ÇANKAYA UNIVERSITY FACULTY OF ENGINEERING COMPUTER ENGINEERING DEPARTMENT

Project Report

Version 2

CENG 408

Innovative System Design and Development I

P2018-08

License Plate Recognition Using Deep Learning

Bengisu Şeyma AYDOĞDU
201411008
Rüştü Kaan Varlı
201411064
Mert Sert
201411052
Mert Yumuşak
201411071

Advisor: Reza HASSANPOUR
Co-Advisor: Erdoğan DOĞDU

Table of Contents

Abstract

Plate recognition systems are widely used around the world to produce automated running solutions in the field of traffic and security. These common systems bring complexity. Current license plate recognition systems use image processing, but deep learning has become the state of art in character recognition. The purpose of License Plate Recognition with Deep Learning is using deep learning techniques for character recognition instead of image processing techniques. Deep learning systems become better at performance as the data used over time increases. Image-based systems produce static solutions and are difficult to improve over time. This is the success of deep learning methods.

Key words:

License Plate Recognition, Image Processing, Deep Learning

Özet:

Plaka tanıma sistemleri, trafik ve güvenlik alanında otomatik çalışan çözümler üretmek için dünya çapında yaygın olarak kullanılmaktadır. Bu ortak sistemler karmaşıklık getiriyor. Mevcut plaka tanıma sistemleri görüntü işlemeyi kullanır, ancak derin öğrenme karakter tanımada son teknoloji haline gelmiştir. Derin Öğrenme ile Plaka Tanıma'nın amacı, görüntü işleme teknikleri yerine karakter tanıma için derin öğrenme teknikleri kullanmaktır. Derin öğrenme sistemleri zamanla kullanılan veriler arttıkça performansta daha iyi hale gelir. Görüntü tabanlı sistemler statik çözümler üretir ve zamanla iyileştirilmesi zordur. Bu, derin öğrenme yöntemlerinin başarısıdır.

Anahtar Kelimeler: Plaka Tanıma Sistemi, Görüntü İşleme, Derin Öğrenme

1. Introduction

1.1 Company Background

Our project partner ISSD company, system design and integration in Turkey, digital signal processing, software development and is a company operating in the electronics design issues, companies in the digital circuit designs and application specific algorithms, starting with developing and developed all the software which ended with the transfer of embedded platforms the whole process is realized. ISSD also manufactures solutions for traffic management, dynamic junction control system, license plate recognition and electronic monitoring and develops products that are market leaders. Solutions for plate recognition and electronic monitoring are generally based on image processing algorithms. The image processing algorithms developed by the company consist of processes that require high processing power as in most image processing algorithms. In addition, other solutions within our knowledge on producing plate recognition systems company in Turkey are developing the system with the image processing methods.

In this project, plate, vehicle and model recognition system will be developed with deep learning methods instead of image recognition algorithms of ISSD firmware. The performance of the new system as a result of this development will be compared with the old system in terms of accuracy and performance.

There are many open source data for plate recognition with deep learning, which is becoming more widespread in the world. However, these data sets cover foreign plates. Within this scope, it is aimed to create a data set suitable for deep plate learning techniques for Turkish license plate standards by using vehicle license plate photographs within ISSD Company.

1.2 Problem Statement

With the increasing number in vehicles, traffic problem has become a major part in our society. The traffic problem in the world is getting bigger, complexity rises despite the work on this field. There are several fields work on traffic problems. With the rise of problems in traffic and today’s technology, solutions come parallel to that. The most important example is that development of smart cities in Europe, dynamic cross roads managements, electronic inspection systems, license plate recognition systems. There are many different areas of the work done with traffic in the world. With the increase in the

problems in traffic and the development of today's technology, common solutions have started to be found. The most important examples are the rapidly developing smart cities in Europe, dynamic junction management, electronic control systems, and license plate recognition systems. The Plate Recognition System is widely used worldwide to produce common solutions in the field of traffic and security. These common systems bring complexity. Nowadays, the technologies that are in place can cause too much cost for plate recognition systems. For example, when we want to set up a plate recognition system in a city, the materials and processes we need to use are increasing too much. This increases the cost. At the same time, License Plate Recognition systems have a high level of accuracy in terms of safety and traffic. Because an unreadable plate greatly increases the safety risk and complexity. The license plate recognition system in Turkey is widely used in security systems and traffic control systems. At the city exits, the shopping center is used for security purposes in all places with a stadium and similar public parking, and in systems such as electronic control systems, for the solution of traffic complexity, and even in the management of customs entry and exit systems to increase security and reduce the traffic complexity. Developing and developing a dataset for Turkish plates for deep learning technique, which is one of the sub-branches of data science which is very developing in the world of computer science, aims to decrease the cost used in Turkish plates and increase the accuracy rate and compare the old techniques.

1.3 Solution Statement

In the current plate recognition systems, image processing algorithms are used. t requires very high processing power, but also the accuracy rate of the plates cannot always give the expected performance. It requires very high processing power, but also the accuracy of the plates cannot always give the expected performance. Developing a new approach to achieving higher accuracy rates with higher performance than existing systems by using deep learning techniques, which is the sub-study area of the widely used data sciences branch, is one of the main objectives of this project. In this project, we aim to compare the existing plate recognition systems with the deep learning-based plate recognition systems and to develop a low cost and high-performance plate recognition system.

1.4 Contribution

In our knowledge of the current license plate recognition system in Turkey there is not a system using deep learning techniques for license plate recognition and there is not a data set and learning model encompassing the Turkish plate standards. Deep learning methods can work at different angles and on different backgrounds, and at the same time adapt to changing lighting automatically. This provides deep learning methods to learn from big data and to have more performance as data increases. We aim to develop a system that uses deep learning techniques for license plate recognition.

2. Literature Search

2.1 Image Processing

Image processing can be clustered into a few groups such as checking for presence, object detection and localization, measurement, identification and verification. In most cases, the acquired image from the camera is not directly processed within the application. Instead, it is preprocessed, to enhance the image according to the specific task. Examples of preprocessing are noise reduction as well as brightness and contrast enhancement. Some of these steps can be done directly by the camera itself, and thus save CPU load on the host side. To use the camera as a measuring device, it must be calibrated to the physical world. [1] Furthermore, we can also determine the relationship between the camera's natural units, meaning pixels and the real-world units, like millimeters or inches, for example. With color calibration we ensure an accurate reproduction of colors. The better the preprocessing, the better the image quality the better the preprocessing, the better the image quality and the results of the image processing within your application. When it comes to locating parts, usually matching is involved. This means looking for regions that are similar to or the same as a predefined template. This template can either be also an image or a geometric pattern which contains information regarding edges and geometric features. These methods are called correlation pattern matching and geometric pattern matching respectively. But for character recognition we use a method called optical character recognition, also called OCR. One way to manage optical character recognition is by separating the characters in the image and comparing them with a set of templates. [2] Afterwards, the software can convert the captured data into editable and searchable data. [1] A popular example for OCR is automatic number

plate recognition, also known as ANPR. Each of these processing techniques covers a wide spectrum of machine vision applications but combining them can give you even more possibilities.

