library dataset.

In case an external camera such as ESP32 is used as an external camera, images are taken via the ESP32 camera and transferred over Wi-Fi to a sufficiently equipped computer to operate the system. Thus, in the images transferred to the computer, first the location of the plate is determined, then the character cementation is done. Finally, character recognition is done and license plate recognition is done.

4.2.4 Overview of Functional Requirements

In the Plate Recognition System, the recognition process of the plates first takes place by finding all the contours in the picture. It happens that each stroke has its bounding rectangle. Then it has to compare and verify the side ratio and area of each bounding rectangle with an average plate. Image segmentation is then applied to the image within the verified contour to find the characters inside the plate. As the last step, the characters on the plate are recognized by using OCR (Optical Character Recognition). In the license plate recognition system, there is also a training phase by using the deep learning method for more recognition of the license plate in the images. Thus, different directions of vehicles and license plate positions can be better detected using multiple images or different sections of a video. In this way, it also provides convenience in terms of accuracy and speed in plate recognition.

4.2.5 General Constraints and Assumptions

Software Conditions: Since the images and times to be used for teaching using deep learning in the license plate recognition system can vary, it can be seen as the challenging part. Since the type of car will be determined with different vehicle images, the teaching time may vary. Every user shall need a computer and compiler that can run the codes in the software part of the license plate recognition system well.

Ambient Conditions: In order for the License Plate Recognition System to work properly, the camera must be in sufficient light and in suitable weather conditions, day or night.

Hardware Conditions: In case of using an external camera such as ESP32 other than the normal camera, the video must be at a suitable distance to be sent. Also the videos need to be sent via the Wi-Fi module over the network. In addition, a connection between 20 and 25 meters can be established to the ESP32 CAM module.

Security Conditions: In case of using an external camera such as ESP32, since the camera will be connected via Wi-Fi when connecting to the system, other devices using the same Wi-Fi must not block or access the camera. For this, the security of the camera is provided via Wi-Fi.

It is assumed that the system should be trained and tested using the sources in the license plate recognition system available on the internet, different vehicle images and the libraries such as TensorFlow library dataset.

4.3 Specific Requirements

4.3.1 Interface Requirements

4.3.1.1 User Interface

Since the users will use more compilers in the License Plate Recognition System, the recognized license plates on the vehicles and their outputs will be displayed here.

Requirement 1: Running the Compiler
In this interface, the user can operate the system with the buttons on the compiler and observe the outputs.

4.3.1.2 Hardware Interface

Requirement 1: For the License Plate Recognition System, a computer that can train the data in deep learning and that has good enough hardware should be used.

Requirement 2: Thanks to the ESP32 camera module as an external camera, video transfer is carried out over Wi-Fi. The video taken from the camera reaches the computer and plate recognition is performed. The connection between the ESP32 camera module and the computer is provided by the FTDI module. In case the ESP32 is used other than the computer, the battery is used as the power supply and the cables are used for the connection.

4.3.1.3 Software Interfaces

Requirement 1: As operating systems, any Windows, Linux and MacOS operating systems with the latest version are chosen as much as possible for the best support and ease of use.

Requirement 2: Libraries such as TensorFlow and OpenCV are planned to be used for the license plate recognition system. For this reason, compilers that support these libraries such as Visual Studio, PyCharm, Spyder and compilers that support Python language are used, and Arduino IDE is used for camera connection.

4.3.1.4 Communication Interfaces

Requirement 1: In case of using the external camera, the network connection is required to be stable in order to establish the connection properly.

4.3.2 Detailed Description of Functional Requirements

By using the deep learning method in the License Plate Recognition System, it provides license plate recognition by training the pictures of the vehicles with their license plates from different angles and frames. Thus, high accuracy results are obtained by minimizing the error rate and false results. In the License Plate Recognition System, a picture is taken first, and the location of the plate is determined in the picture taken. Then, character segmentation is done and finally the process is completed by character recognition.

Requirement 1: The user operates and uses the License Plate Recognition System.

Requirement 2: Pictures or videos are uploaded to the License Plate Recognition System.

Requirement 3: For the License Plate Recognition System, firstly, the defined pictures or videos are trained to detect different angles and frames by deep learning method.

Requirement 4: Vehicle image defined in License Plate Recognition System is converted to grayscale format.

Requirement 5: The image is binarized to reveal the plate.

Requirement 6: In the License Plate Recognition System, all the contours of the vehicle picture are found.

Requirement 7: The bounding rectangle of each contour in the vehicle image is found.

Requirement 8: In the License Plate Recognition System, a comparison and verification of the side ratio and area of each bounding rectangle with an average plate in the vehicle image is made.

Requirement 9: Image segmentation is applied to the image within the verified contour to find the characters in the license plate found in the system.

Requirement 10: In the Plate Recognition System, the characters are recognized separately by using OCR (Optic Character Recognition) as the last process.

Requirement 11: Users can see the license plates of the defined vehicles as a result of the operations in the License Plate Recognition System.

Requirement 12: Support System Manager assists the user and fixes any problem that occurs during the operation of the License Plate Recognition System.

4.3.3 Non-Functional Requirements

Requirement 1: Speed

In the License Plate Recognition System, it is important that there is a minimum delay in the image source.

Requirement 2: Size

According to the size of the images in the Plate Recognition System and the size of the libraries used, it will occupy a certain space on the computer.

Requirement 3: Performance

In the License Plate Recognition System, the performance of the recognition process of the license plates may vary depending on the camera or the current operating speed of the computer.

Requirement 4: Usability

In order to use the deep learning method in the License Plate Recognition System, certain libraries must be on the computer and the images must be transferred to the computer to be run.

Requirement 5: Reliability

If an external camera such as ESP332 is used in the License Plate Recognition System, since the connection of this camera with the computer is done via Wi-Fi, necessary protection measures must be taken over Wi-Fi.

Requirement 6: Robustness

License plate recognition shall be done according to changing ambient conditions and ambient light.

