College OF Computer Engineering and sciences
Computer Science Department

Parking Reservation System Based on image processing

Student: Anas Ahmed AL Wohaib (ID: 441050184)
Student: Saleh Abdullah Bin Shaheen (ID: 441050865)
Student: Nasser Fahad AL Dawood (ID: 441051871)

Project Supervisor: Dr. Bellhassen Akrout

June 4, 2023

Acknowledgement

First, we would like to thank all the faculty members for what they did for us during the first graduation project, and we especially thank our supervisor, Dr. Belhassen Akrout, as the project will not be complete without his advice, suggestions and guidance, and we wish you and us all the best.

Abstract

Parking Reservation System Based on image processing

We usually suffer from finding a parking space at peak times, as well as wasting time searching for a parking lot in public places, the large population in cities and the increase in the number of cars increases with it the problem of the lack of car parks, which are usually crowded at peak times, and no one can guarantee the car park. In this paper, we want to solve a problem that we face daily in the car parks areas when you go, so we decided to find a way to ensure that the car is parked without wasting time searching for it as usual, and we have an idea to solve the problem which is a website to reserve parking spaces cars before heading to the public place. This is done through a method that determines the parking area, installs a camera at the entrance gate, and then displays the number of parking spaces available to the customer on the site. Then, the customer books through the site and enters his information. The license plate will be checked to see whether it matches the same reservation or not. After completing the project idea on the ground, it is expected that the crowding of car parks in public places and workplaces will decrease. The process of searching for a parking lot will be smooth, easy, and faster. Also, what results from the implementation of the idea is to reduce emissions from cars, which positively affects the environment. This process will result in saving time and reducing congestion in parking lots. our aim is "Your car park is secured for you"

Abstract (Arabic)

نظام حجز مواقف السيارات باستخدام معالجة الصور

كثرة عدد السكان في المدن وزيادة عدد السيارات يزيد معها مشكلة عدم وجود مواقف للسيارات والتي عادة ما تكون مزدحمة في أوقات الذروة، ولا يمكن لأحد أن يضمن موقف لسيارته، ولذلك أصبحت معاناة لدى السائقين في إيجاد مكان لوقوف سياراتهم خصوصاً في أوقات الذروةس، وكذلك تضييع الوقت في البحث عن موقف لسيارته في الأماكن العامة. في هذا المشروع، نريد ان نحل مشكلة نواجهها يوميًا في مواقف السيارات، لذلك قررنا إيجاد طريقة لضمان إيقاف السيارة دون إضاعة الوقت في البحث عن موقف كالمعتاد ، ولدينا فكرة لحل هذه المشكلة وهي ان نصمم موقع الكتروني لحجز مواقف السيارات قبل التوجه للمكان العام، ويتم ذلك من خلال طريقة لتحديد منطقة وقوف السيارات اللتي تريد الوقوف بها، وتثبيت كاميرا عند بوابة الدخول، ثم تعرض عدد أماكن وقوف السيارات المتاحة للعميل في الموقع، ثم يقوم العميل بالحجز من خلال الموقع ويدخل معلوماته، وعند الوصول إلى موقع المواقف سيتم فحص لوحة السيارة لمعرفة ما إذا كانت تتطابق مع نفس معلومات الحجز أم لا. بعد الانتهاء من فكرة المشروع على أرض الواقع، من المتوقع أن ينخفض الازدحام في مواقف السيارات في الأماكن العامة وأماكن العمل، ايضاً ستكون عملية البحث عن موقف للسيارة سلسة وسهلة وسريعة، كما أن ما ينتج عن تطبيق الفكرة هو تقليل الانبعاثات من السيارات مما يؤثر بشكل إيجابي على البيئة، ستؤدي هذه العملية إلى توفير الوقت وتقليل الازدحام في ساحات الانتظار. هدفنا آموقف سيارتك مؤمن لكْ.

Contents

Acknowledgement **i**

Abstract **ii**

Contents **iv**

List of Figures **vii**

List of Tables **ix**

General Introduction **1**

Context 1

Problem Statement and Objectives 2

Report Structure 2

1 Background and related work **4**

1.1 Introduction 5

1.2 Literature Review 5

1.3 Discussion 14

1.4 Conclusion 17

2 Methodology **18**

2.1 Introduction 19

2.2 General description of the proposed approach 19

2.3 System Approach 20

2.3.1 Pre-processing 20
2.3.2 OCR 22
2.3.2.1 OCR approach 22
2.3.3 CNN approach 23
2.4 Proposed transferring approach 24
2.4.1 GoogleNet 25
2.4.2 LeNet 26
2.4.3 ResNet 27
2.5 Conclusion 27

3 Requirement Structuring **28**
3.1 Introduction 29
3.2 Use-Case Diagram 29
3.3 Sequence Diagram 30
3.4 Class Diagram 31
3.5 Conclusion 31

4 Experimental Methods **32**
4.1 Introduction 33
4.2 Implementation Environment 33
4.2.1 Python 33
4.2.2 Dataset 34
4.3 Implementation 34
4.3.1 Image processing 34
4.3.2 OCR 36
4.3.3 CNN 37
4.3.4 Source code 39
4.4 Accuracy result 39
4.5 Discussion 41
4.6 Conclusion 41

5 General Conclusion **42**

5.1 Conclusion and future work 43

Bibliography **44**

List of Figures

1.1 Number Plate Recognition using CCTV Camera 6
1.2 Intelligent car parking system management. 6
1.3 System Module 7
1.4 How an Ultrasonic Sensor Works 7
1.5 Detect parking slot 8
1.6 Image segmentation process 8
1.7 Parking System App, (a) Real Time Parking Status, (b) Chat Room Layout 9
1.8 (a) shows the input image used, (b) shows the output obtained 10
1.9 (a) Preprocessing stages, (b) Block diagram of the system 10
1.10 Car sizes that depend on sensors 11
1.11 WSN Diagram 11

2.1 System Approach 19
2.2 Character segmentation 21
2.3 CNN approach 23
2.4 Single InceptionNet Module 25
2.5 LeNet architecture 26
2.6 ResNet architecture 27

3.1 Use-Case Diagram 29
3.2 Sequence Diagram 30
3.3 Class Diagram 31

4.1 Sample of datasets 34
4.2 original image , gray image 34
4.3 BilateralFilter, Canny Edge detection 35
4.4 Finding best Contour 35
4.5 Characters after segmentation 36
4.6 OCR Result 36
4.7 CNN architecture LeNet style 37
4.8 Our model training 38
4.9 CNN Result 38

List of Tables

1.1 Recognition Result Comparison 12
1.2 Recognition Result Comparison 13
1.3 Total experimental results 13
1.4 Results 13

4.1 Methods and Accuracy results 40
4.2 Research comparison 41

General Introduction

Context

More convenient than on-street parking, as it is easier to find a spot. It is also usually cheaper than on-street parking. However, off-street parking can be limited in some areas, and it can be expensive in some cities. Parking can also be a source of pollution. Cars that are left running while parked can emit pollutants into the air, and cars that are parked in the sun can cause heat islands. This can be a problem in cities, as it can contribute to air pollution and global warming.

Parking can also be a source of congestion. When cars are parked on the street, they can take up space that could be used for other parking is an important part of our daily lives. It is a necessary part of getting around in our cities and towns, and it is essential for businesses to provide parking for their customers. Unfortunately, parking can be a source of frustration for many people. Finding a parking spot can be difficult, and it can be expensive. The most common type of parking is on-street parking. This is when cars are parked on the side of the street, usually in designated spots. On-street parking can be convenient, but it can also be difficult to find a spot. It can also be expensive, as many cities charge for on-street parking.

Off-street parking is another option. This is when cars are parked in a lot or garage. Off-street parking is usual purposes, such as bike lanes or bus lanes. This can lead to traffic congestion, as there is less room for cars to move around. Overall, parking is an important part of our lives, but it can also be a source of frustration and pollution. It is important to find ways to make parking more efficient and less of a burden on our

cities and towns. This could include providing more off-street parking, charging for on-street parking, and encouraging people to use public transportation.

Problem Statement and Objectives

Parking problems are a common issue in many cities and towns across the country. From overcrowded parking lots to limited parking spaces, it can be difficult to find a spot to park your car. This can be especially frustrating when you're running late for an appointment or trying to get to work on time. The most common parking problems are overcrowding and limited parking spaces. In many cities, the number of cars on the road has increased significantly over the years, but the number of parking spaces has not kept up. This leads to overcrowded parking lots and limited parking spaces, making it difficult to find a spot to park. Overall, parking problems can be a major issue in many cities and towns. To address these problems, cities should implement parking management systems and enforce parking regulations. This can help to ensure that everyone has access to a parking spot and that parking regulations are followed.

We have thought about a solution to this problem and there are some objectives to solve the problem: First, identify and analyze the parking area, Secondly, detect whether the parking lot is empty or not, Third, count the number of empty parking spaces and save them in the database. Fourth, designing a website for booking and linking the database to the website, Fifth, matching the reservation information to the car plate using image processing when the car arrives at the parking lot.

Report Structure

The rest of this report is organized as follows:

— **In Chapter 1**, we will review the previous study and discuss the solution and techniques that used for the car parking problem.

— **In Chapter 2**, we present our Methodology for solving The problem by using image

processing and deep learning (CNN).

— In Chapter 3 we will review the diagram of our system

— In Chapter 4 we will review the implementation of the system and accuracy result With a comparison of several other studies

Finally, we conclude our research with the conclusion

1 Background and related work

1.1 Introduction

In this section, we will review the existing solutions in the literature review section, and will provide an overview of previous research That focused on solving parking problems by using image processing and other technique, After that, we will discuss These Solutions in the Discussion section, We will end this chapter with a conclusion.

1.2 Literature Review

We will present some of the previous researches outcomes that provide solutions to the parking problem using various technologies. One of the solutions is to use IoT technology to solve the parking problem as this research [1], The internet of things "IoT" technology has various applications to work with, such as computer science, medical things, etc, This Research uses "IoT" with a parking availability system each parking space has an "IoT" gadget that includes sensors and an embedded system, The user is given a real-time update on the availability of all parking spaces, This study aims to make the parking area fully automated and image analysis, The customers can book parking by their license plate and an OTP is created for the user and stored in a database, and if the OTP doesn't correspond an alarm is triggered, image processing is used to transform the plate number into words that were entered into the database with IN time, after that the OUT time is computed and associated bill amount is created and payment is made through a payment machine at the gate, as seen in Fig1.1.