2.2 License Plate Recognition Using Image Processing

Image processing methods requires very high processing power. It has low accuracy. Input: Video frame (frame)

Output:

1. The input video frame is converted into a gray scale format [3].
2. Enhancement [4] algorithms are applied.
3. Edge detection [5] algorithm is applied.
4. With certain location algorithms plate locations will be found
5. With the possible location data 'Y' axis will be checked pixel by pixel in binary
6. When characters has been found in 'Y' axis same operation will be applied on 'X' axis
7. The plate location is found when an array of characters of a certain height and width is found.
8. The characters in the picture are cut to a certain width, from the acquired binary image.
9. These characters are entered into the SVM algorithm [6].
10. These algorithms are applied and completed until the vehicle enters the image and exits the image.
11. It is compared with the best string plate designs.

2.3 Deep Learning

Artificial intelligent is a field of computer engineering which is aims to create machines think, behave and act like human beings. Machine learning is subfield of Artificial intelligent. It creates a software that can predicts outcomes with experience from pervious data. And finally, deep Learning is subfield of machine learning [7]. It is inspired by working principle of human brain. The purpose of using human brain is making learning algorithms easier and better to use. Deep learning has neural networks capable of learning from data like machine learning, but deep learning is more advance with more layers, neurons and large amount of

data. Deep learning models use large set of labeled data and neural networks with many layers. Using labeled data is called supervised learning. Every neuron level selects more acceptable prediction and carry it to next level. Eventually last level come with an outcome. Complexity increase from level to level [7]. Deep learning models learn classifications from images, text and sounds. They can also achieve state-of-the-art accuracy. Deep learning is also known as deep neural network. The “deep” expression was first used by Geoffrey Hinton who is the grandfather of deep learning. “Deep” refers to number of hidden layers in neural networks. Deep learning theorized in 80’s but it recently become popular. There are two main reasons for that. First, it requires larges amount of labeled data. Second, it requires high GPU performance which have parallel architecture. We use deep learning applications almost in every are in our life’s. For example, voice control in electronic devices, phones, automated driving cars, automatically detecting cancer cells, speech translations…

2.4 License Plate Recognition Using Deep Learning

Computers can automatically classify images with deep learning. They can also describe and distinguish objects in pictures. Convolutional Neural Networks are one of the common and effective neural networks to do image processing. It learns patters, takes images and classifies them according to patterns. First it takes images as input data. Then categorize them. Input data needed to label so by time deep learning model will be able to label them automatically [8]. Region based convolution neural network can be used to detect locations of objects in images. Deep learning models is very useful with their multilevel structures. It helps to classify complicated images. It is also very useful in terms of reduce computation time with high GPU performance. If we want to create a deep learning model for character recognition, first we need to define large amount of number images as input data. Deep learning model gets these images as pixels and locate them into neurol network[9]. The neural network is enlarged as the number of input data. According to image classification neural network will has several outputs. These outputs are defining according to characters. If we want to predict the image is “1” outputs are “It is one” and “It is not one”. There are separate outputs for each object, so model can classify objects into groups. All these images train neural network. In time, it starts to learn and classify by itself.

Input: Video frame (frame)

Output:

1. The first module is the input image provided.

2. Detects plate location.

3. The plates found are clipped according to the bounding boxes and are given as inputs to the second module.

4. The character network detects and estimates the characters in the image.

5. Estimated full plate according to the order of bounding boxes.

2.5 Optical Character Recognition

Optical character recognition (OCR) represents the procedure of data changes gathered from images and video records with deep learning techniques. These techniques comprise the combination of the state-of-the-art deep neural systems for both crucial tasks: plate detection, character segmentation and character recognition which are prepared and approved on artificially generated vehicle plates' dataset [10]. The different frameworks proposed in license plate (LP) detection depends on various properties. These properties utilize basic rules based on deterministic methods such as environment analyze, morphological operations and measurable analyzes while others settle on learning and grouping frameworks [11]. Context shape technique, contour detection in low light conditions and vehicle image color localization can be good example to these operations and analyzes. The second step in license plate recognition with deep learning methodologies is the character segmentation phase which organizes procedures and techniques have been performed for identify the region character in a license plate image [12]. Character segmentation based on two approaches which are reinforcement learning and hybrid learning. Reinforcement learning approach revealed by behaviorism, which arranges with actions to do for achieving the most efficient way in an environment. State, reward, policy, value and environment might be an element of a reinforcement learning. In the other hand, hybrid approach deals with speed and simplicity for segmenting the character. The last step in license plate recognition is the character recognition phase which is built on breaking down each character to constituent elements like curves and

corners. In this phase, the machine will look for matching physical features and actual letters with comparing the collected previous data.

2.5 Data Sets

The photos of 4000 vehicles obtained from the ISSD Inc. Photos are ready to be used within the project. Approximately 600MB of data will be used under this project. File name, date, vehicle brand model, plate location etc. When saving photos in the system, information is written to photos For example 16J7XX-TUR_XXXXXXXXXXXXXXX_2018-10-14_16-12-02-888_OTOMOBIL_OPEL_SIYAH_0_809-1363-1129-1440_14_98_2.jpg is written on the photos.

Figure 1. Example of a data

2.6 Conclusion

Image processing and image processing using deep learning are relevant subjects. Both based on processing input data, but they have different mythologies. Image processing is a traditional method, better for low processing power. Image processing techniques can be used for increasing the power of deep learning. Deep learning is a rising method. It has better, faster solutions for high processing power. It also has better prediction ration with big data sets.

3. Software Requirements Specification

3.1 Introduction

3.1.1 Purpose

Automatic License Plate Recognition (ALPR) used in various areas such as security, monitoring and gathering vehicle's data. This is a literature review that searches previous works about license plate recognition using image processing methods and license plate recognition using deep machine learning methods. This literature review aims to inform about differences between these methods and their efficiency. Our vision is to prove that deep machine learning methods is way more accurate and efficient compared to image processed license plate recognition. Therefore, our project name is "License Plate Recognition Using Deep Learning".