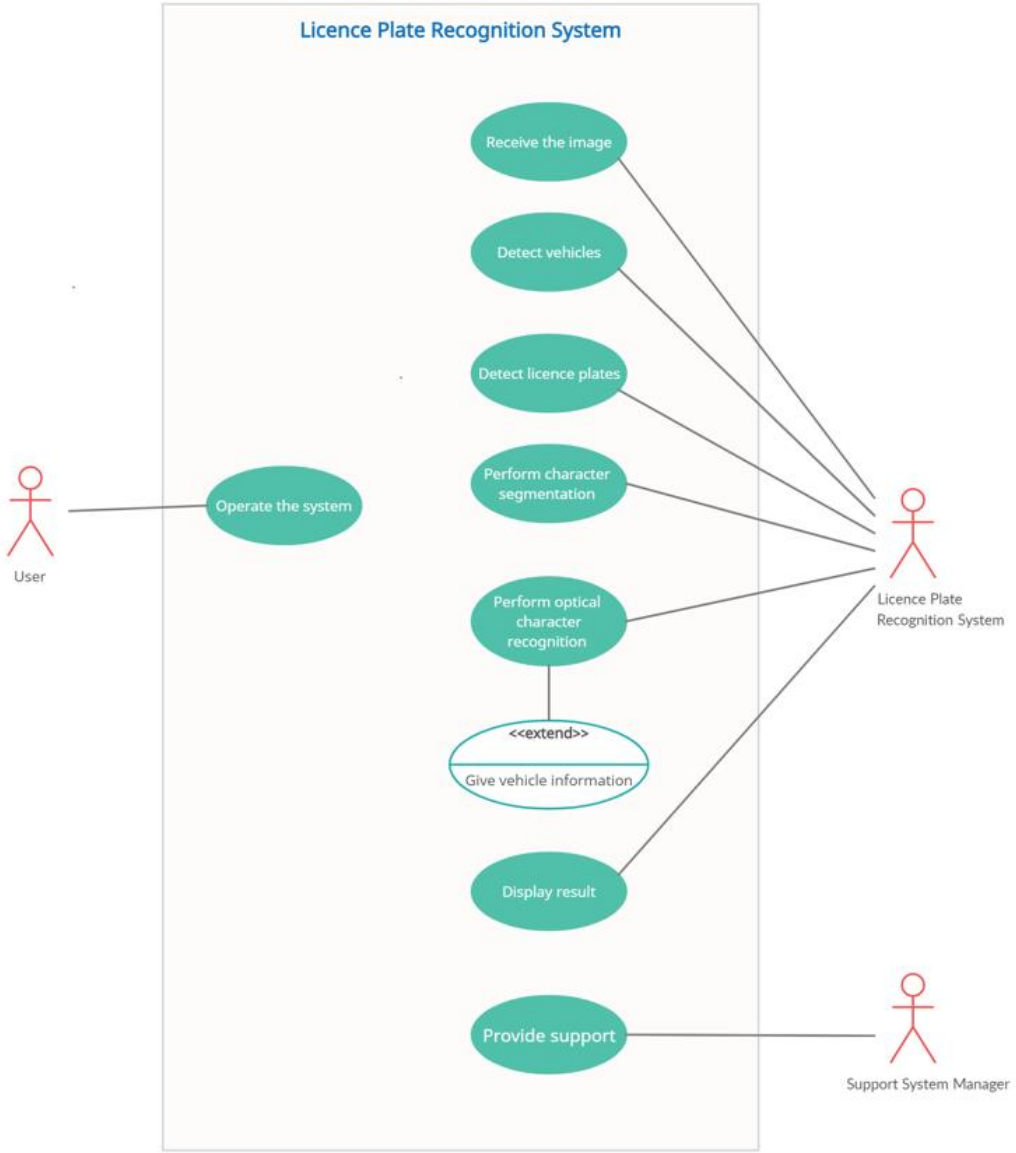
Requirement 7: Portability

License Plate Recognition System can be used in any computer environment.

4.4. Analysis –UML

4.4.1 Use cases

4.4.1.1 Drawing Use Case Diagram



4.4.1.2 Describing Most Use Cases

4.4.1.2.1 Operate The System

Use Case Name	Operate The System
Actor	Users, Licence Plate Recognition System
Description	1. It is necessary to start the software from the compiler.
Trigger	Compile the program
Preconditions	Users starts the Licence Plate Recognition System project source code.
Postconditions	After users start the system, they can observe the results.

4.4.1.2.2 Receive The Image

Use Case Name	Receive The Image
Actor	Licence Plate Recognition System
Description	1. The system compiles images from vehicle images and video frames in the IDE.
Trigger	Receiving the image.
Preconditions	Licence Plate Recognition system gets the image.
Postconditions	Pictures are made ready to apply the operations by the License Plate Recognition System.

4.4.1.2.3 Detect Vehicles

Use Case Name	Detect Vehicles
Actor	Licence Plate Recognition System
Description	1. The processed image is converted to grayscale format and the image is blurred. 2. The blurred image has vertical edges. 3. The image is binarized to reveal the plate. 4. All contours in the picture are found.
Trigger	Process the image.
Preconditions	The system detects the vehicle by using software.
Postconditions	After the vehicle is detected, the license plate recognition phase starts in the License Plate Recognition System.

4.4.1.2.4 Detect Licence Plates

Use Case Name	Detect Licence Plates
Actor	Licence Plate Recognition System
Description	1. The transformation is applied to the thresholded image. 2. The rectangular white box of the plate is revealed. 3. Contours are drawn in the binary and transformed image. 4. By drawing all the extracted contours to the original image, the boundaries of the license plate and vehicle are determined. 5. For the plates, the minimum area rectangle enclosed by each contour is found and the side proportions and area are verified. 6. The minimum and maximum area of the plate is defined according to the plate to be used. 7. The plate has contours in the verified region, and the lateral proportions and area of the bounding rectangle of the largest contour in that region are verified. 8. The found contour is subtracted from the original image and the view of the plate is obtained.
Trigger	Detection of vehicles
Preconditions	Determination of the contours of vehicles.
Postconditions	Character segmentation process is started.

4.4.1.2.5 Perform Character Segmentation

Use Case Name	Perform Character Segmentation
Actor	Licence Plate Recognition System
Description	1. To fully recognize the characters on the plate, image segmentation needs to be applied and the value channel is extracted from the HSV format of the plate image. 2. Adaptive thresholding is applied to the value channel image of the plate to binarize and reveal characters. 3. In order to find the connected components in the image, not bitwise processing is applied on the image. 4. A mask is created to display all character components and their contours in the mask. By subtracting the contours, the largest is taken, the bounding rectangle is found, and the side ratios are verified. 5. By verifying the lateral proportions, the convex shell of the contour is found and drawn on the character candidate mask. 6. All contours in the character candidate mask are found and these contour areas are extracted from the threshold value image of the plate, all characters are taken separately.

Trigger	Detect Licence Plates
Preconditions	Defining the plate perimeter and contour.
Postconditions	The optical character recognition stage is applied.

4.4.1.2.6 Perform Optical Character Recognition

Use Case Name	Perform Optical Character Recognition
Actor	Licence Plate Recognition System
Description	1. Plate texts are obtained from the picture with character segmentation.
Trigger	Completion of character segmentation.
Preconditions	Obtaining the characters on the plate separately as a picture.
Postconditions	Determination of vehicle type and features and recognition of license plate.

4.4.1.2.7 Give Vehicle Information

Use Case Name	Give Vehicle Information
Actor	Licence Plate Recognition System
Description	1. Providing vehicle type and features by using plates.
Trigger	End of Optical Character Recogniton.
Preconditions	Completion of license plate recognition processes.
Postconditions	Proceeding to the display stage.

4.4.1.2.8 Display Result

Use Case Name	Display Result
Actor	Licence Plate Recognition System
Description	1. After the license plate recognition process is completed, the plate is printed in text format.
Trigger	End of Optical Character Recognition processes.
Preconditions	Vehicle information is defined.
Postconditions	Printing the License Plate Recognition System.

4.4.1.2.9 Provide Support

Use Case Name	**Provide Support**
Actor	Support System Manager, Licence Plate Recognition System
Description	1. System problems are addressed and necessary solutions are realized.
Trigger	Error message.
Preconditions	Error occurred in the system.
Postconditions	Support system manager solves the problem.

5. Software Design Description

5.1 Introduction

5.1.1 Purpose

This software design document describes the License Plate Recognition System architecture and system design. This includes the architectural features of the system which operations each module will perform. The aim is to guide a design that can be easily implemented by any user reading this report. The primary audience for this document is License Plate Recognition System users and developers.