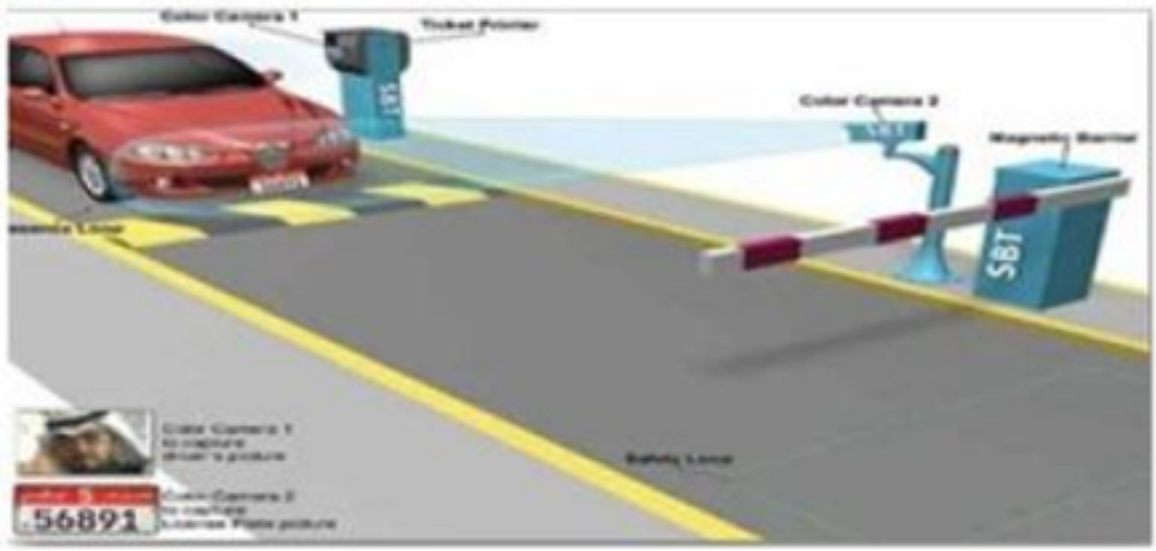


Figure 1.1: Number Plate Recognition using CCTV Camera

Ibrahim et al. [2], took pictures of the parking lots to count the number of parking lots and know the structure of the parking lots, then saved them in the system. After this is done, the process of numbering each car park in the parking lots takes place. After the user arrives, a ticket is printed after making sure that the parking lots are not full as is shown in Fig.1.2, then when he arrives at the parking lot, the camera detects the presence of a car in the parking lot, and then sends notifications to the system that the parking lot has become occupied Fig 1.2.

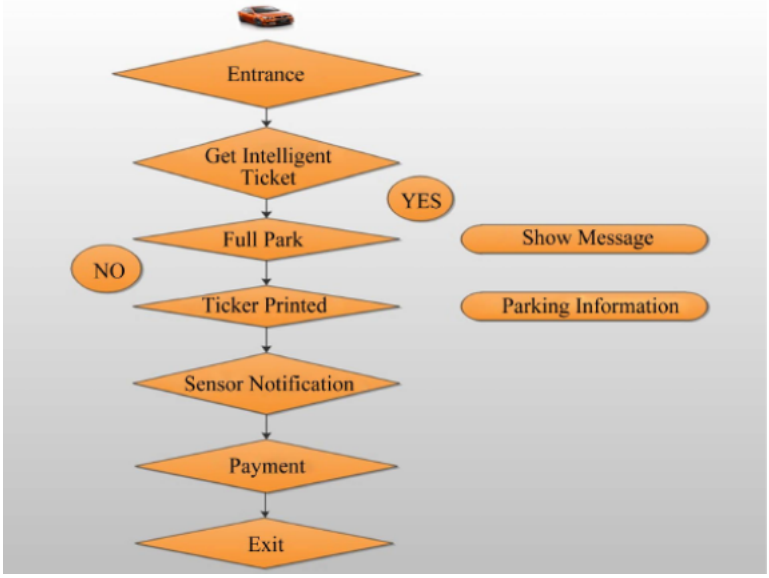


Figure 1.2: Intelligent car parking system management.

Fariza et al. [3], made an alignment drawing of what the empty parking lots would look like, and then collected more than one image of the parking lot under different conditions. After collecting the photos, they converted them from color photos to black and white photos to analyze the empty and occupied car park in real Time. This is done by using five modules, as shown in Fig 1.3

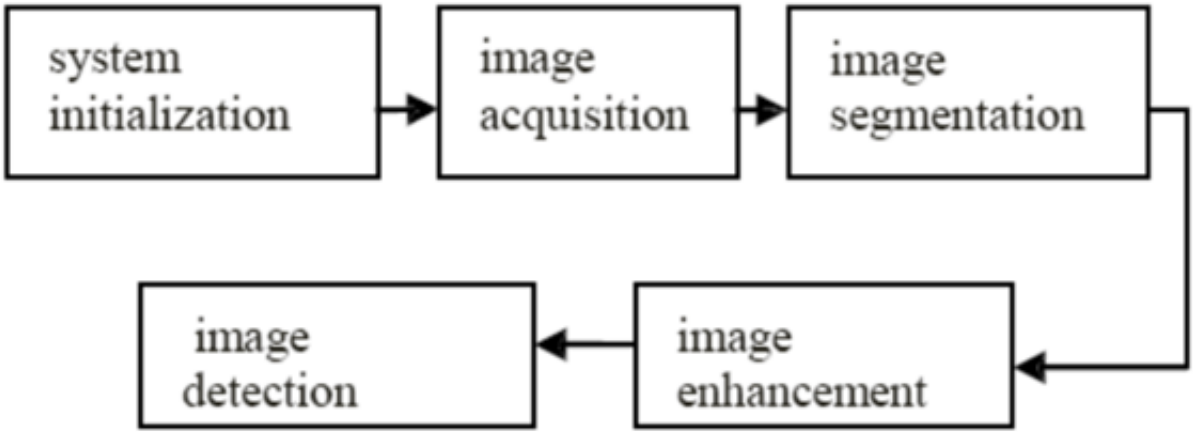


Figure 1.3: System Module

Amin et al. [4], used a unique method to detect whether the parking lot is occupied or not, which is the ultrasonic sensors, which release varying frequencies every 60 milliseconds, and to verify the parking lot they depend on the reflected energy. as shown in Fig 1.4 shows how it works

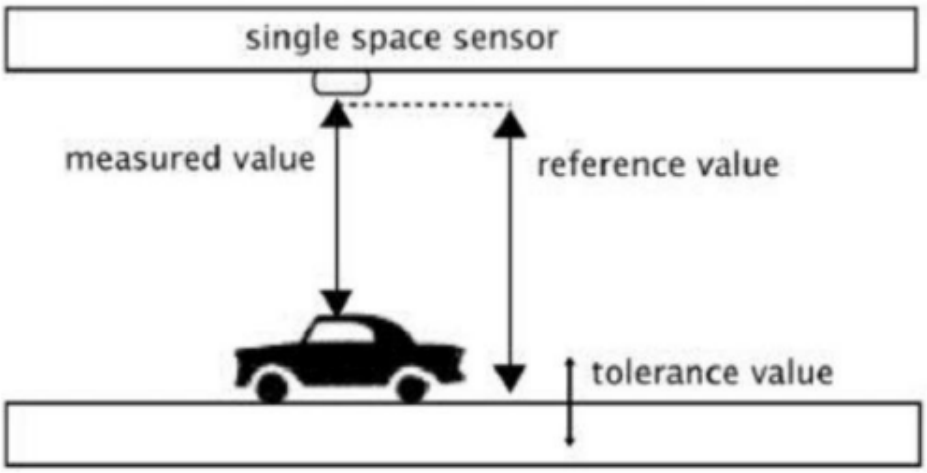


Figure 1.4: How an Ultrasonic Sensor Works

Ashu et al. [5], Focus to develop a fully automated parking system, they used vehicle detection and SIFT Detection to recognize cars that park that apply by the use of image processing technique, Fig 1.5, image segmentation used analyze the image and convert it to a binary image, as described in Fig 1.6.

Figure 1.5: Detect parking slot

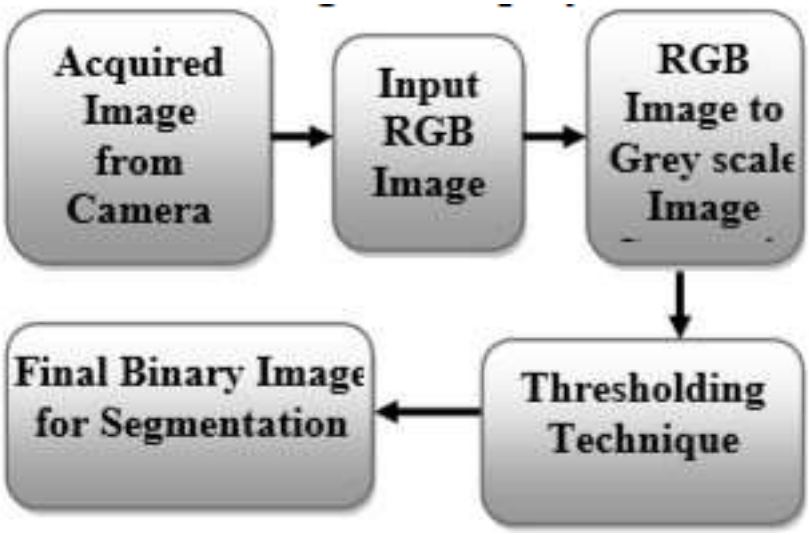
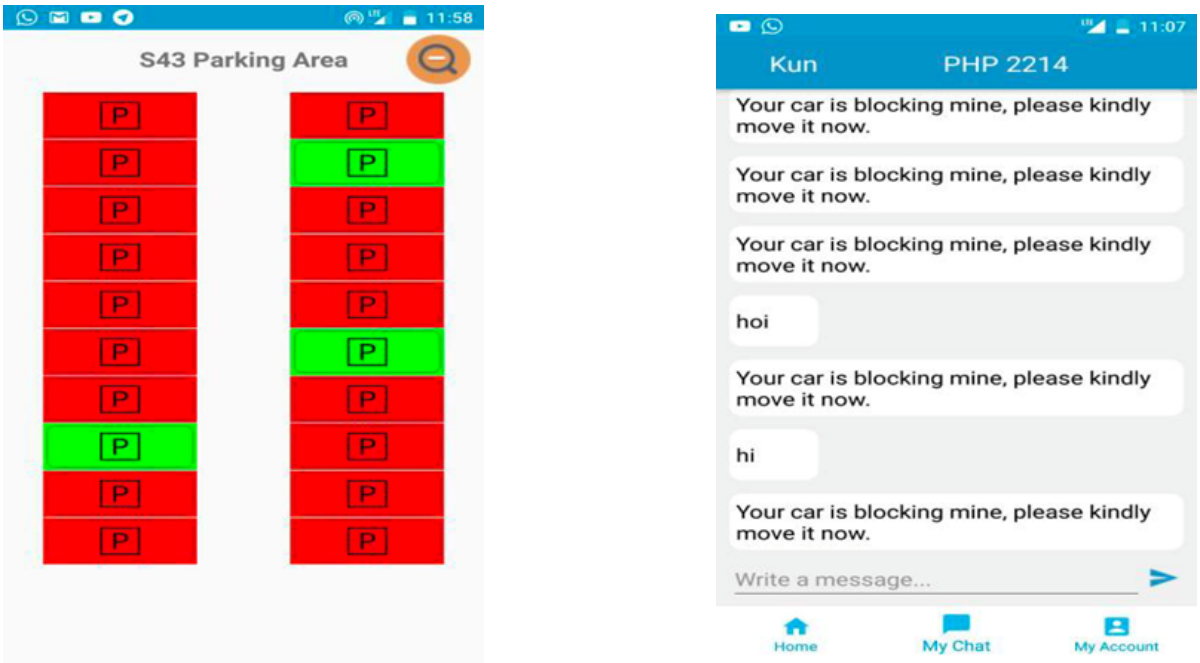