3.1.2 Scope of Project

The goal of this project is comparing license plate recognition with image processing and license plate recognition with deep learning. Proving that deep learning has higher accuracy rate than image processing. Image processing is the oldest form of character recognition. Deep learning has become a popular term recently. Deep learning can work with different angles, backgrounds. It can also automatically learns changing the lighting, flipping the image. It has neural networks, so it is better in detecting specific objects in complicated images, videos. Deep learning can be applied to many image processing problems like license plates recognition, hand writing recognition, human faces detection. License plate recognition has many application areas. It is most commonly used in traffic monitoring. There are cameras almost in every road, street, cross road… They are monitoring us 7/24. They can detect violation of traffic rules. For example, vehicles with high speed, red light violations, vehicles driving on bus line. They are also used in automated systems like KGS. They can even detect stolen cars. All systems use almost the same principle. When sensors capture a violation, cameras take picture of the vehicle. Find the location of the plate in the picture, the after necessary read the license plate numbers. Driver can be found with license plate number.

3.1.3 Glossary

Term	Definition
Stakeholders	Any person who has contribution in the project
s	Vehicles images and plate numbers
Data of Character	Character images
End User	Person who receive plate number
PR	Plate recognition
OCR	Optical Character Recognition
CNN	Convolutional Neural Networks
ALPR	Automatic License Plate Recognition
LP	License Plate

3.1.4 Overview of Document

The second part of the document describes functionalities of License Plate Recognition with Deep Learning project. Informal requirements are described, and it is a context for technical requirement specification in the Requirement Specification part. Requirement Specification part is written for software developers and details of the functionality of the project are described in technical terms.

3.2 Overall Description

3.2.1 Product Perspective

The purpose of the license plate recognition using deep learning project is to prove that deep learning methods are better than image processing methods. At the end of this project, we want to increase the success rate of currently used license plate recognition systems in Turkey. This project is a desktop application developed using python language.

3.2.2 Development Methodologies

For developing the project, we have planned to use waterfall methodology. Our aim on this project is to improve something already exist. We have a small group. We defined requirements before started to project. We do not develop this project interactively with users. We do not divide the project into individual sprints. We want to have certain resolution at the end of the project. Waterfall also eliminate the risk of failure toward that goal.

3.2.3 User Characteristic

We have two different kind of users. First one is user second one is API user. User can upload images, process images, browse and report. API user can upload images and process images.

3.3 Requirements Specification

3.3.1 External Interface Requirements

User Interfaces The user interface will be worked on Windows.

3.3.2 Hardware interfaces

The simulation requires a camera and computer hardware. Also, it requires HDMI and USB support.

3.3.3 Software interfaces

Image handler software might be needed.

3.3.4 Communications interfaces

There are no external communications interface requirements.

3.4 Non-Functional Requirements

3.4.1 Performance

Performance is one of the most important aspect for this system. The system should have a high performance such that the user can see the view of the area streaming with 15 fps. Works on blurry and dark images.

3.4.2 Accuracy

The system should have at least 95% accuracy.

3.4.3 Availability

System will be available while the computer is running on.

3.4.4 Security

The data gathered from data set should be only accessible for certain security forces and police.

3.5 Functional Requirements

3.5.1 User Use Case

Use Case:

• Upload image(s)

• Process image(s)

• Browse
• Report

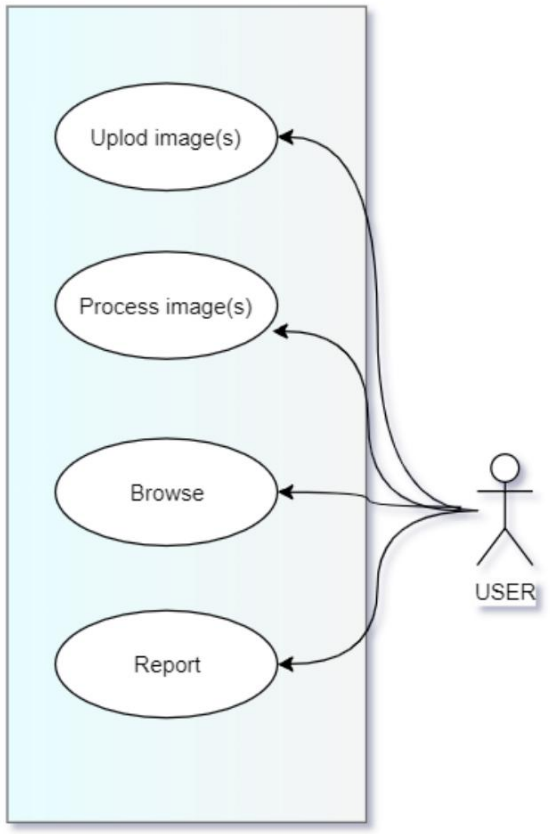


Figure 2. User Use Case

Figure 1. Use Case for User

1. User can upload image for license plate recognition.
2. User can process image and see the result of uploaded image.
3. User can browse for image.
4. User can see the report of the result.

3.5.2 API User Use Case

Use Case:
• Upload image(s)
• Process image(s)

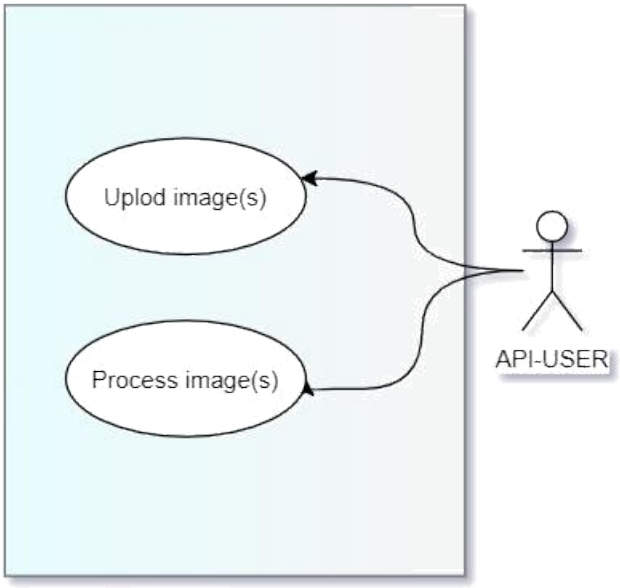


Figure 3. API-User Use Case

1.User can upload image for license plate recognition.

2.User can process image and see the result of uploaded image.

3.5.3 Admin Use Case

Use Case:

• Upload labeled data

• Train & Validate & Test model

• Manage users

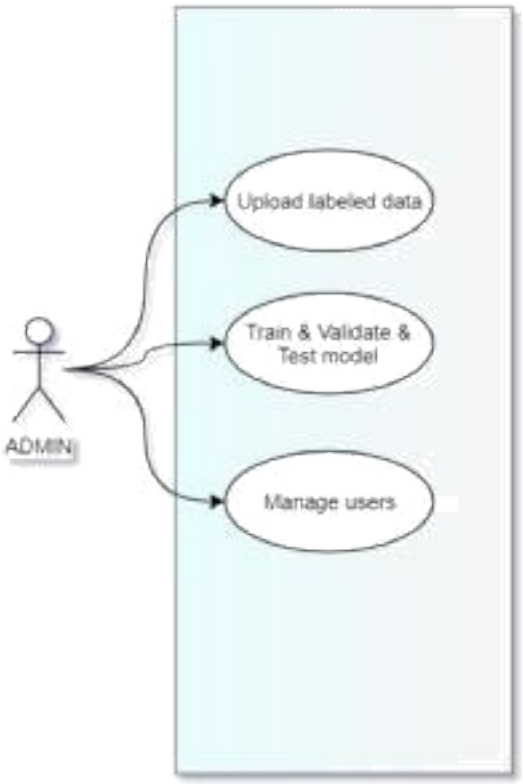


Figure 4. Admin Use Case

Admin can upload labeled data, images and license plate number information will be uploaded.