5.1.2 Overview

Each chapter contains the following topics in order.
Section 5.2 introducing the system context and design, and discussion the background to the project.
Section 5.3.1 is the Architectural Design, which determines the design to perform all functions included in the system. Each of these entities has a brief description concerning the services that it provides to the rest of the system.
Section 5.3.2 includes the providing decomposition of software components, including the hierarchy and control and data flows.
Section 5.4 discusses the User Interface Design, and how it can be created with maximum user efficiency and ease of use.

5.1.3 Definitions and Acronyms

Term	Definition
Activity Diagram	Activity diagrams are graphical representations of workflows with support for choice, iteration, and concurrency in stepwise activities and actions. Activity diagrams are used in the Unified Modeling Language to model both computational and organizational processes, as well as data flows that cross with the linked activities.
Architectural Design	The process of establishing the foundation for the creation of a computer system by defining a collection of hardware and software components, as well as their interfaces.
Camera	A device for recording visual images in the form of photographic, film or video signals.
Class Diagram	A class diagram is a form of static structural diagram that depicts a system's structure by displaying the system's classes, their properties, operations, and object relationships.
Character Recognition	Character recognition is a method for computers to recognize written or printed characters like numbers or letters and convert them into a format that the computer can understand.
Character Segmentation	Character segmentation is a process that attempts to break down a picture of a series of characters into individual symbol subimages. It's one of the decision-making stages of an optical character recognition system (OCR).
Decomposition Diagram	A decomposition diagram depicts the breakdown of a complicated entity, such as a process, organization, data topic area, or other sort of object, into lower level, more specific components. Division diagrams, for example,

	can depict organizational structure or the decomposition of functions into processes.
Interface	A shared boundary across which two or more different components of a computer system exchange information is referred to as an interface.
Licence Plate	A license plate is a sign that is placed on the front and back of a car and displays the vehicle's license number. A rectangular, generally metal plate with a series of numbers, letters, or both that is issued by a government to identify a legally registered vehicle one of the signs with numbers on it at the front and back of a car.
Licence Plate Recognition System	A license plate recognition system is a form of technology, primarily software, that allows computers to read the registration number (license number) of automobiles from digital images automatically.
Optical Character Recognition (OCR)	The electronic or mechanical translation of images of typed, handwritten, or printed text into machine-encoded text, whether from a scanned document, a photo of a document, a scene photo, or subtitle text superimposed on an image, is known as optical character recognition or optical character reader.
Software Design Description	A software design description is a visual depiction of a software design that will be used to keep track of design information, handle various design problems, and communicate that information to the design.
Support System Manager	Someone who helps the License Plate Recognition System when there is a problem with the system, camera or license plate reading.
Users	The person using the License Plate Recognition

	System.
User Interface Design	Predicts what users will need to do and ensures that the interface has features that are easy to access, understand, and utilize to assist them in doing those activities. UI combines interaction design, graphic design, and information architecture.

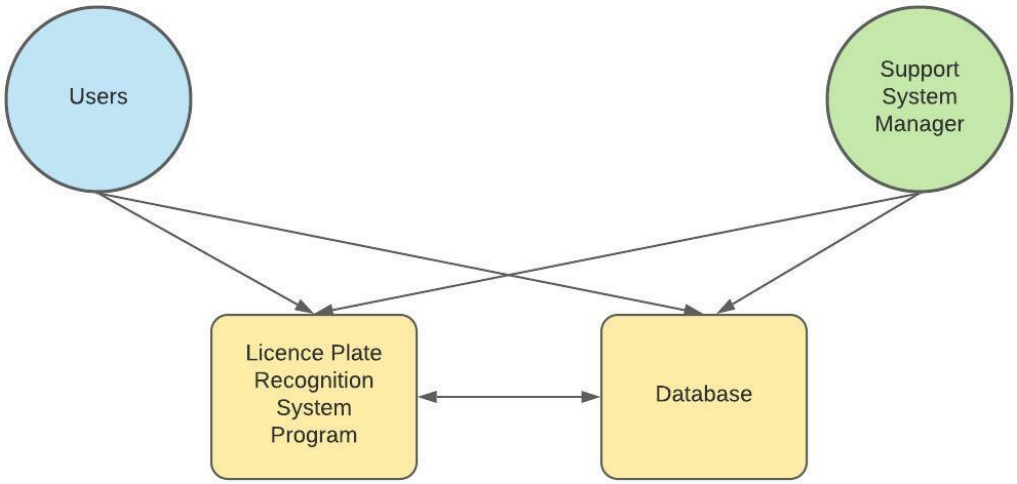
5.2. System Overview

The license plate recognition system is made by detecting the license plate region of the vehicles with the vehicle image and reading and separating the characters on the plate with image processing. By developing the license plate recognition system with deep learning and image processing, it will be tried to work towards detecting license plate recognition.

In the Plate Recognition System, the recognition process of the plates first takes place by finding all the contours in the picture. It happens that each stroke has its bounding rectangle. Then it has to compare and verify the side ratio and area of each bounding rectangle with an average plate. Image segmentation is then applied to the image within the verified contour to find the characters inside the plate. As the last step, the characters on the plate are recognized by using OCR (Optical Character Recognition). In the license plate recognition system, there is also a training phase by using the deep learning method for more recognition of the license plate in the images. Thus, different directions of vehicles and license plate positions can be better detected using multiple images or different sections of a video. In this way, it also provides convenience in terms of accuracy and speed in plate recognition.

5.3 System Design

5.3.1 Architectural Design

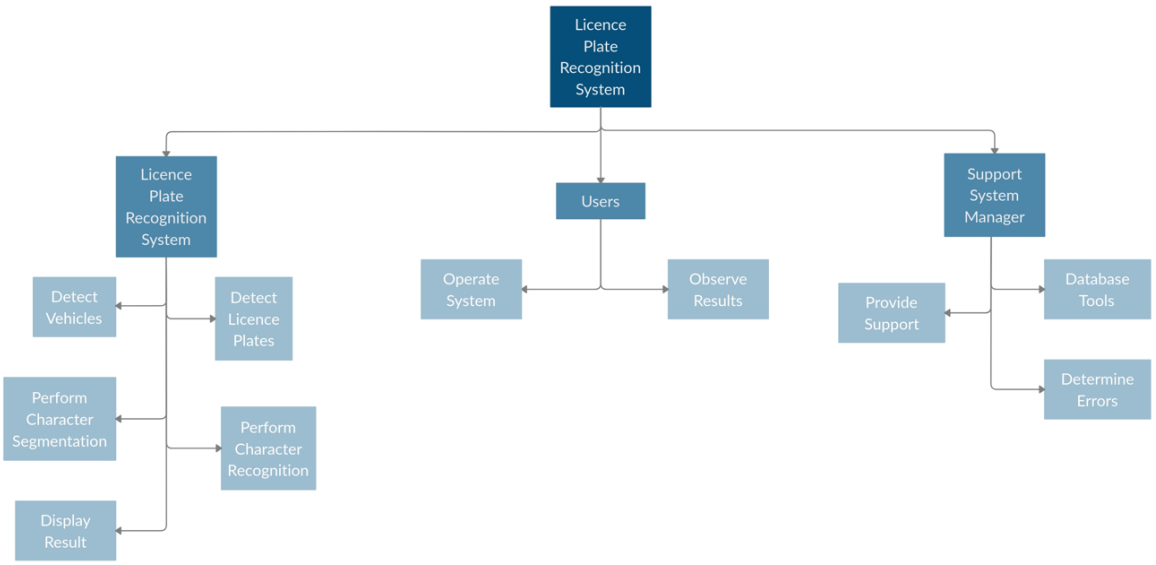


1. Licence Plate Recognition System Program: It is the part where the users run the License Plate Recognition System project and the whole project is located.