Figure 1.6: Image segmentation process

Yusmeeraz et al. [6], designed a system that discuss the problem of parking System in a new way by providing an intelligent solution, Offers a cost-effective IoT smart

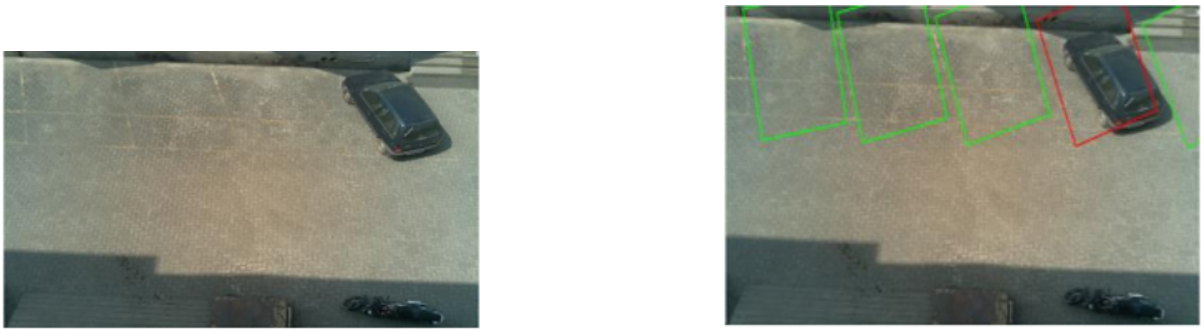
parking system to monitor city parking space and provide real-time parking information to drivers, and Allow user to communicate with each other by using an android application, also camera image and machine vision technology are used to obtain the parking status, They use two ways to obtain The Parking statute:1-Canny edge detection which help to recognize the edges of the image for instance to apply canny edge detection we use two Threshold Value one for minimum and one for maximum threshold values Only the edges with an intensity gradient than the minimum threshold value and less than the maximum threshold value will be displayed, 2-Blob detection technique that refers to modules that are designed to find spots and/or regions in the picture that have different brightness or color characteristics from the background, Look at **??**.



(a) (b)

Figure 1.7: Parking System App, (a) Real Time Parking Status, (b) Chat Room Layout

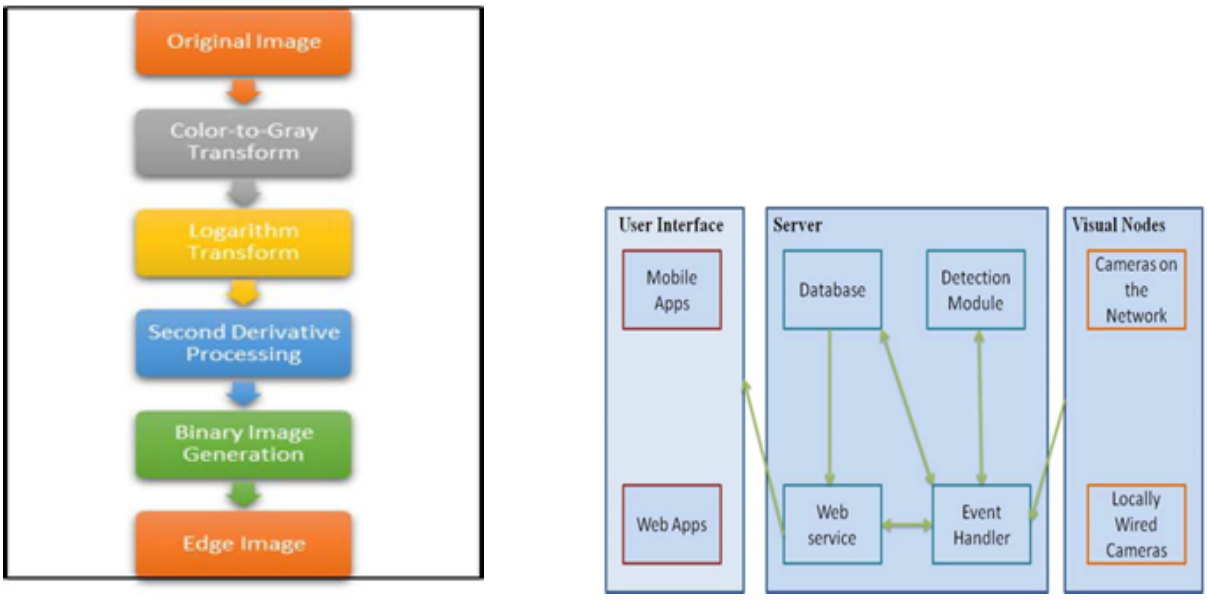
Chetan et al. [7], Put two algorithms in this research, the first is to store the images and extract them from the cloud to the raspberry pi, and then re-scale them to a suitable size so that all the images are processed easier. The second algorithm takes the output from the first algorithm and then detects the empty parking lot as a green square and the occupied parking lot as a red square. Look at Fig1.8



(a) (b)

Figure 1.8: (a) shows the input image used, (b) shows the output obtained

Tom Thomas et al.[8], solve The parking problem by Using Deep Learning Algorithm The algorithm he uses a Convolutional Neural Network (CNN) which is capable of finding if the parking lot is occupied or not by using a Camera (Computer Vision) to capture visual information to determine if a parking spot is vacant or not, that's done through several steps in preprocessing the images and CNN layers, Also this System is Divided into Three parts: 1- User Interface, 2- Server, 3- Visual Node, as seen in Fig1.9.



(a) (b)

Figure 1.9: (a) Preprocessing stages, (b) Block diagram of the system

Jatuporn et al. [9], applied their optical WSN to count the cars in a smart parking garage, they install a sensor in all parking which is connected to a server then dis-

plays the number of available parking, Fig1.10, they install two sensor heads, the two sensors are set 2.6 - 3.5 meters apart from each other, that calculate the size of the car, Fig1.11

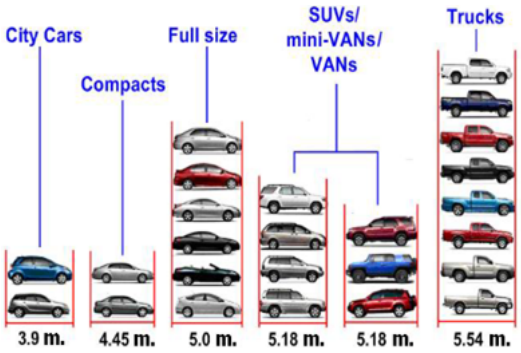


Figure 1.10: Car sizes that depend on sensors

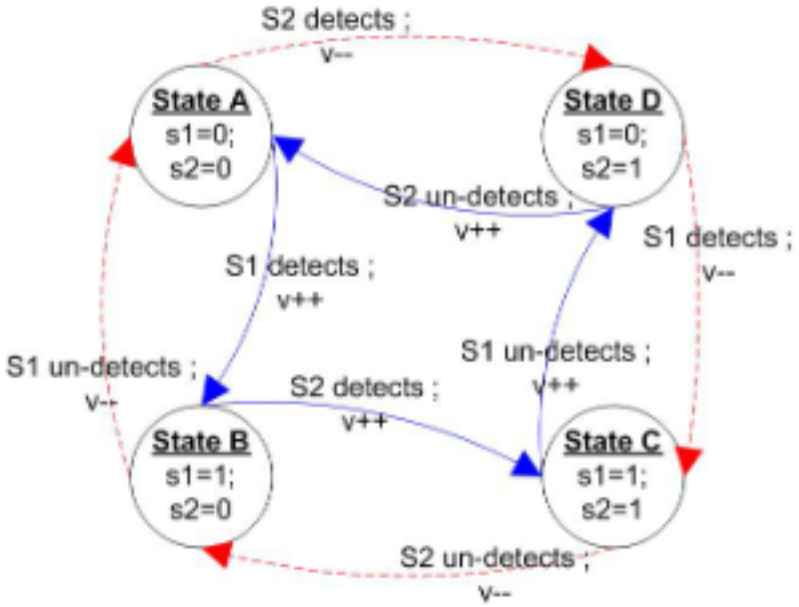


Figure 1.11: WSN Diagram

Hongwei et al. [10], show three components in their system, which are parking lots, users, and management system, The management system calculates the price which is based on real-time parking information also, and the user chooses the desired parking lot and reserves it, Eq 1. The management system updates the state of the

parking lot depending on its status, he made a formulation for the dynamic pricing that depends on usage price (Pu) " Calculate the real-time state, " statistical price (Ps) " Measured by the historical price ", Congestion price (Pc) " reflects the congestion price ", Eq 1, Also the Equation to calculate the profit, Eq 2, the system depends on Bluetooth without connection with the network, and the system will disconnect if the driver is inactive.

$$p = \beta_u p_u + \beta_s p_s + \beta_c p_c \tag{1}$$

$$\max_{p_u^i} \left[\sum_{i}^{I} (x^i p_u^i - f(j^i, J)) \right] \tag{2}$$

Rayana Antar et al. [11],In the field of their research, they relied on image processing to extract the car plate and identify its place in the image. After extracting the plate, they divided it into two parts, an English section and an Arabic section, and applied filters from image processing to it. After that, they use OCR technology to recognize letters and numbers the accuracy of The detecting and recognition as shown in Table 1.1 .