1.Admin can train, validate and test model functions,

2. New PR model will be trained and tested.

3.Admin can manage users.

4. Software Design Description

4.1 Introduction

4.1.1 Purpose

Deep learning methods are not widely used in license plate recognition systems in Turkey. Deep learning methods can work at different angles and on different backgrounds, and at the same time it can easily adapt to change in lighting automatically. Image-processing methods produce static solutions and difficult to improve over time. As a result, deep learning systems become better at performance as the data used over time increases. This is the success of deep learning methods. Artificial neural networks-based systems are more successful in identifying specific objects in mixed images and videos. Because of the lack of image processing methods according to deep learning methods in many ways, we aim to move to deep learning-based methods which are an innovative approach to improve the systems used today. At the same time, we aim to make comparisons to prove that the deep learning methods to be used in this project are more successful and performance than the image processing methods.

4.1.2 Scope

This license plate recognition system is based on the principle of operation in Turkey image processing algorithms executed on the processor broadcasts from the camera. These algorithms determine the position of the plate on the camera broadcast and determine the characters on it by character analysis and then combine them as a whole. When the plate position is found and after the character analysis on the image, the pixel image processing methods are used. While these images work at 15 fps during the video, the image processing methods require a lot of processing power and are also reduced due to frameworks that the accuracy rate misses. Therefore, using our computer science techniques in our project, it is aimed to change the methods used and to achieve lower cost and higher accuracy rate. Our aim in this project is to explain plate recognition with deep learning and to compare these methods with image processing and performance. Turkey also had the license plate recognition systems, has been working with image processing methods. The purpose of the plate identification system using deep learning methods is to perform character analysis with deep learning methods instead of image processing and character analysis and to provide

more efficiency in terms of power and performance. Deep learning methods can work at different angles and backgrounds with high efficiency and at the same time automatically adapt to changing lighting. It is more successful in identifying specific objects in mixed images and videos with the help of neural networks. In the prototypes and simulations, 95% accuracy rate is aimed in plate reading and character analysis with deep learning methods. In the current system, image processing algorithms are developed using the C ++ programming language and OpenCV library in the processor, and tests are carried out using MATLAB in the development phase. In this project, we aim to develop a deep learning model using the PYTHON programming language and to train a powerful gpu with the PYTHON language. We aim to convert the learned data formed after the training to the C + + programming language in order to be able to work in older systems.

4.1.3 Glossary

Term	Definition
Stakeholders	Any person who has contribution in the project
Data of Vehicles	Vehicles images and plate numbers
Data of Character	Character images
End User	Person who receive plate number
PR	Plate recognition
OCR	Optical Character Recognition
CNN	Convolutional neural network
ALPR	Automatic License Plate Recognition
LP	License Plate

4.1.4 Overview of document

The remaining chapters and their contents are listed below.

Section 2 is Architectural Design which describes the project development phase. Also it contains activity diagram of the project.

Section 3 is Use Case Realization. This section is designed according to use cases in SRS document.

Section 4 is Interface Design. In this section, we have shown the user interface design.

4.1.5 Motivation

Plate recognition systems are widely used in the world to produce common solutions in the field of traffic and security. These common systems bring complexity. Today's technologies are costing plate recognition systems. For example, when we want to install a plate recognition system in a city, the equipment power and equipment need to be used is increasing. This situation increases the cost greatly. At the same time, plate recognition systems, safety and traffic management in terms of accuracy must be high in terms of systems. Because each unreadable plate greatly increases the security risk and traffic complexity. This project started in one of the most important reasons for not being used in deep learning techniques available in Turkey and license plate recognition system is not created a data set consisting of Turkish plates.

4.2 Architecture Design

4.2.1 License Plate Recognition Using Deep Learning Design Approach

For developing the project, we have planned to use waterfall methodology. Our aim on this project is to improve something already exist. We have a small group. We defined requirements before started to project. We do not develop this project interactively with users. We do not divide the project into individual sprints. We want to have certain resolution at the end of the project. Waterfall also eliminate the risk of failure toward that goal.

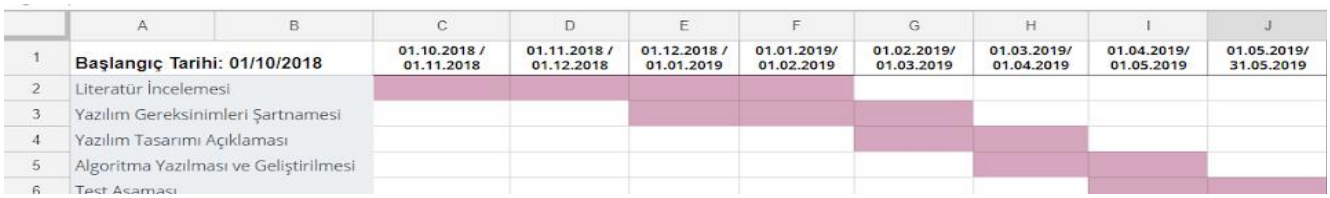


Figure 5. Time Table

4.2.2 Class Diagram

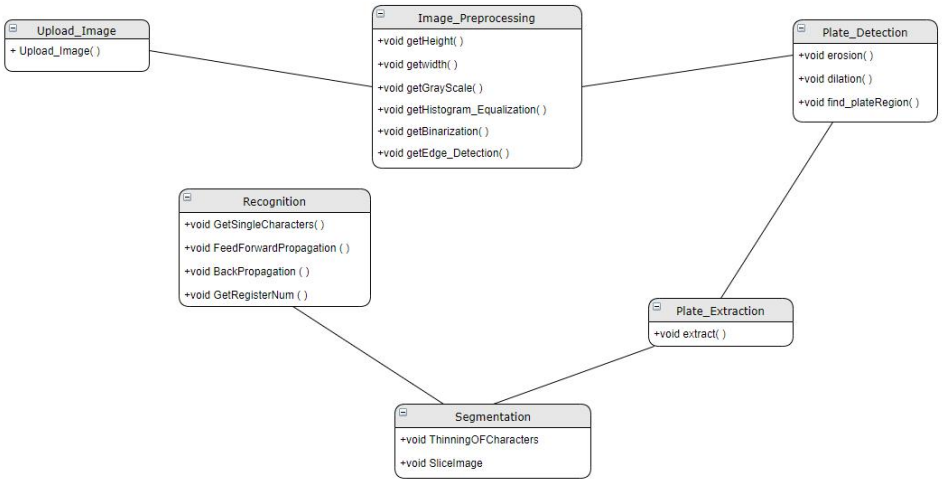


Figure 6. Class Diagram

This diagram shows the structure of the system; expresses the classes in the system. Anyone who uses the application must upload an image to process. After that, preprocessing will be applied to uploaded image. These preprocessing steps are getting height and width, gray scale, histogram equalization, binarization and edge detection. Then it gets the plate location with erosion, dilation and plate region. After that it extracts the plate from the image. Then segmentation will happen. Segmentation steps are thinning of characters and slicing the image. Finally, recognition happens. Steps are getting single characters, feed forward propagation, back propagation and getting register number.