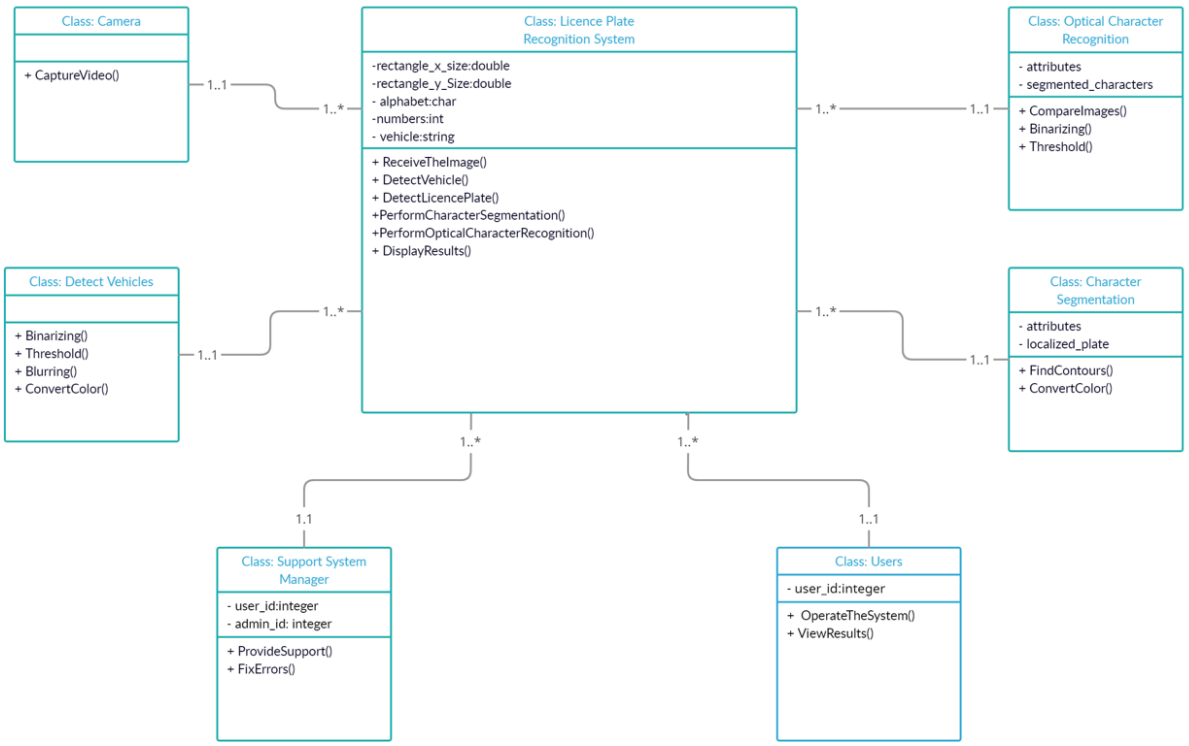
2. Database: It is the place where license plate information is stored in the License Plate Recognition System project.

5.3.2 Decomposition Description

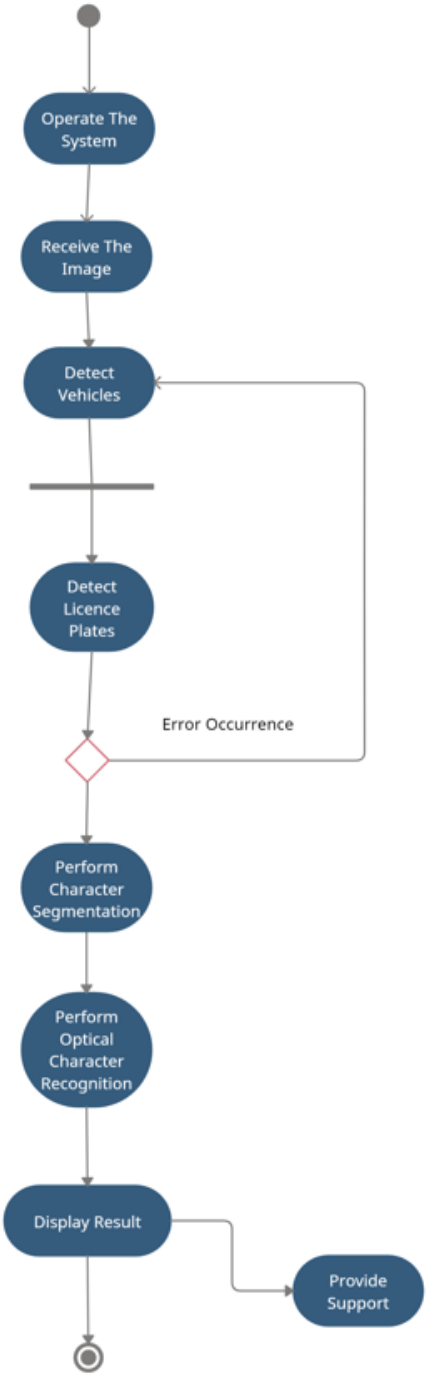
Decomposition Diagram:



Class Diagram:

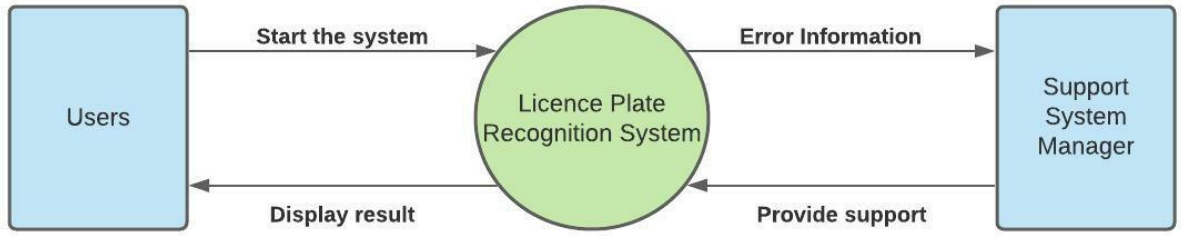


Activity Diagram:

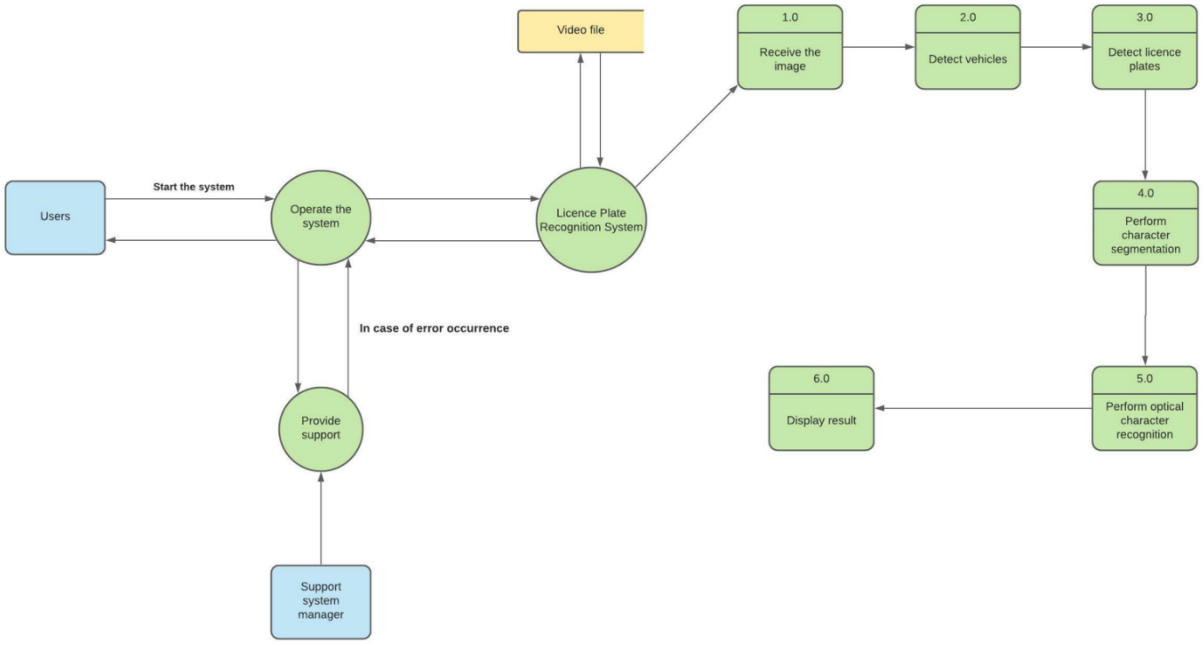


Data Flow Diagram (DFD):

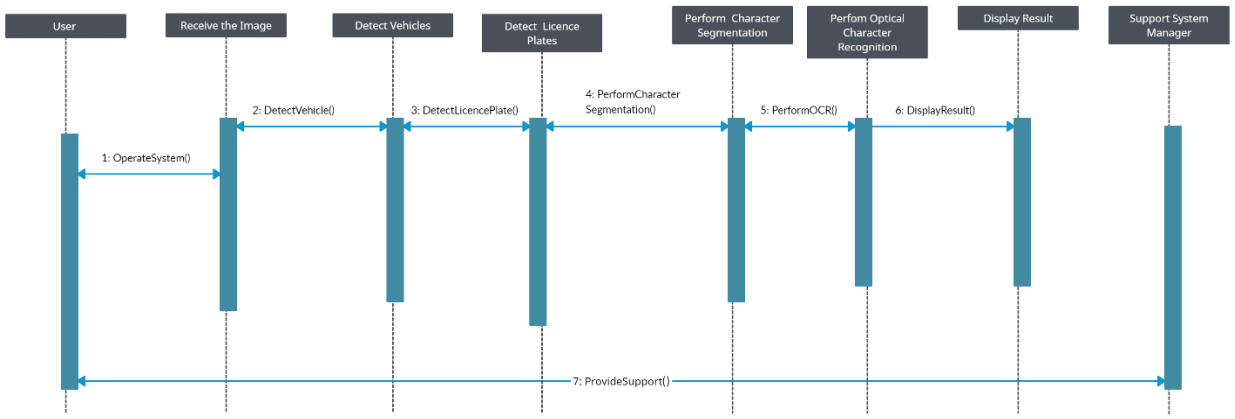
LEVEL 0:



LEVEL 1:

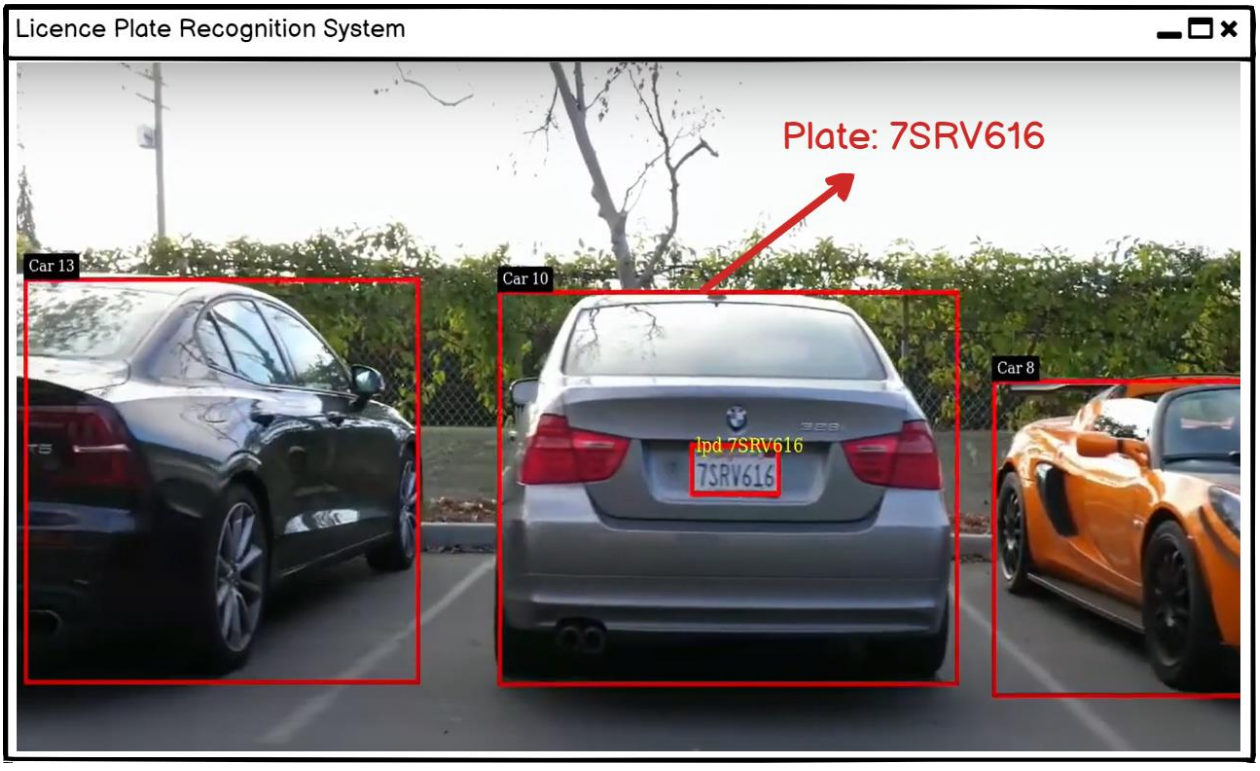


Sequence Diagram:



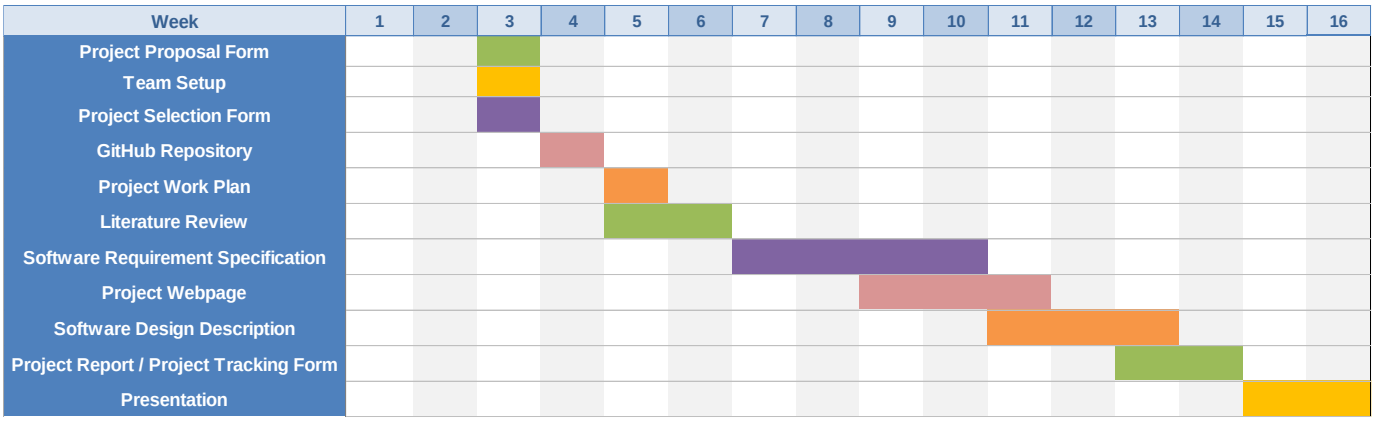
5.4. User Interface Design

- **Display Result**



After users run the License Plate Recognition System, the system performs license plate recognition. The license plate of the vehicles is recognized through the image taken from the video. After the users run the program, the result of the license plate recognition process is written on the screen as in the interface. In this way, users can clearly see the result of the license plate recognition process.

6. Project Work Plan



7. Test Plan

7.1 INTRODUCTION

7.1.1 Version Control

Version No	Description of Changes	Date
1.0	First Version	April 1, 2022

7.1.2 Overview

Our project is License Plate Recognition System. The use cases of the system are explained in the SRS document. Accordingly, the features required for license plate recognition will be tested.

7.1.3 Scope

Scope of this document specifies test plan specifications according to the use cases found in the Software Requirements Specification Document. This document includes features that should be tested, features that will not be tested, test design features, and detailed test cases of use cases according to the test plan.

7.1.4 Terminology

Acronym	Definition
SRS	Software Requirements Specification
SDD	Software Design Document
CS	Character Segmentation

Acronym	Definition
Licence Plate	A license plate is a sign that is placed on the front and back of a car and displays the vehicle's license number.
Licence Plate Recognition System	Technology that allows computers to read the license number of vehicles.
OCR	Optical Character Recognition
TP	Test Plan

7.2 FEATURES TO BE TESTED

This section lists and gives a brief description of all the main features of our project that will be tested. For each major feature there will be a Test Design Specification appended to the end of this document.

7.2.1 Detection Vehicles (DV)

License Plate Recognition System can recognize vehicles in order to detect license plates.

7.2.2 Detection Licence Plates (DLP)

License Plate Recognition System can detect plate through the software used in the system.

7.2.3 Optical Character Recognition (OCR)

License Plate Recognition System detects the characters on the plate using OCR method in plate detection.

7.3 FEATURES NOT TO BE TESTED

7.3.1 Working in Different Environmental Conditions

Since the License Plate Recognition System will take images with the camera and transfer them to the computer, it can detect license plates when the ambient conditions are good. Therefore, the operation of the License Plate Recognition System in different environmental and weather conditions will not be tested.

7.4 ITEM PASS/FAIL CRITERIA

License Plate Recognition System performs plate recognition by taking images in real time with the camera. If the camera is disconnected, it cannot detect the license plate due to the lack of image transmission.

7.4.1 Exit Criteria

- 100% of the test cases are executed
- 90% of the test cases passed
- All High and Medium Priority test cases passed

7.5 REFERENCES

[1] CENG408_Group21_SRS, April 1, 2022. Available https://github.com/CankayaUniversity/ceng-407-408-2021-2022-Licence-Plate-Recognition-System/wiki/Software-Requirements-Specification

[2] CENG408_Group21_SDD, April 1, 2022. Available https://github.com/CankayaUniversity/ceng-407-408-2021-2022-Licence-Plate-Recognition-System/wiki/Software-Design-Description

[3] CENG408_Group21_ProjectReport, April 1, 2022. Available https://github.com/CankayaUniversity/ceng-407-408-2021-2022-Licence-Plate-Recognition-System/blob/main/CENG%20407%20Licence%20Plate%20Recognition%20System%20Project%20Report.pdf

7.6 TEST DESIGN SPECIFICATIONS

7.6.1 Detection Vehicles (DV)

7.6.1.1 Subfeatures To Be Tested

7.6.1.1.1 Camera Connection (DV.CC)

In order to detect vehicles in the License Plate Recognition System, a real-time connection is provided with the ESP32 camera module using Wi-Fi. After the camera connection is established, image transfer is provided and vehicle detection becomes possible.

7.6.1.1.2 Vehicle Size (DV.VS)

While detecting vehicles in the License Plate Recognition System, whether the objects are vehicles or not is determined according to their dimensions and features.

7.6.1.2 Test Cases

Here are the system's test cases for the Detection Vehicles feature.

TC ID	Requirements	Priority	Scenario Description
DV.CC.01	3.1.2.2	H	ESP32 camera module works by providing Wi-Fi connection.
DV.CC.02	3.1.2.2	H	Detection software works over real time video with Wi-Fi connection.

TC ID	Requirements	Priority	Scenario Description
DV.VS.01	3.2.6	H	Real time video image works with ESP32 camera module.
DV.VS.02	3.2.6	H	Vehicle recognition is performed by dividing them by size from other objects.

7.6.2 Detection Licence Plates (DLP)

7.6.2.1 Subfeatures To Be Tested

7.6.2.1.1 Camera Connection (DLP.CC)

Real-time connection is provided with the ESP32 camera module using Wi-Fi for plate detection. After the camera connection is established, image transfer is provided and plate detection becomes possible.

7.6.2.1.2 Licence Plate Size (DLP.LPS)

After the vehicle is detected, the license plate recognition process is performed by considering the smallest rectangle with the plate size.

7.6.2.2 Test Cases

Here are the system's test cases for the Detection Licence Plates feature.

TC ID	Requirements	Priority	Scenario Description
DLP.CC.01	3.1.2.2	H	ESP32 camera module works by providing Wi-Fi connection.
DLP.CC.02	3.1.2.2	H	Detection software works over real time video with Wi-Fi connection.

TC ID	Requirements	Priority	Scenario Description
DLP.LPS.01	3.2.8	H	License plate recognition is performed to detect the smallest rectangle on vehicles.

7.6.3 Optical Character Recognition (OCR)

7.6.3.1 Subfeatures To Be Tested

7.6.3.1.1 Character Segmentation

After the license plate recognition process is performed in the License Plate Recognition System, the parts containing the characters are separated by the character segmentation method.