Reference	Techniques used	Dataset	Result
[5]	Sobel edge. Morphological operation	English	-
[6]	Boundary line-based technique. Contour algorithm- OCR	English	92.85%
Current study	Canny edge detection With horizontal projection - OCR	English	96.0%

Table 1.1: Recognition Result Comparison

Safaa Omran et al. [12], In this paper they want to recognize the characters on the Iraqi plate, they first retrieved the plate position, then used segmentation to separately separate the plate characters, followed by An optical character recognition (OCR) is used with correlation approach and templates matching for plate recognition, this technique is intended to identify Iraqi license plates, and it has been tested on 40 different photos. Finally, it is demonstrated that the entire system performance is 86.6% recognition rate, with 87.5% accuracy for extraction of the plate region and 85.7% accuracy for the recognition unit.

Unit of LPR system	Number of accuracy	Percentage of accuracy
Extraction of plate region	35/40	87.5%
Recognition	30/35	85.7%

Table 1.2: Recognition Result Comparison

Hyuntae Kim et al.[13], In this research, a CNN model was used to identify and extract license plates, as it downloaded more than 500 images of license plates in different positions, then trained the model to find the best accuracy, and then used the DIGITS feature to identify the numbers and letters wich is a web-based application that simplifies the process of training deep neural networks (including convolutional neural networks or CNNs) on GPUs (Graphics Processing Units).

Input Image (Number)	Recognition Rate(%)	Error Rate (%)
63	95.24	4.76

Table 1.3: Total experimental results

Hui Li et al. [14],In this paper they have presented a jointly trained network for simultaneous car license plate detection and recognition. With this network, they detected and recognized the car license plate all at once in a single forward pass, with both high accuracy and efficiency. By sharing convolutional features with both detection and recognition network, The whole network can be trained approximately end-to-end, without intermediate processing like image cropping or character separation.

Method	End-to-end Performance (%)	Detection-only Performance	End-to-end Speed(per image single scale)(ms)
Ours(Jointly-trained)	97.13	98.33	310
Ours(Two-stage)	94.09	97.05	450

Table 1.4: Results

1.3 Discussion

After we reviewed the research related to our research, in this section we will evaluate the research used to solve the problem of parking lots and its effectiveness on the ground.

At the beginning, Suthir et al.[1] relied on the Internet of Things technology, which had a positive impact on solving the parking problem, such as by checking the person who booked the parking lot according to his license plate and making sure that the parking lot was not occupied before the arrival of the car. But to apply this technology, there are drawbacks, such as the high cost and the fact that it uses a sensor for each parking lot and the system stops if the payment system fails, and this negative is the most dangerous.

Ibrahim et al. [2] used a camera based on image processing technology to photograph car parks, which made it possible to detect the largest number of cars and parking lots at one time through aerial photography of the parking lots. One of the positive aspects that he relied on in the project is the method of supplying the system with energy. Solar panels as a source of energy, which makes this project environmentally friendly and reduces the chance of system downtime. Just as there are positives, there are negatives. When this camera is placed in indoor parking, it does not detect parking as well as when it is placed in outdoor parking. Also, the weather condition affects the camera and the energy negatively, for example, if the weather condition is overcast with clouds, the power source will be less efficient than when the weather is clear, and so on.

Fariza et al.[3] used image processing and MATLAB as a software platform to detect the status of the parking lots and display their results on the LCD screen. One of the advantages of this project is its low cost. It employs the camera as a single sensor capable of detecting all parking lots in a single image, as well as the ease of moving the camera in any direction for a better result and for this project. The disadvantages,

including the location of the camera, must be high in order to capture the largest number of parking lots; otherwise, you need more than one camera to cover the parking lots.

Amin et al.[4] distinct from the others because it is based on ultrasonic sensors. Among its advantages, when the sensors send inputs to the system, the LED lights up green if it is not busy or red if it is busy. This will help the driver more because it will reduce the time to search for the parking lot. Also, this project is of high quality for internal and large parking spaces because it was designed for four floors of parking lots, and each floor has 100 parking spaces. As for the negatives, this project is very expensive because it is designed for large buildings.

Ashu et al.[5] note that this project is similar to some of the previous projects that were discussed in terms of discovering the condition of parking lots using image processing, but in addition to that, methods were used to detect vehicles and their structure, which helps to show the advantages of one of them: the speed in determining the type of vehicle, whether it is a car or something else; unlike some other projects that do not recognize the type of vehicle covering the parking space, when the parking lines are not clear, the process of identifying the parking is not completed.

Yusmeeraz et al.[6] It is similar to the previous projects in terms of detecting parking status using image processing technology. But it was distinguished by the fact that the project was designed to be a flexible system that is not intended for one parking area only, and added to that was the feature of parking reservations. This project is also distinguished by the fact that it is inexpensive because all you need is one camera for the detection of parking lots. But just as there are advantages, there are disadvantages. As the application was designed for reservations, parking is only available for Android users. Also, this system will be expensive for indoor parking because you will need more than one camera to detect parking lots.

Chetan et al.[7] replace the sensors in the camera because they will get the number of parking spaces and detect the locations of cars using real-time images. In this way,

it will reduce the overall cost of this idea and be more flexible than building a car parking system. Also, the camera can detect parking even if there are no lines on the road. But if the camera sees something other than the car, it means that the parking lot is already full and not available.

In Tom's research [8], he used deep learning techniques (CNN) with image processing, which helped raise the accuracy of detecting the parking status. One advantage of Tom's research is the low cost of implementing the high accuracy rate. On the other hand, an important flaw is that it depends a lot on the Internet, and indoor parking may be expensive.

Jatuporn et al.[9] use of the WSN depends on sensors. This seems like a good idea because it's difficult to afford sensors if there are a lot of parking spaces, and their idea to include motorcycles in the system detection is brilliant. Indeed, the range of vehicle sizes may not be very accurate. The payment system they developed is smart; the customer can pay with their credit card or debit, which makes the cars move faster.

Hongwei et al.[10] His idea relies on Bluetooth because there is no need to connect to the network. The disadvantage of Bluetooth is that it has a limited range that may cause problems, and some people may not have one. It also shows something good because you don't have to find a network to park in that area. It's low-cost because it's a decentralized system.

Rayana Antar et al. [11], relied on two techniques to extract the license plate letters: image processing and the OCR library to obtain all parts of the license plate, the Arabic and English parts. Also, prepare a table of Arabic letters and their equivalents in English, because not all English letters are present on the license plate of Saudi cars. However, the use of image processing without OCR did not produce good results, some plates were not detected on all the letters of the car, and other plates were not captured from the original

Safaa Omran et al. [12],In this research, the focus was on extracting the plate in the best form and reading the Arabic letters, and one of the advantages of this research

is that it works on more than one model of Iraqi car plates using image processing, where through this technology it is possible to identify the type of plate, but there are two drawbacks from our point of view first: The accuracy of the result in recognizing letters and numbers is 85.5%. Secondly, the accuracy of identifying the plate area is 87.5%, and we believe that the accuracy is low.

Hyuntae Kim et al.[13], in this research he use CNN model for detecting the car plate and he get a high accuracy for that but he has a some disadvantages that may affects the final result. The GPU he used is slow in execution time .

Hui Li et al. [14], In this paper they trained a network, with this network they detect and recognized license plate all at once in a single forward pass, without segmentation and it gives high accuracy and efficiency, but we think they used a small dataset like what was cleared from them.

1.4 Conclusion

In this chapter, we review previous works that presented some technologies that helped develop solutions to the problem of parking in cities. These works were discussed, as well as their advantages and disadvantages. In the next chapter, we will introduce our methodology that makes it possible to solve the problem cited above.

2 Methodology

2.1 Introduction

In the last chapter, we mentioned how useful image processing technology is for solving the parking problem. In this chapter, we will provide a detailed description of our methodology using image processing to facilitate access to parking . We will explain how to capture license plate information And convert it into recognizable text.

2.2 General description of the proposed approach

In this section, we provide a general description of our proposed approach. There is more than one effective methodology to solve the problem. In fact, our idea is based on using IR sensors to detect the parking spot, then using image processing and a convolutional neural network (CNN).

In image processing, the license plate will be recognized using filters to make the image clearer. Then, we use deep learning to analyze the images and read letters and numbers.

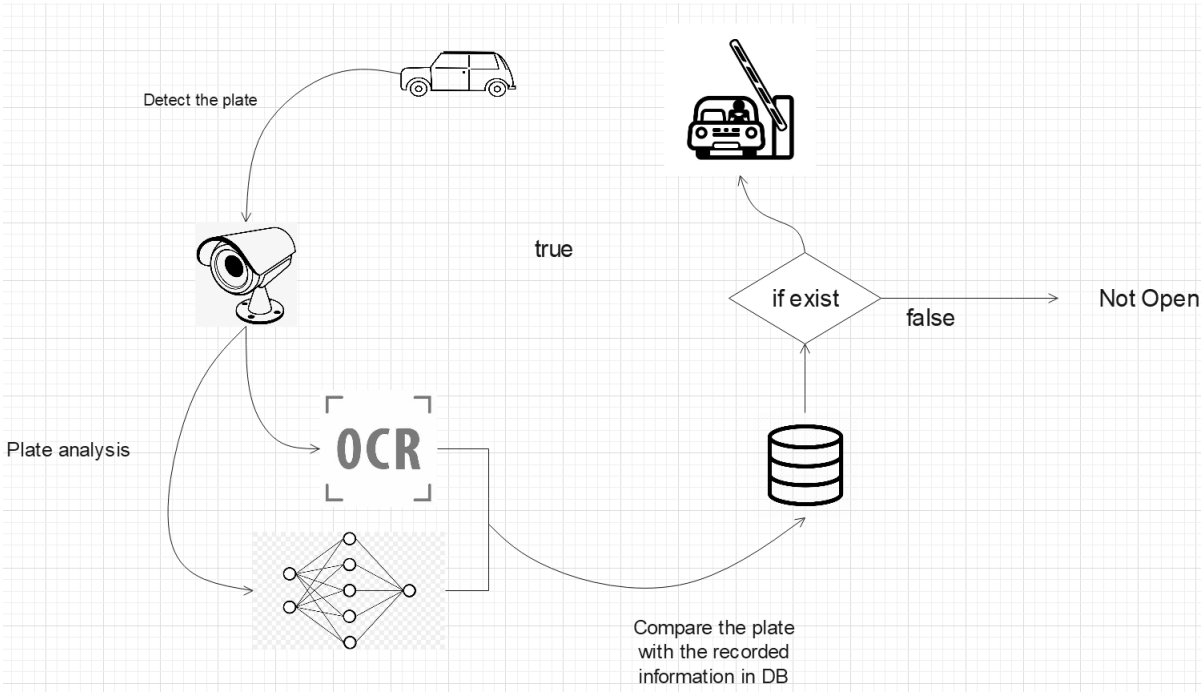


Figure 2.1: System Approach

2.3 System Approach

In this section, we show how the system will work and what we will use depends on two basic sections