4.2.3 Architecture Design

Upload Image

Summary: This system is used by end user. End user can upload any image according to his or her choice by clicking upload picture button.

Actor: End User

Precondition: End user must run the program.

Basic Sequence: User can open the program. User can click the “Upload Picture” button. User can choose any picture from his/her documents to upload.

Exception: None Post Conditions: None Priority: Low

Report

Summary: End user can compare the results of image processing and deep learning by clicking “Test” button.

Actor: End User Precondition: End user must upload a picture and must click to “Test” button.

Basic Sequence: End user can view the character analysis. End user can click “Test” button to view the results of image processing and dep learning.

End user can view character analysis based on OCR and CNN. End user can determine the best algorithm by comparing the results.

Exception: None.

Post Conditions: None. Priority: Medium.

4.2.4 Activity Diagram

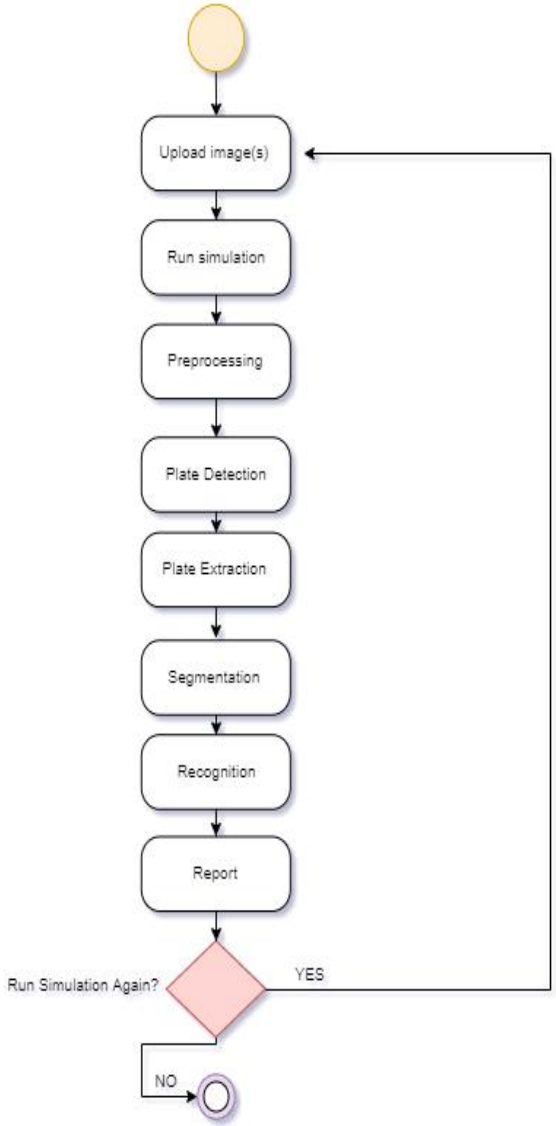


Figure 7. Activity Diagram

Figure 7 shows how the License Plate Recognition Using Deep Learning works as an activity diagram. User upload an image or images that him/her wants to perform character recognition. Then runs the simulation. Program starts to apply character recognition using deep learning algorithms. These algorithms are plate detection, plate extraction, segmentation and recognition. Finally, user see the report. This report is about compression between image processing and deep learning methods and plate information.

4.2.5 Flow Chart

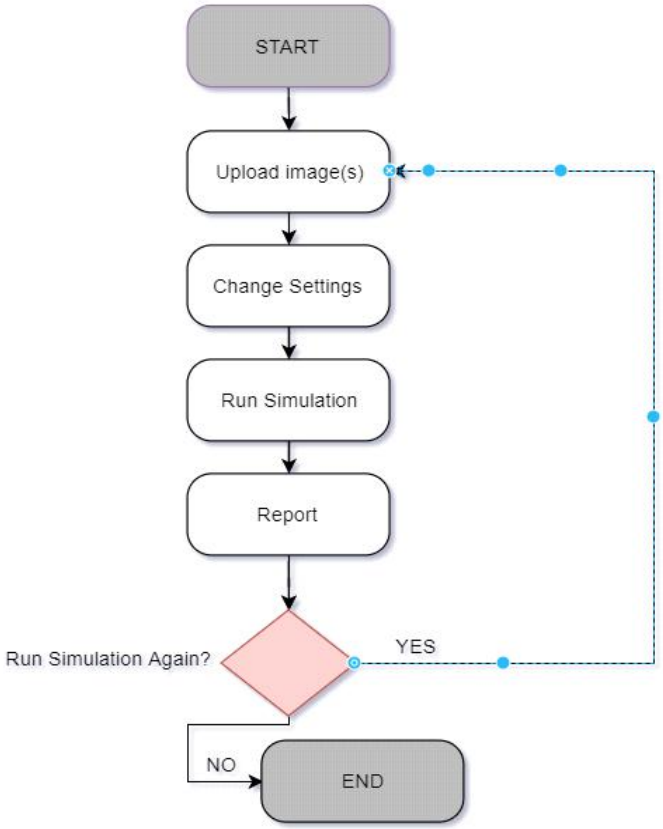


Figure 8. Flow Chart

Figure 8 shows how the License Plate Recognition Using Deep Learning works as a flow chart. First users or admin needs to upload an image or images. They can change the settings if they it is required. Then they run the simulation and see the result as report. They can run the simulation again.