7.6.3.2 Test Cases

Here are the system's test cases for the Optical Character Recognition (OCR) feature.

TC ID	Requirements	Priority	Scenario Description
OCR.CS.01	3.2.10	H	Image segmentation is applied to the image inside the plate to find the characters within the detected plate.
OCR.CS.02	3.2.10	H	Characters are recognized using the OCR method.

7.7 Detailed Test Cases

7.7.1 DV.CC.01

TC_ID	DV.CC.01
Purpose	Provide the Wi-Fi connection for the ESP32 camera module to work.
Requirements	3.1.2.2
Priority	High.
Estimated Time Needed	20 Seconds
Dependency	Internet connection required.
Setup	ESP32 camera module and computer should be ready for operation.
Procedure	[A01] Provide internet connection for computer.
	[A02] Power up the ESP32 camera module.
	[A03] Connect the computer and the ESP32 camera module with a Wi-Fi connection.
	[V01] Observe that the ESP32 camera module is running.
Cleanup	Turn off ESP32 camera module and Wi-Fi connection.

7.7.2 DV.CC.02

TC_ID	DV.CC.02
Purpose	Run the detection software over live video.
Requirements	3.1.2.2
Priority	High.
Estimated Time Needed	20 Seconds
Dependency	The ESP32 camera module and computer must be connected and ready to work.
Setup	ESP32 camera module, computer and software should be ready for operation.

Procedure	[A01] Video transfer takes place from the ESP32 camera module to the computer.
	[A02] Real time video is taken from the computer.
	[A03] The software for vehicle detection works.
	[V01] Observe that vehicle detection is successful.
Cleanup	Close the software.

7.7.3 DV.VS.01

TC_ID	DV.VS.01
Purpose	Provide live video transmission via ESP32 camera module.
Requirements	3.2.6
Priority	High.
Estimated Time Needed	20 Seconds
Dependency	Internet connection required.
Setup	ESP32 camera module and computer should be ready for operation.
Procedure	[A01] Provide internet connection for computer.
	[A02] Power up the ESP32 camera module.
	[A03] Connect the computer and the ESP32 camera module with a Wi-Fi connection.
	[A04] Video transfer takes place from the ESP32 camera module to the computer.
	[A05] Real time video is taken from the computer.
	[V01] Observe live video streaming.
Cleanup	Turn off ESP32 camera module and Wi-Fi connection.

7.7.4 DV.VS.02

TC_ID	DV.VS.02
Purpose	Perform vehicle recognition.
Requirements	3.2.6
Priority	High.
Estimated Time Needed	25 Seconds
Dependency	The DV.VS.01 operation should be successful.
Setup	ESP32 camera module, computer and software should be ready for operation.
Procedure	[A01] Provide video transmission with the ESP32 camera module.
	[A02] The vehicle image is converted to gray scale format.
	[A03] The image is binarized to reveal the plate.
	[A04] All contours of the vehicle image are found.
	[V01] Observe that vehicle recognition is successful.
Cleanup	Close the software.

7.7.5 DLP.CC.01

TC_ID	DLP.CC.01
Purpose	Provide the Wi-Fi connection for the ESP32 camera module to work.
Requirements	3.1.2.2
Priority	High.
Estimated Time Needed	20 Seconds
Dependency	Internet connection required.
Setup	ESP32 camera module and computer should be ready for operation.
Procedure	[A01] Provide internet connection for computer.
	[A02] Power up the ESP32 camera module.
	[A03] Connect the computer and the ESP32 camera module with a Wi-Fi connection.
	[V01] Observe that the ESP32 camera module is running.
	Turn off ESP32 camera module and Wi-Fi connection.
	20 Seconds
Cleanup	Internet connection required.

7.7.6 DLP.CC.02

TC_ID	DLP.CC.02
Purpose	Run the detection software over live video.
Requirements	3.1.2.2
Priority	High.
Estimated Time Needed	20 Seconds
Dependency	The ESP32 camera module and computer must be connected and ready to work.
Setup	ESP32 camera module, computer and software should be ready for operation.
Procedure	[A01] Video transfer takes place from the ESP32 camera module to the computer.
	[A02] Real time video is taken from the computer.
	[A03] The software for vehicle detection works.
	[V01] Observe that vehicle detection is successful.
Cleanup	Close the software.

7.7.7 DLP.LPS.01

TC_ID	DLP.LPS.01
Purpose	Perform licence plate recognition.
Requirements	3.2.8
Priority	High.
Estimated Time Needed	25 Seconds
Dependency	The DLP.CC.01 and DLP.CC.02 operations should be successful.
Setup	ESP32 camera module, computer and software should be ready for operation.
Procedure	[A01] Provide video transmission with the ESP32 camera module.
	[A02] Provide DV.VS.02 operation.
	[A03] The bounding rectangle of each contour in the vehicle image is found.
	[A04] Comparisons are made according to average plate sizes.
	[A04] Plate detection is provided by taking the smallest rectangle.

	[V01] Observe that plate recognition is successful.
Cleanup	Close the software.

7.7.8 OCR.CS.01

TC_ID	OCR.CS.01
Purpose	Provide image segmentation for character recognition.
Requirements	3.2.10
Priority	High.
Estimated Time Needed	20 Seconds
Dependency	Vehicle detection and license plate detection operations should be carried out.
Setup	ESP32 camera module, computer and software should be ready for operation.
Procedure	[A01] Provide video transmission with the ESP32 camera module.
	[A02] Provide DV.VS.02 and DLP.LPS01 operations.
	[A03] Image segmentation is applied to the image within the validated contour.
	[V01] Observe that image segmentation is successful.
Cleanup	Close the software.

7.7.9 OCR.CS.02

TC_ID	OCR.CS.02
Purpose	Enable character recognition with OCR.
Requirements	3.2.10
Priority	High.
Estimated Time Needed	20 Seconds
Dependency	OCR.CS.02 operation should be carried out.
Setup	ESP32 camera module, computer and software should be ready for operation.
Procedure	[A01] Provide OCR.CS.01 operation.
	[A02] OCR method is used.
	[A03] Characters are recognized separately.
	[V01] Observe that the characters on the license plates of the vehicles.
Cleanup	Close the software.

7.8 Test Results

7.8.1 Individual Test Results

TC ID	Priority	Date Run	Run By	Result	Explanation
DV.CC.01	H	22.04.2022	Mehmet Furkan Turan	Pass	Provide the Wi-Fi connection for the ESP32 camera module to work.
DV.CC.02	H	22.04.2022	Burak Çetin	Pass	Run the detection software over live video.
DV.VS.01	H	29.04.2022	Doğukan Tutak	Pass	Provide live video transmission via ESP32 camera module.
DV.VS.02	H	29.04.2022	Arda Kayış	Pass	Perform vehicle recognition.
DLP.CC.01	H	29.04.2022	Muhammed Emin Atalık	Pass	Provide the Wi-Fi connection for the ESP32 camera module to work.
DLP.CC.02	H	29.04.2022	Doğukan Tutak	Fail	Run the detection software over live video.
DLP.LPS.01	H	06.05.2022	Mehmet Furkan Turan	Pass	Perform licence plate recognition.
OCR.CS.01	H	06.05.2022	Burak Çetin	Pass	Provide image segmentation for character recognition.
OCR.CS.02	H	06.05.2022	Arda Kayış	Pass	Enable character recognition with OCR.