- Pre-processing
- CNN Approach

2.3.1 Pre-processing

The purpose of pre-processing is to improve the quality of image data by ignoring unnecessary details.
Read Image, in this step we read images in different formats like png, jpg.
Image segmentation is used to convert the image from RGB level to grayscale using Eq.1, because gray scaling is common in all image processing steps. This speeds up other following processes since we no longer have to deal with the color details when processing an image.

$$0.2989 * R + 0.5870 * G + 0.1140 * B \tag{1}$$

Image enhancement Most of the images will have useful and unuseful details, The plate numbers and letters are useful details for us, and the useless details are called "Noise ", there are many filters that enhance the image and they consist of two types:

- linear
- non-linear

We suggest using non-linear filters because they remove noise better than linear. There are many filters such as median filter, Min-Max filter, and bilateral filter which is more helpful in our approach Eq.2, because it makes the image smoother and easier to detect edges.

$$I^{filtered}(x) = \frac{1}{W_p} \sum_{x_i \epsilon \Omega} I(x_i) f_r(||I(x_i) - I(x)||) g_s(||x_i - x||) \quad (2)$$

Edge detection part appears the shape of the object. So we must use edge detect filter, such as sobel filter, Laplacian filter, and canny edge filter which is the easiest, most popular way and better in detecting circular edge and convert it to binary image with detected edges.

Character segmentation part segments each character of the plate as an image and send it to CNN to recognize the character Fig.2.2.

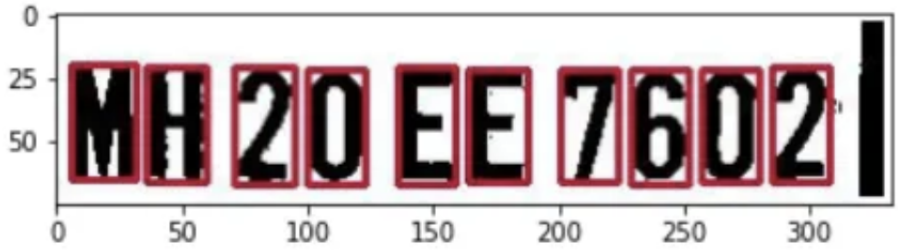


Figure 2.2: Character segmentation

2.3.2 OCR

Optical character recognition (OCR) is the process of converting an image, such as a scanned paper document or electronic fax file, into computereditable text. The text and digits in an image is not editable: the letters/characters and digits are made of tiny dots (pixels) that together form a picture of text §and numbers. During OCR, the software analyzes an image and converts the pictures of the characters and digits to editable based on the patterns of the pixels in the image. After OCR, you can export the converted text and use it with a variety of wordprocessing, page layout and spreadsheet applications. OCR also enables screen readers and refreshable Braille displays to read the text contained in images.

2.3.2.1 OCR approach

Optical character recognition (OCR) uses a scanner to process the physical form of a document. Once all pages are copied, OCR software converts the document into a two-color or black-and-white version. The scanned-in image or bitmap is analyzed for light and dark areas, and the dark areas are identified as characters that need to be recognized, while light areas are identified as background. The dark areas are then processed to find alphabetic letters or numeric digits. This stage typically involves targeting one character, word or block of text at a time. Characters are then identified using one of two algorithms — pattern recognition or feature recognition.

2.3.3 CNN approach

Convolutional neural networks are a development of artificial neural networks. They use pooling layers and convolutional layers differently. In CNN, the Convolutional always comes first, Fig2.3 shows how it works.

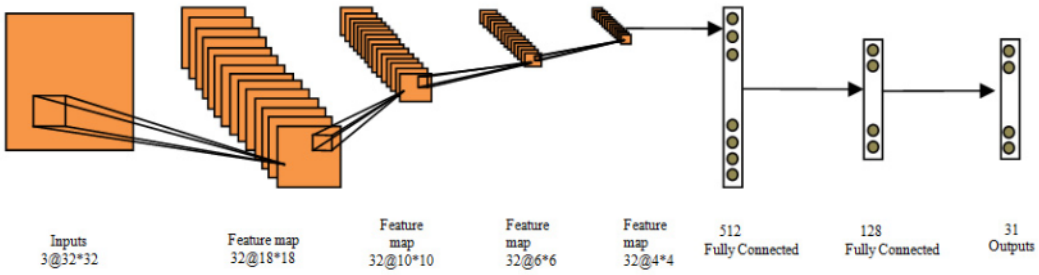


Figure 2.3: CNN approach

Convolutional layer Let's picture the convolutional layer as a yellow light that glides around the region while shining over the image. It multiplies the values in the filter by the original pixel values of the picture in the extremely straightforward convolutional procedure. The Convolutional layer's output is either an activation map or a feature map.

Pooling Layer The pooling layer is an important Layer used After The Conventional Layer to reduce the dimension of the hidden Layer, there is a variety of pooling Layers:

- Average Pooling
- Max Pooling
- Global Max Pooling

And the most used layer is Max pooling, which is a pooling operation that calculates the maximum value for patches of a feature map, and uses it to create a downsampled (pooled) feature map

Fully Connected Layer Finally, the fully connected layer comes after the convolutional and max-pooling layer, the neuron applies a liner transformation to the input

vector through a weights matrix, and fully connected layers have neurons that have connections to all activation in the previous layers.

2.4 Proposed transferring approach

In CNN the dataset is divided into three categories training, validation, and testing, there are various CNN models, The main difference between CNN models lies in the architecture, which affects the number of layers, the type of layers, and the number of parameters. Additionally, different CNN models may use different activation functions, pooling layers, and regularization techniques. In the future there are CNN models will be tested :

- GoogleNet
- LeNet
- ResNet

2.4.1 GoogleNet

Is a deep convolutional neural network developed by Google. It is a 22-layer network consisting of convolutional layers, pooling layers, normalization layers, and fully connected layers. It is designed to use fewer parameters while providing better accuracy than traditional network architectures. GoogLeNet is presently used for a range of computer vision applications, such as adversarial training, face detection, and identification. GoogleNet contains nine modules of InceptionNet architecture stacked on top of each other. (see Fig.2.4)

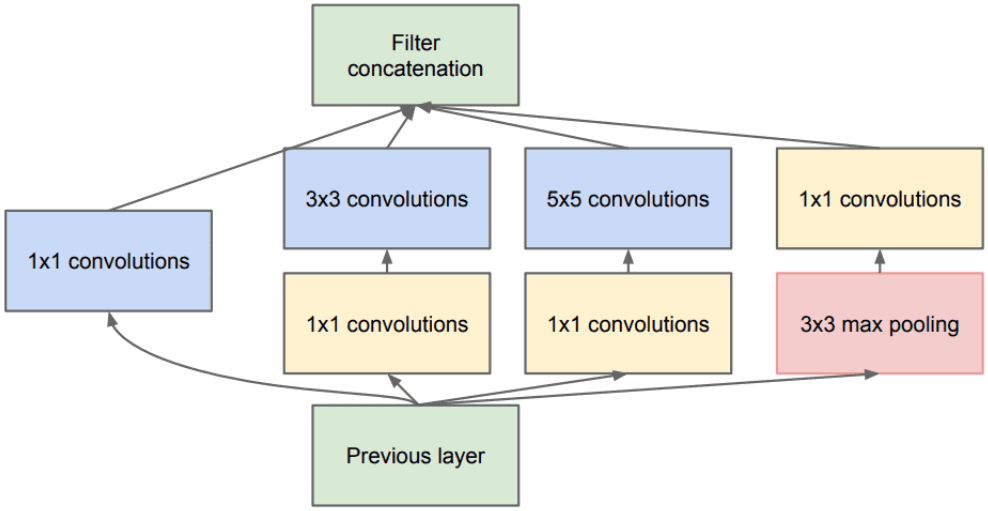


Figure 2.4: Single InceptionNet Module

2.4.2 LeNet

LeNet is a convolutional neural network that was developed by Yann LeCun. It is a model of a traditional neural network and is used for character recognition in documents such as bank checks. The architecture of the LeNet model consists of two sets of convolutional and average pooling layers, followed by a flattening layer and two fully-connected layers. The convolutional layers extract features from the input image, and the fully-connected layers classify those features into various categories. The LeNet model has been used in different computer vision tasks, including classification, object detection, and image segmentation. (see Fig.2.5)

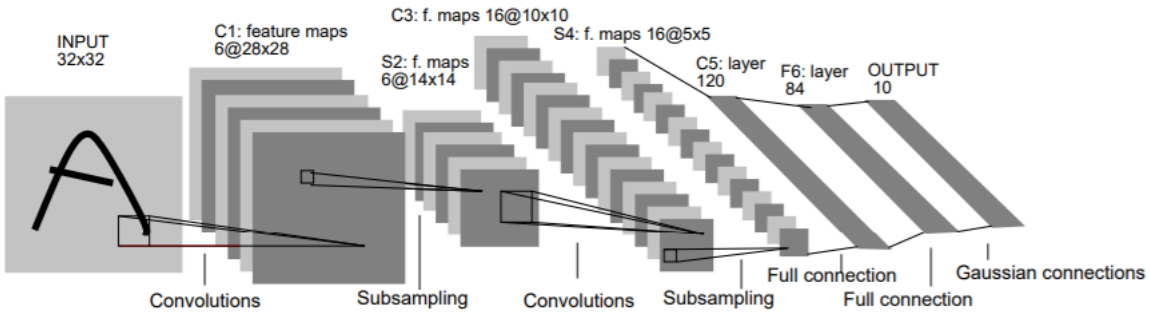


Figure 2.5: LeNet architecture

2.4.3 ResNet

ResNet is a convolutional neural network (CNN) architecture developed by Microsoft Research. It is one of the most widely-used CNN architectures for image classification and has been the basis of many modern image recognition systems. ResNet is notable for its ability to enable the training of very deep neural networks — up to 152 layers — with fewer training parameters than traditional networks. This is achieved by using a "skip connection" method, whereby the output of one layer is added to the input of a deeper layer. The architecture is also known for its excellent generalization capabilities, allowing it to achieve high accuracy on unseen data. (see Fig.2.6)