4.3 Use Case Realization

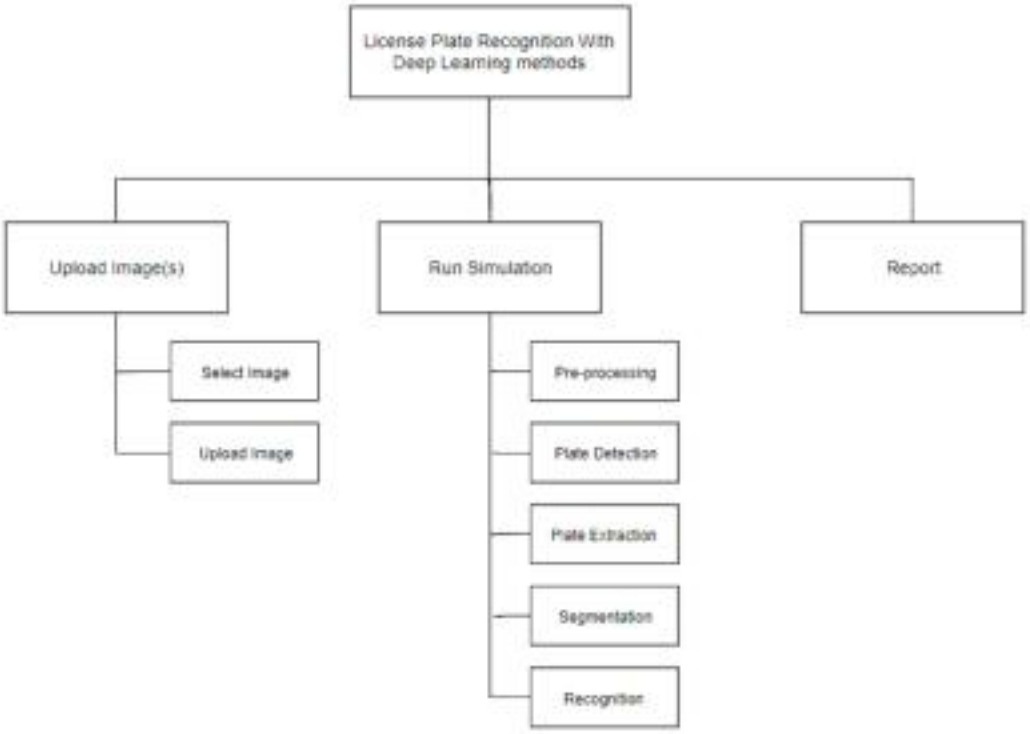


Figure 9. Functional Decomposition Diagram

Functions of License Plate Recognition Using Deep Learning are shown in figure 9. All designed systems of the project are displayed in the block diagram in the figure. There three four main components of the system which have their own sub-systems.

4.3.1 Upload Image(s)

Upload image(s) is for the user to upload an image in order to begin character recognition. It has two sub-systems which are select image and upload image. User can select then upload image of his/her choice.

4.3.2 Run Simulation

Run simulation is for application to start beginning character recognition. It has five sub-systems which are select pre-processing, plate detection, plate extraction, segmentation

and recognition. Character recognition happens in this order. Algorithm in Python script language will be integrated into system developed with C ++ language.

4.3.3 Report

User see the report. It is about compression between image processing and deep learning methods and plate information. The plate recognition system have at least 95% accuracy.

4.4 User Interface Design

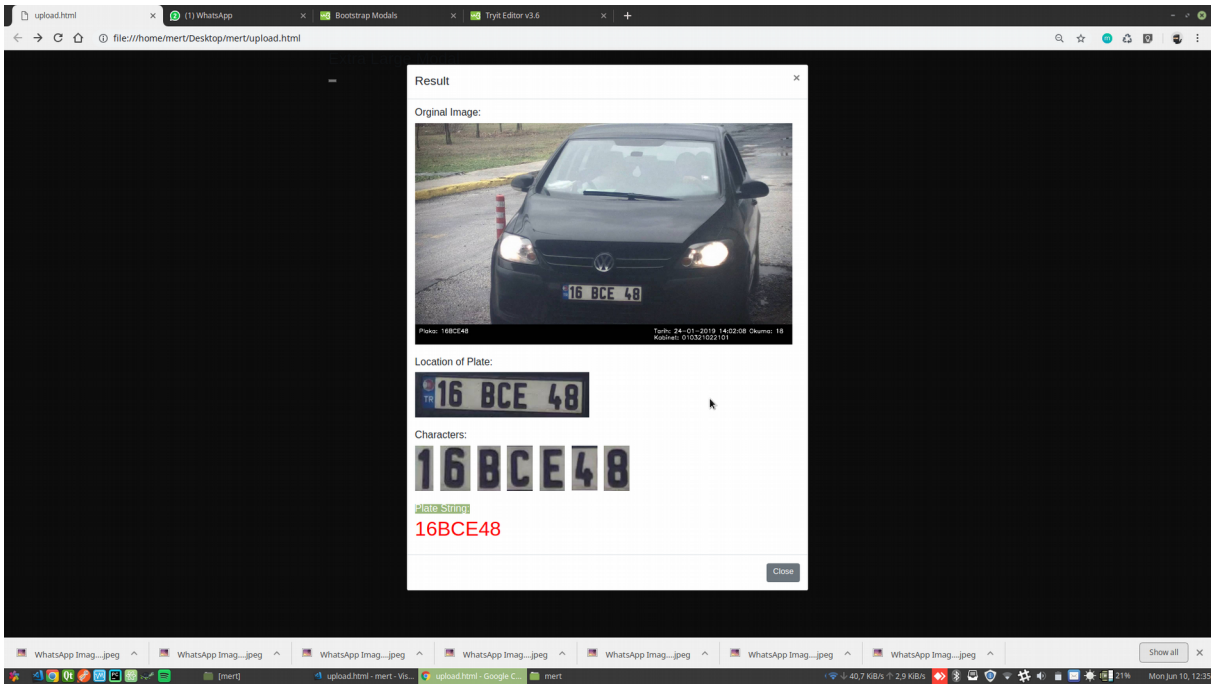


Figure 10. Example Interface

5. Test Plan
5.1 Introduction
5.1.1. Version Control

Version No	Description of Changes	Date
1.0	First Version	May 31, 2019

5.1.2 Overview

The use case of License Plate Recognition Using Deep Learning users namely participant and system administrator which has been determined in SRS document will be tested.

5.1.3. Scope

This document includes the test plan of use cases, test design specifications and test cases correspond to test plan. Scope of this project is plate recognition using deep learning. We have a dataset which includes 14,000 labeled plate images.

5.1.4. Terminology

Acronym	Definition
UWP	User Web Page
IU	Image Upload
PD	Plate Detection
Re	Report

5.2. Features to be tested

This section lists and gives a brief description of all the major features to be tested.

5.2.1 User Web Page (UWP)

In our project, User Web Page is the one and only page. When users open the project, they see he main page. In main page, there is one “Upload” button and free space for detailed process’ images. After users select the image, they can see the report of plate recognition of the image step by step. These steps include the system which works beyond the page with deep learning model.

5.2.1 Image Upload (IU)

User can upload image using two different ways. They can select an image using “Upload Image” button. They can also select and drag an image into page.