7.8.2 Summary of Test Results

Priority	Number of TCs	Executed	Passed
H	9	9	8
M	0	0	0
L	0	0	0
Total	**9**	**9**	**8**

We have executed 9 test cases and 8 test cases are passed. Also, 8 of high priority test cases are passed. Exit criteria is met.

7.8.3 Exit Criteria

We have executed all test cases and 88% of test cases are passed. Software development activities are completed within the expected timeline. Exit criteria is met.

Criteria	Met or Not
100% of the test cases are executed	M
85% of the test cases passed	M
All High Priority test cases passed	N

7.8.4 Known Problems

Optimization of the camera connection for the detection algorithm could not be fully achieved. Therefore, the test case failed. The problem will be solved when the optimization between the camera and the detection software is achieved.

7.8.5 Conclusion

This section contains the test results of the “License Plate Recognition System” project. Test cases have been applied and 88% of test cases have been successfully completed.

The current phase of the project is available for use. Failed test cases to improve quality will be resolved over time.

8. Conclusions

The design and development of an automated license plate recognition system requires careful selection of specifications and techniques to work with different operating and hardware constraints. We have briefly summarized the history of the License Plate Recognition System in the Literature Review. Thus, we observed how and why the License Plate Recognition System came from the past to the present. We examined in detail the research and analysis of current approaches and techniques used in the solution of the Plate Recognition System. We have discussed in detail the main methods required for the License Plate Recognition System. We examined these methods in detail under the titles of License Plate Detection and License Plate Recognition. We also explored the open challenges of the License Plate Recognition System and suggested solutions to these challenges.

Based on our research, we aim to determine the features required for the recognition of license plates by using image processing, image detection and recognition methods for the License Plate Recognition System. In addition to these methods, we aim to make the Plate Recognition System by using the deep learning method with libraries such as TensorFlow.

Since it is intended to use deep learning, image optimization, camera adjustment and image processing, data processing speeds can be seen as potential problems. With more specific algorithms and methods, the license plate recognition system can be made in accordance with speed and time. Thus, we aim to perform the process of recognizing the license plates of the vehicles by taking images from the camera by using the libraries and methods required for the License Plate Recognition System.

In the License Plate Recognition System, we provided the connection between the ESP32 CAM camera hardware and the computer, and enabled the license plate recognition and car detection software to run on the camera. In addition, we performed car detection and license plate recognition operations on the computer using the recorded videos and pictures.

Acknowledgement

We would like to express our appreciation to our advisor Prof. Dr. Ahmet Coşar because of his guidance and contributions.

References

[1] A Review Paper on License Plate Recognition System
a-review-paper-on-license-plate-recognition-system.pdf (imedpub.com)

[2] The History of License Plate Recognition Technology
The History of License Plate Recognition Technology | NGSC (ngscinc.com)

[3] History of Automatic Number-Plate Recognition
https://en.wikipedia.org/wiki/Automatic_number-plate_recognition

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

[5] Atul Patel Chirag Patel Dipti Shah 2013 Automatic Number Plate Recognition System (ANPR): A Survey International Journal of Computer Applications Volume 69– No.9 pp. (0975 – 8887).

[6] Automatic number plate recognition using deep learning
https://iopscience.iop.org/article/10.1088/1757-899X/1084/1/012027/pdf

[7] R. Zunino and S. Rovetta, ‘‘Vector quantization for license-plate loca-tion and image coding,’’ IEEE Trans. Ind. Electron., vol. 47, no. 1,pp. 159–167, Feb. 2000.

[8] What is Image Segmentation?
https://www.analytixlabs.co.in/blog/what-is-image-segmentation/

[9] Texture image analysis and texture classification methods
https://arxiv.org/abs/1904.06554

[10] H.-K. Xu, F.-H. Yu, J.-H. Jiao, and H.-S. Song, ‘‘A new approach of the vehicle license plate location,’’ in Proc. 6th Int. Conf. Parallel Distrib. Comput. Appl. Technol. (PDCAT), 2005, pp. 1055–1057
[11] J. Matas and K. Zimmermann, ‘‘Unconstrained licence plate and text localization and recognition,’’ in Proc. IEEE Intell. Transp. Syst., Sep. 2005, pp. 225–230

[12] A deep learning-based method for vehicle licenseplate recognition in natural scene
https://www.cambridge.org/core/journals/apsipa-transactions-on-signal-and-information-processing/article/deep-learningbased-method-for-vehicle-licenseplate-recognition-in-natural-scene/5C9EE6C71559B5E0B2EA1B89401A068C

[13] S. Saha, ‘‘A review on automatic license plate recognition system,’’MCKV Inst. Eng., Howrah, India, Tech. Rep. 711204, Feb. 2019.
[14] X. Xu, Z. Wang, Y. Zhang, and Y. Liang, ‘‘A method of multi-view vehiclelicense plates location based on rectangle features,’’ in Proc. 8th Int. Conf.Signal Process., vol. 3, 2006, pp. 1–4.

[15] M.-S. Pan, J.-B. Yan, and Z.-H. Xiao, ‘‘Vehicle license plate character seg-mentation,’’ Int. J. Autom. Comput., vol. 5, no. 4, pp. 425–432, Oct. 2008.

[16] M.-S. Pan, Q. Xiong, and J.-B. Yan, ‘‘A new method for correcting vehiclelicense plate tilt,’’ Int. J. Autom. Comput., vol. 6, no. 2, pp. 210–216,May 2009

[17] S. Du, M. Ibrahim, M. Shehata, and W. Badawy, ‘‘Automatic license platerecognition (ALPR): A state-of-the-art review,’’ IEEE Trans. CircuitsSyst. Video Technol., vol. 23, no. 2, pp. 311–325, Feb. 2013

[18] R. Laroca, A. L. Zanlorensi, R. G. Gonçalves, E. Todt, W. R. Schwartz,and D. Menotti, ‘‘An efficient and layout-independent automatic licenseplate recognition system based on the YOLO detector,’’ Sep. 2019,arXiv:1909.01754. [Online]. Available: https://arxiv.org/abs/1909.01754

[19] https://dergipark.org.tr/tr/download/issue-file/34765

[20] “Chart & Diagram Drawing Tool”, [Online]. Available:
https://creately.com

[21] “Interface Drawing Tool”, [Online]. Available:
https://balsamiq.cloud/s1drwsu/pnckn7k

[22] “Sample Software Design Document,”, [Online]. Available: https://arxiv.org/ftp/arxiv/papers/1005/1005.0595.pdf

[23] “Dictionary”, [Online]. Available: https://dictionary.cambridge.org/

[24] “Chart & Diagram Drawing Tool”, [Online]. Available: https://lucid.app/

[25] “Dictionary”, [Online]. Available: https://www.oxfordlearnersdictionaries.com/

[26] “Software Architecture Examples and Templates”, [Online]. Available: https://www.edrawsoft.com/software-architecture-example.html

[27] “IEEE Standard for Information Technology Systems Design Software Design Descriptions”, [Online]. Available: http://cengproject.cankaya.edu.tr/wp-content/uploads/sites/10/2017/12/SDD-ieee-1016-2009.pdf