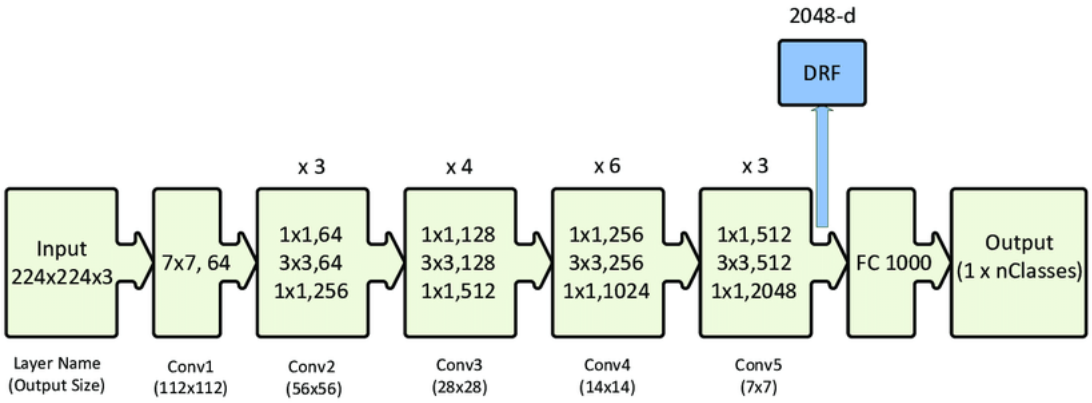


Figure 2.6: ResNet architecture

2.5 Conclusion

In this chapter, We present our methodology in the way to solve the problem using pre-processing approach which involved read image, image segmentation, image enhancement, edge detection, also we identify The OCR and how it works ,finally we review CNN approach which includes three parts of layers.

3 Requirement Structuring

3.1 Introduction

In this chapter, we shows some diagrams that describe the environment between the system and clients.

3.2 Use-Case Diagram

Use-Case diagrams describe the high-level function and scope of the system. In 3.1 we design the system use-case diagram, and we described what the services provide to the user from the system.

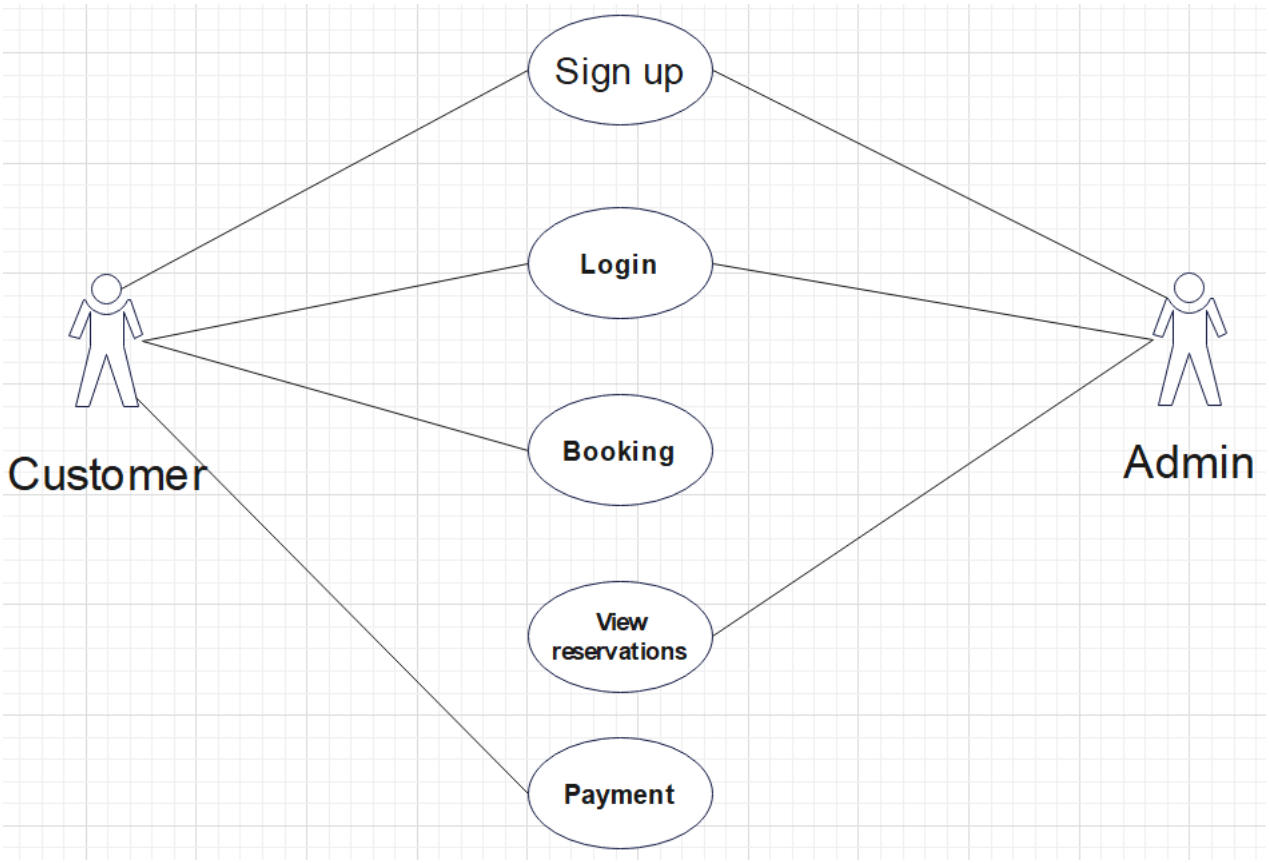


Figure 3.1: Use-Case Diagram

3.3 Sequence Diagram

The sequence diagram described the lifetime of the system, In Fig.3.2 we show our system lifetime. The barrier will open if the camera detects the car plate and make sure if it's found in the database after that the car will go to the parking slot if the plate reader doesn't validate the plate the barrier will not open.

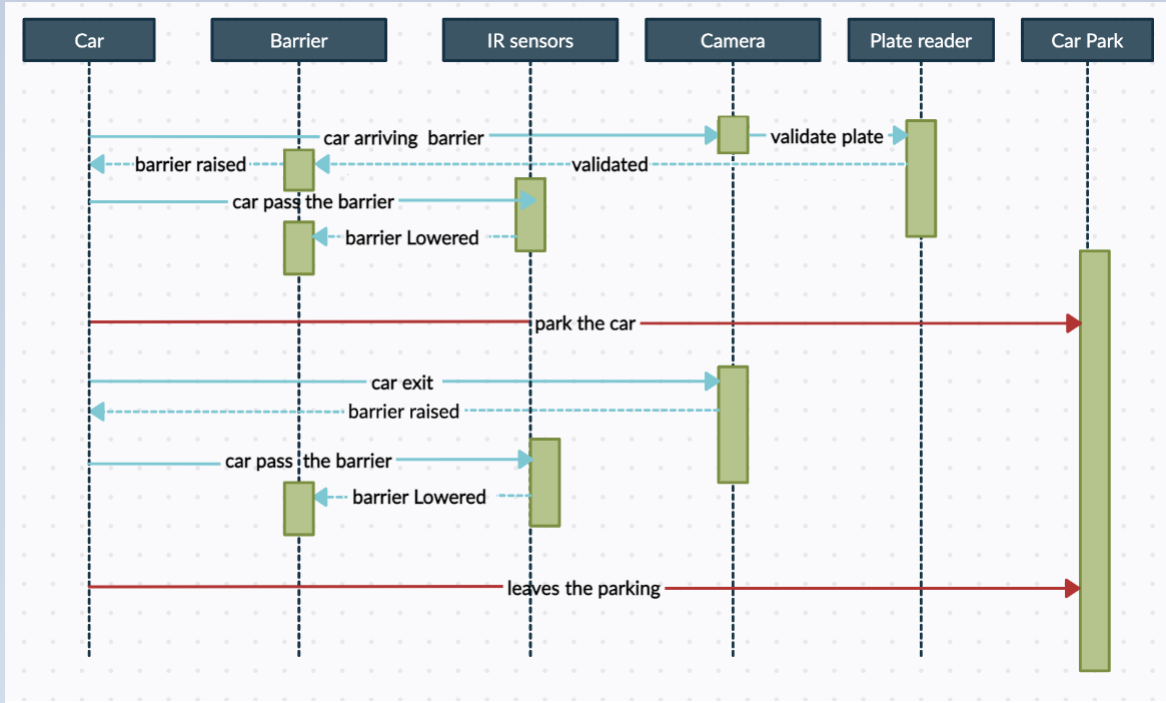


Figure 3.2: Sequence Diagram

3.4 Class Diagram

Class Diagram Diagram described the relationships, classes, and attributes, In 3.3 we described the system what are the classes and the relationships between them, and the attributes we must use.