5.2.2Plate Detection (PD)

After user Upload the image, algorithm starts to run automatically beyond the system. Firstly, algorithms find the location of the plate. The plates found are clipped according to the bounding boxes and are given as inputs to the module that algorithm uses. Module provides the input image. Then system prints the cropped plate image. After that, the character network detects and estimates the characters in the image. Then system prints every character by one by. Finally, system uses deep learning algorithm and estimates full plate according to the order of bounding boxes.

5.2.3Report (Re)

User can see the result of plate recognition.

5.3. Item Pass/Fail Criteria

Describe under what conditions the testing of the product is considered successful. Some examples are:

- 100% of the test cases are executed
- 95% of the test cases passed
- All High and Medium Priority test cases passed

5.4. References

[1] LicensePlateRecognitionUsingDeepLearning_SRS,January 8,2019
[2] LicensePlateRecognitionUsingDeepLearning_SDD,January 8,2019

5.5. Test Design Specifications

5.5.1 User Web Page (UWP)

5.5.1.1. Test Cases

Here list all the related test cases for this feature

TC ID	Requirements	Priority	Scenario Description
UWP.01	3.5.1	H	User page uploaded correctly.

5.5.2 Image Upload (IU)

5.5.2.2. Test Cases

Here list all the related test cases for this feature

TC ID	Requirements	Priority	Scenario Description
IU.01	3.5.1	L	User uploads a image.

5.5.3 Plate Detection (PD)

5.5.3.3. Test Cases

Here list all the related test cases for this feature

TC ID	Requirements	Priority	Scenario Description
PD.01	3.5.3	H	Algorithms find the location of the plate. Module provides the input image. Then system prints the cropped plate image. The character network detects and estimates the characters in the image. System prints every by one by character. System uses deep learning algorithm and estimates full plate according to the order of bounding boxes.

5.5.4 Report (Re)

5.5.4.4. Test Cases

Here list all the related test cases for this feature

TC ID	Requirements	Priority	Scenario Description
Re.01	3.5.3	M	Result of plate recognition printed correctly.

5.6. Detailed Test Cases

5.6.1. User Web Page (UWP)

TC _ID	UWP.01
Purpose	Uploading main page.
Requirements	3.5.1
Priority	High
Estimated Time Needed	5 second
Dependency	Page needed to open for further operations.
Setup	Open the web page.
Procedure	[A01] Open the application. [A02] Check if the page is correctly uploaded.
Cleanup	

5.6.1. Image Upload (IU)

TC _ID	IU.01
Purpose	Uploading selected image.
Requirements	3.5.1
Priority	Low
Estimated Time Needed	2 second
Dependency	User should open the program.
Setup	User should click the “Upload Image” button or select and drag an image.
Procedure	[A01] Click the “Upload Image” button or select and drag and image. [A02] Observe that the image is uploaded.
Cleanup	Click “Upload Image” button.

5.6.1. Plate Detection (IU)

TC _ID	PD.01
Purpose	Detecting plate location from image than character recognition.
Requirements	3.5.2
Priority	High
Estimated Time Needed	1 minutes
Dependency	User should upload an image to run the program.
Setup	After user select an image, algorithm starts to run automatically.

Procedure	[A01] Algorithm runs automatically. [A02] Algorithms find the location of the plate. [A03] Cropped image given as inputs to the module that algorithm uses. [A04] Module provides the input image [A05] Cropped plate image is printed by the system. [A06] the character network detects and estimates the characters in the image [A07] Then system prints every character by one by [A08] system uses deep learning algorithm and estimates full plate according to the order of bounding boxes.
Cleanup	

5.6.1. Report (Re)

TC _ID	Re.01
Purpose	Final plate number is printed on the screen.
Requirements	3.5.3
Priority	Medium
Estimated Time Needed	5 second
Dependency	Plate detection test case should pass.
Setup	System automatically print plate number after plate detection case.
Procedure	[A01] Observe that final plate number is correct.
Cleanup	

5.7. Test Results

We select 5 random images for showing accuracy rates and listed blow in the table.
The test cases are implemented and 50% of the test cases are completed successfully.

Image number	Input	Output	Correct/False
1	06 TCE 28	06 TCE 28	Correct
2	58 OD 217	58 OD 217	False
3	37 MR 398	37 MR 398	Correct
4	02 GRJ 03	02 DRJ D3	False
5	06 FTE 98	06 FTE 98	Correct

6. Conclusions

Developing a plate recognition system using image processing and deep learning is relevant to each other. Both relate to processing input data, but they have different mythologies. Image processing is a conventional method that is better for high processing power. Image processing techniques can be used to increase the power of deep learning methods. Deep learning is the most recent approach to its use in the world. It has better and faster solutions for high processing power. It also has the ability to better estimate because it uses large data sets. The biggest expectation in our project is that the system performance is sufficient for the user.

Acknowledgement

We are grateful for guidance we have received from Prof. Dr. Erdoğan Doğdu and Doç. Dr. Reza Hassanpour. We received the biggest support and courage from our advisers. We are also grateful to help we received from ISSD Inc. They let us to use company's data and helped us during the project.

Appendix A: Installation Guide

Requirements

- Neural Computer Stick, Intel Movidius (optional)
- Any CPU and GPU

Building the Application

- Download Anaconda Navigator.
- Download and install python3 from website.
- Download and install pip from website.
- Open the terminal of your computer.
- Enter “pip install keras” into terminal
- Enter “pip install tensorflow” into terminal
- Enter “Python3 app.py” into terminal
- Download or copy repository from GitHub.
- Extract the code into Anaconda Navigator.
- Run the code on Anaconda Navigator using Jupyter.