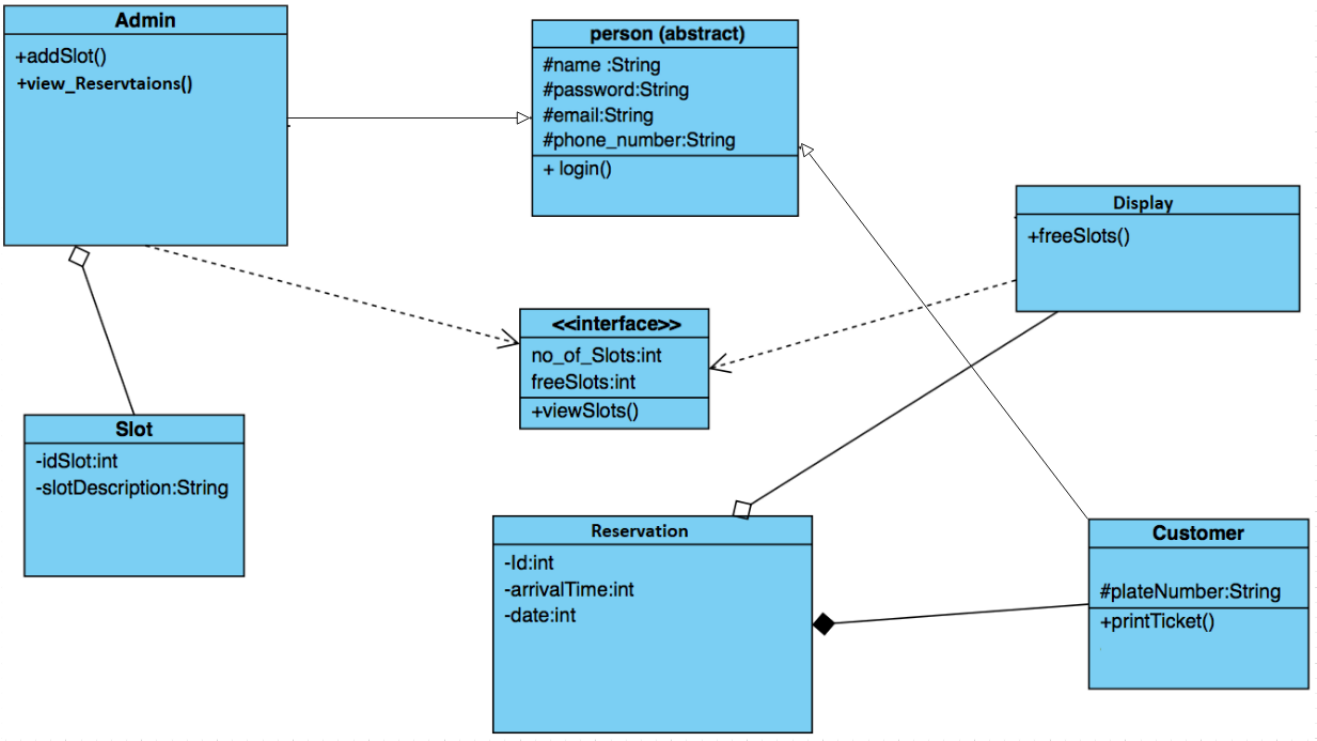


Figure 3.3: Class Diagram

3.5 Conclusion

In this chapter, we presented our diagrams such as Uses-Case, Sequence, and Class diagrams that describe how the system works and the environment of the system.

4 | Experimental Methods

4.1 Introduction

In the previous chapter, we presented in detail the approach that we propose for this project. This chapter presents our conducted experimentation and evaluates the obtained results. Thus, we first define the implementation environment. Then, we present the experimental results.

4.2 Implementation Environment

This section presents the implementation environment of the CNN model classifiers and the OCR model, in order to construct each model we used Python programming language in Google colab notebook.

4.2.1 Python

Python is a popular, easy-to-learn programming language known for its readability and versatility. Its concise syntax and large library make it a popular choice for a variety of applications, from web development to data analysis and machine learning, and we used some of them, such as:

- Matplotlib: is a Python charting framework that works with NumPy, offering an object-oriented API for creating charts in programs. It is widely used in scientific computing and is utilized by SciPy.
- Tensorflow: It provides a versatile set of tools for building and training models, supports CPU and GPU computations, and integrates well with other libraries.
- Cv2: Python library that allows you to perform image processing and computer vision tasks.
- Numpy: Tool for manipulating matrices and arrays, required for all machine learning applications.
- Easyocr: It uses deep learning models to extract text from images.

4.2.2 Dataset

In The implementation, we used a dataset for the CNN model that contains 1080 images of characters in different formats, then we divided into three parts: 864 train sets, 108 validation sets, and 108 testing sets as shown in fig 4.1.

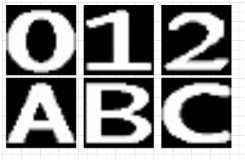


Figure 4.1: Sample of datasets

4.3 Implementation

In this section we show the implementation of the project

4.3.1 Image processing

In this part we use two library: Cv2 , Matplotlib

Step 1: **Read the Image and Convert to Gray scalse**

Figure 4.2: original image , gray image

Converting an image into grayscale is an important step in image processing because simplifies the image, reduces dimensionality, removes color variations, emphasizes texture and structure, and ensures compatibility with various algorithms. However, preserving color information may be necessary for specific tasks.

Step 2: **Apply bilateralFilter and Canny edge detection**

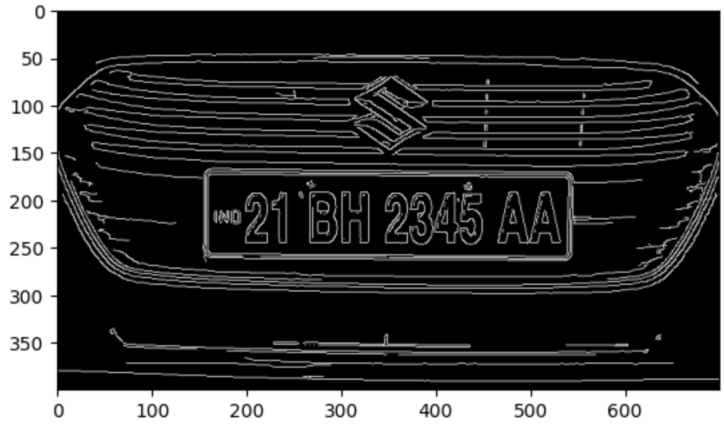


Figure 4.3: BilateralFilter, Canny Edge detection

Bilateral filter and Canny edge detection are image processing techniques used together for edge enhancement and noise reduction. The bilateral filter smooths the image while preserving edges, reducing noise. Canny edge detection detects sharp edges accurately. By applying the bilateral filter to reduce noise and refine edges, followed by Canny edge detection to enhance strong edges, a clearer and more accurate representation of image features is achieved.

Step 3: **Search for contour**

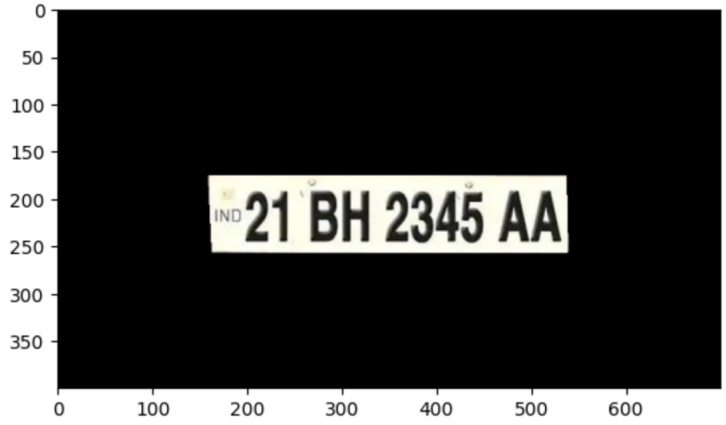


Figure 4.4: Finding best Contour

Looking for the best Contour with 4 vertices in our image to get just the shape of the car plate, to use for the next step

Step 4: **Characters segmentation**

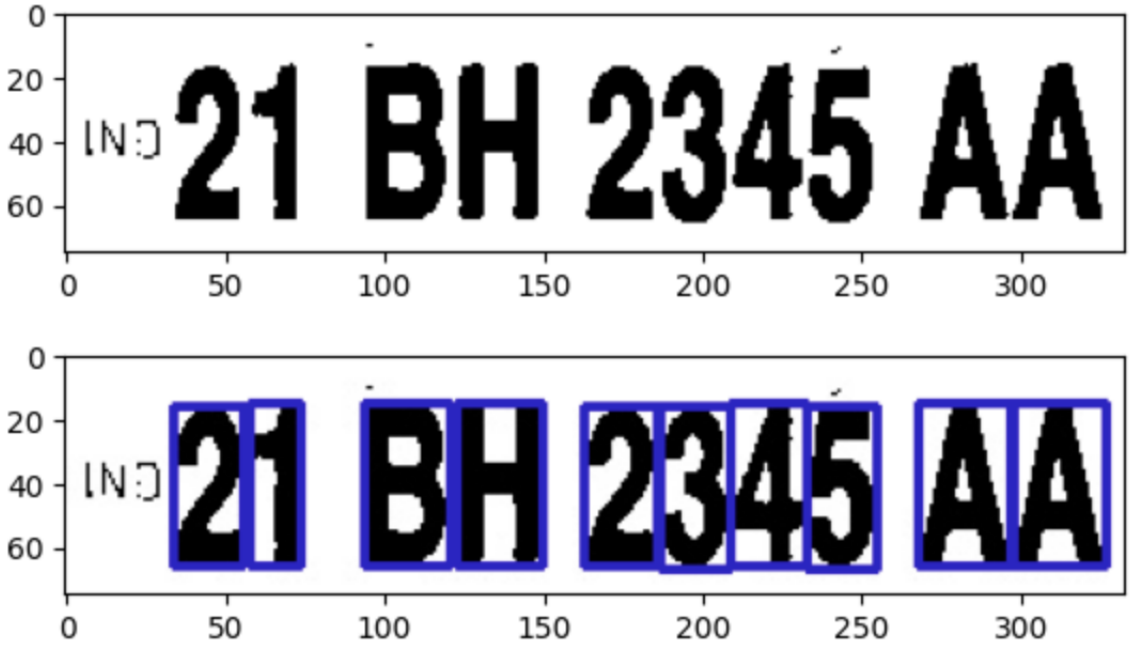


Figure 4.5: Characters after segmentation

Segmenting characters in an image is an important step in character recognition. It allows individual characters to be processed and recognized separately, which can improve the accuracy of the algorithm. Segmentation identifies the boundaries of each character and separates them from the rest of the image.

4.3.2 OCR

in this part we use the library: easyocr

Figure 4.6: OCR Result

The accuracy of OCR can vary depending on various factors, including the quality of the input image, the clarity and legibility of the text, the language and font used, and the complexity of the image layout. OCR accuracy can be influenced by factors such as noise, blur, low resolution, unusual fonts, and skewed or distorted text.