Appendix B: User Manuel

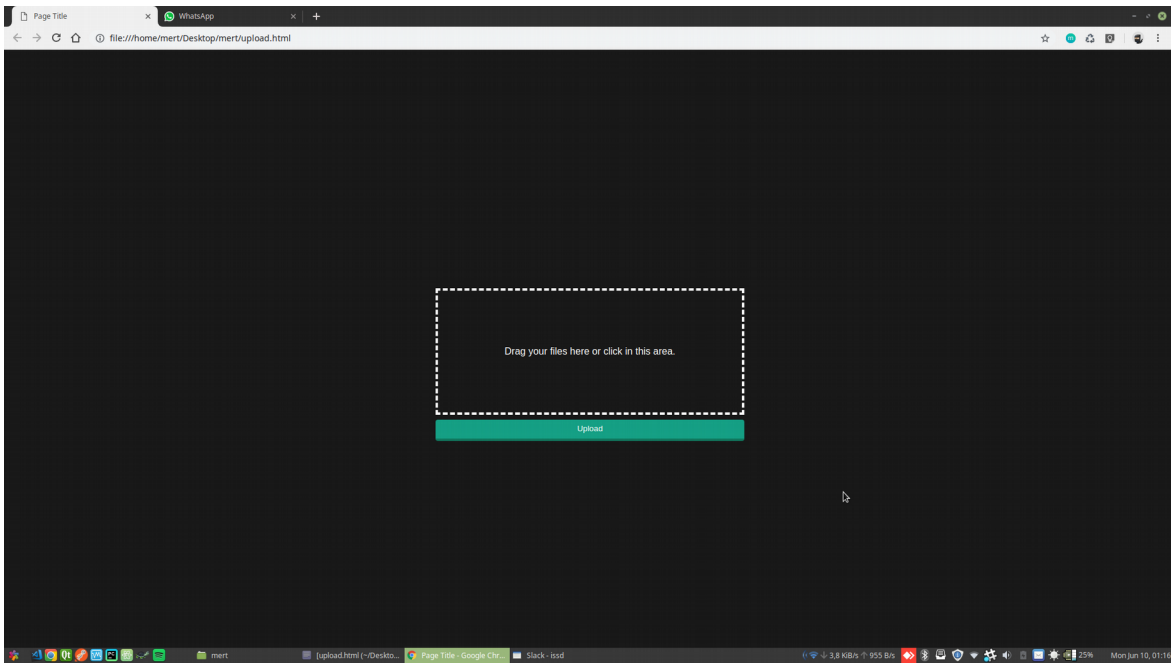
Requirements

Overview

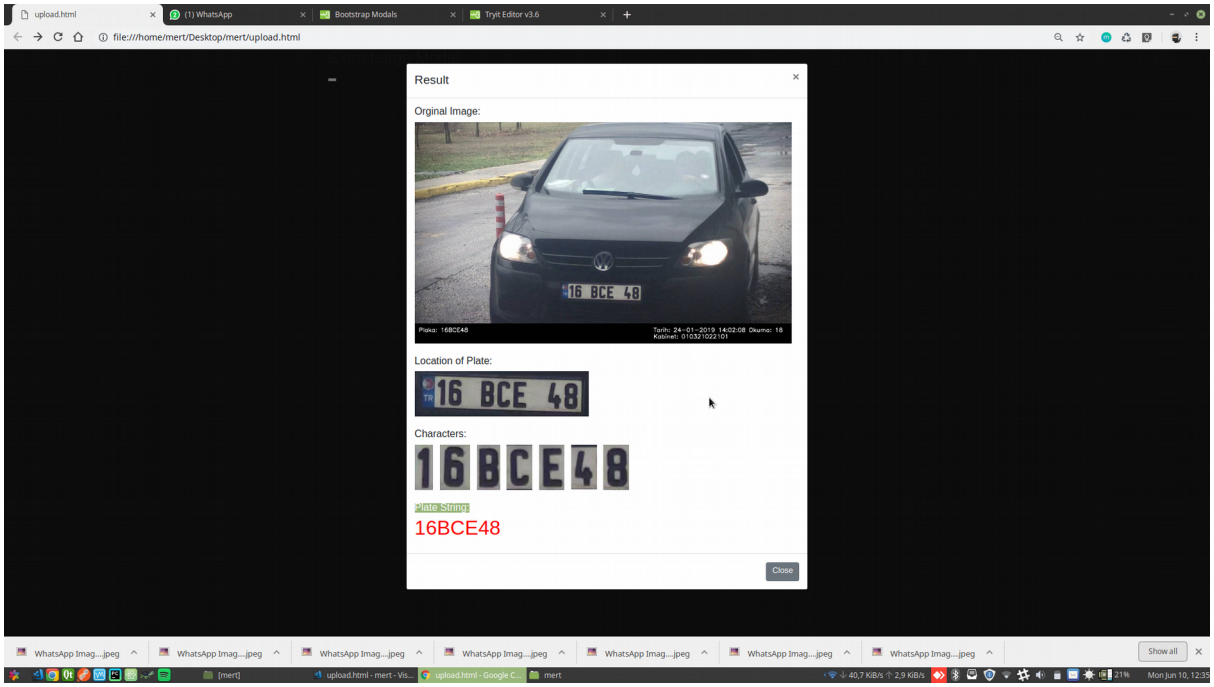
License Plate Recognition System is a web site for user who want to do character recognition using deep learning. User can upload an image of a vehicle where license plate is visible enough for the system to detect. System uses algorithms and models. After that prints the plate number on image using deep learning algorithms.

User Interface

Firstly, all users enter the system through the main page. User can upload image using two different ways. They can select an image using “Upload Image” button. They can also select and drag an image into page.



After user upload the image, algorithm starts to run automatically beyond the system.



Firstly, algorithms print the cropped plate image. Then system prints every character by one by. User can see the result of plate recognition.

References

1.Ben-Ari, M., Mondada, F. (2018). Image Processing. Elements of Robotics.

2. Alata, M., Al-Shabi, M. (2006). TEXT DETECTION AND CHARACTERRECOGNITION USING FUZZY IMAGE PROCESSIN. Journal of Electrical Engineering, 57(5), 258–267.

3. Kumar, T., & Verma, K. (2010). A Theory Based on Conversion of RGB image to Gray image. International Journal of Computer Applications, 7(2), 7-10.

4. Lee, J. S. (1980). Digital image enhancement and noise filtering by use of local statistics. IEEE Transactions on Pattern Analysis & Machine Intelligence, (2), 165-168.

5. Shen, J., & Castan, S. (1992). An optimal linear operator for step edge detection. CVGIP: Graphical Models and Image Processing, 54(2), 112-133.

6. Wang, L. (Ed.). (2005). Support vector machines: theory and applications (Vol. 177). Springer Science & Business Media.

7. LeCun, Y., Bengio, Y., & Hinton G. (2015). Deep learning. Nature International Journal of Science. 521(7553), 436-444.

8. Sermanet, P., Chintala, S., & LeCun, Y. (2012). Convolutional Neural Networks Applied to House Numbers Digit Classification. arXiv:1204.3968.

9. Mnih, V., Kavukcuoglu, K., Silver, D., Graves, A., Antonoglou, I., Wierstra, D., & Riedmiller, M. (2013). Playing Atari with Deep Reinforcement Learning. arXiv:1312.5602.

10. Arsenovic, M., Sladojevic, S., Anderla, A., Stefanovic, D. (2017). Deep Learning Driven Plates Recognition System. XVII International Scientific Conference on Industrial Systems (IS'17).

11. Selmi, Z., Halima. M. B., Alimi, A. M., (2017). Deep Learning System for Automatic License Plate Detection and Recognition. 14th IAPR International Conference on Document Analysis and

12. Puarungroj, W., Boonsirisumpun, N. (2018). Thai License Plate Recognition Based on Deep Learning. Procedia Computer Science, 135, 214-221.