4.3.3 CNN

Step 1: **Designing architecture**

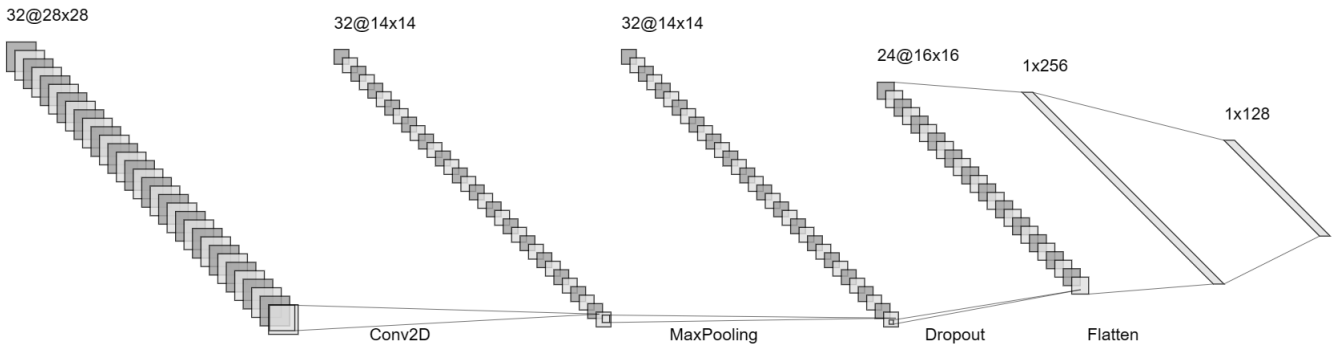


Figure 4.7: CNN architecture LeNet style

This CNN architecture defines consists of a series of layers, including a convolutional layer, pooling layer, dropout layer, and dense layers. The model is designed to take 28x28 RGB images as input. It uses the ReLU activation function for the convolutional layer and the softmax activation function for the output layer to produce class probabilities

Step 2: **Training model**

```
Epoch 5/20
216/216 [==============================] - 4s 16ms/step - loss: 0.3189 - accuracy: 0.8657
Epoch 6/20
216/216 [==============================] - 4s 17ms/step - loss: 0.3073 - accuracy: 0.9120
Epoch 7/20
216/216 [==============================] - 5s 22ms/step - loss: 0.3003 - accuracy: 0.9028
```

Figure 4.8: Our model training

Training a model is of utmost importance in the field of machine learning and artificial intelligence. Training involves feeding a model with a large amount of data and enabling it to learn patterns, relationships, and representations from that data, Some reasons why training a model is important:

- Learning and Generalization
- Pattern Recognition
- Optimization and Improvement

Step 3: **Testing CNN model after training**

Figure 4.9: CNN Result

In this stage, we check the accuracy of the model we created and trained to identify letters and numbers.

4.3.4 Source code

This link contains the code that was used in the project:

Google colab

4.4 Accuracy result

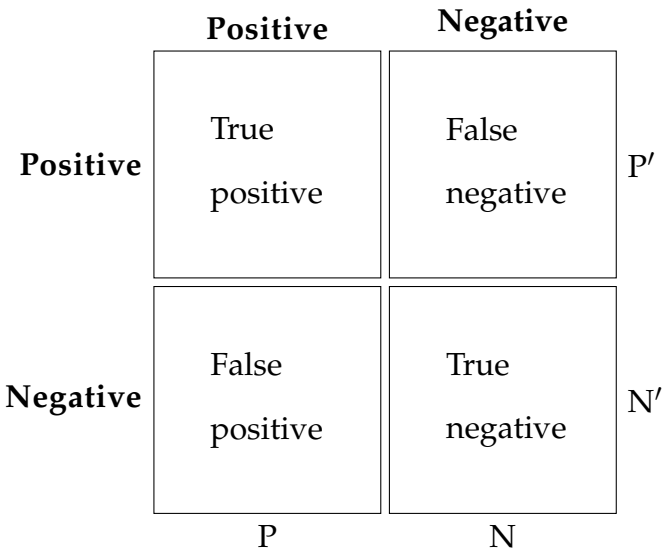
This section presents the results achieved by the CNN and OCR models. To evaluate each model performance, we use the following measures: accuracy (Equation 1), precision (Equation 2), Recall (Equation 3) and F1-score (Equation 4).

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{1}$$

$$Precision = \frac{TP}{TP + FP} \tag{2}$$

$$Recall = \frac{TP}{TP + FN} \tag{3}$$

$$F1 = \frac{2 * Precision * Recall}{Precision + Recall} = \frac{2 * TP}{2 * TP + FP + FN} \tag{4}$$



This table shows that TP (True Positive) referred to the case where the actual label and the predicted label are positive. While FN (False Negative) is when the actual label is positive and the predicted label is negative. FP (False Positive) is the actual label is negative but the predicted label is positive. Thus, TN (True Negative) is when the actual label and the predicted label is negative.

Method	Accuracy	Precision	Recall	F1-score
OCR	85.92%	81.81%	78.71%	79.47%
CNN	97.22%	93.5%	90.2%	91.82%

Table 4.1: Methods and Accuracy results

Overall, CNN methods are often preferred over OCR methods for character recognition tasks because they are specifically designed to extract features from images, can be trained end-to-end for higher accuracy and reduced error, and are highly adaptable to diverse input images.the accuracy result is obtained on CNN higher than OCR, as shown in table 4.1 it is likely due to these advantages of CNN.

Research	CNN	OCR
[11]	-	96%
[12]	-	87.5%
[13]	95.24%	-
[14]	98.33%	-
Current study	99%	85.92%

Table 4.2: Research comparison

as shown in 4.2, accuracy was compared in several studies, after comparing it with the results obtained in our research.

4.5 Discussion

In this section, we provide and discuss the results of our research in order to answer the mainly essential research question: Why we use Bilateral and canny edge filters ? We used a Bilateral filter to smooth the image and clarify all special edges in the image, and then we use the Canny edge filter to define edges before it is extracted. Why we use OCR to recognize the license plate ? We have used OCR technology to recognize the letters and numbers in the car plate and then we compare the outputs of the same technology in CNN. Why we want to compare CNN and OCR outputs? In order for us to find the best accuracy for the correctness of letters and numbers. Why we choose this dataset ? Because the dataset that we used has several possibilities for most possible conditions to know letters and numbers.

4.6 Conclusion

In this chapter, we presented our implementation and how we applied image processing filters, using OCR to recognize license plate, train dataset with CNN to recognize license plate too and show accuracy, precision, recall, F1-score to each one.

5 | General Conclusion

5.1 Conclusion and future work

After we read several previous projects, checked their content, and deduced the best tools used in them, we came up with a specific way to solve the problem in our own way. First, we used image processing to read the image and then convert the image to a grayscale image to apply several filters to it, including the bilateral filter, which in turn deletes the noise and softens the image. After that, we used Canny Edge Detection to show the borders in the image, and then we used a feature to select and cut the target area, followed by the segmentation feature to extract letters and numbers separately. In reading letters and numbers, we used two models, namely the OCR and CNN models. In OCR, each letter is read, distinguished, and printed. It is a ready and trained library that is used to recognize letters and numbers through images only, but in the CNN model, we have dataset that will carry many images of letters and numbers in different positions, and then we trained the model and tested the accuracy of its knowledge in distinguishing letters and numbers, and all of this was applied in the Python language using several libraries, In the future, we will try to avoid all the threats associated with this project as much as possible and produce successful outcomes. Depending on many image processing algorithms and many CNN architectures other than that we used.

Threats to validity

In this section, we discuss the threats to validity, there are three different threats that may affect the validity:

— First, sensor failure, which may affect the number of parking spaces available in the system

— Second, weather factors that may affect the camera's view of the license plate

— Third, the website or mobile application may shut down due to an attack or a large number of users

Bibliography

[1]. Suthir S, Pon Harshavardhanan , Kavitha Subramani, P. Senthil, T. Veena , Julia Faith S , Nivethitha V . “Conceptual approach on smart car parking system for industry 4.0 internet of things assisted networks” . VIT Bhopal University, India . Published online 4 October 2022. [CrossRef]

[2]. Hilal Al-Kharusi, Ibrahim Al-Bahadly . ”Intelligent Parking Management System Based on Image Processing”. School of Engineering and Advanced Technology, Massey University, Palmerston North,New Zealand. Published Online May 2014. [CrossRef]

[3]. R. Yusnita, Fariza Norbaya, and Norazwinawati Basharuddin. “ Intelligent Parking Space Detection System Based on Image Processing”. International Journal of Innovation, Management and Technology, Vol. 3, No. 3, June 2012 . [CrossRef]

[4]. Amin Kianpisheh, Norlia Mustaffa, Pakapan Limtrairut and Pantea Keikhosrokiani. “ Smart Parking System (SPS) Architecture Using Ultrasonic Detector”. Universiti Sains Malaysia(USM), Malaysia. Published online July, 2012. [CrossRef]

[5]. Ashu Yadav, Kirti Bhatia. “Car parking space detection using Image Processing” . Sat Kabir Institute of Technology and Engineering, Bahadurgarh, Haryana. Published online 2021. [CrossRef]

[6]. Daniel Ng Chiu Loong, Suhaila Isaak, Yusmeeraz Yusof. “ Machine vision based smart parking system using Internet of Things”. School of Electrical Engineering, Faculty of Engineering, Universiti Teknologi Malaysia. Published Online August

2019 . [CrossRef]

[7]. Chetan Sai Tutika, Charan Vallapaneni, Karthikeyan. “Image Handling and Processing for Efficient Parking Space Detection and Navigation Aid” . School of Electronics Engineering VIT University, India , 2019. [CrossRef]

[8]. Tom Thomas,Tarun Bhatt. “Smart Car Parking System Using Convolutional Neural Network” . Department of Electronics & Telecommunication Padre Conceição College of Engineering Verna Goa. Published online 03 January 2019. [CrossRef]

[9]. Jatuporn Chinrungrueng Udomporn Sunantachaikul Satien Triamlumlerd. “Smart Parking: an Application of optical Wireless Sensor Network”. Thailand Ministry of Science and Technology. Published 2007. [CrossRef]

[10]. Hongwei Wang. “A Reservation-based Smart Parking System ”. University of Nebraska-Lincoln. Published July 2011. [CrossRef]

[11]. Rayana Antar, Shimaa Alghamdi, Johara Alotaibi, Manal Alghamdi. “Automatic Number Plate Recognition of Saudi License Car Plates ”,Umm Al-Qura University ,vol.12 ,2022. [CrossRef]

[12]. Safaa S. Omran, Jumana A. Jarallah. “Iraqi Car License Plate Recognition Using OCR”,pp.4-5, march 2017 . [CrossRef]

[13]. Lee, S., Son, K., Kim, H., & Park, J.,"Car plate recognition based on CNN using embedded system with GPU". 2017 10th International Conference on Human System Interactions. [CrossRef]

[14]. Hui Li , Peng Wang, and Chunhua Shen.“Toward End-to-End Car License Plate Detection and Recognition With Deep Neural Networks". 2018. IEEE Transactions on Intelligent Transportation Systems”. [CrossRef]