ISSN: 2277-3754
ISO 9001:2008 Certified
International Journal of Engineering and Innovative Technology (IJEIT)
Volume 4, Issue 4, October 2014

Practical Design of an Automatic License Plate Recognition Using Image Processing Technique

Mohammad Zakariya Siam
Electrical Engineering Department, ISRA University, Amman-Jordan

Abstract: ***The strategy of this paper is to present a practical design of an automatic system for opening a gate without mounting any signal transmitter on the car. License plate recognition (LPR) is an image processing technology used to identify vehicles by their license plates. This paper's goal is to build a practical prototype system, which is capable of recognizing a license plate number from a standard license plate. Thus, this paper is to investigate and construct an application whereby the system will recognize the vehicle license plate at a gate entrance of the parking lot. The system will be based on a personal computer and software packages available such as MATLAB and a digital camera that helps in capturing images of vehicles. The software recognizes the plate number, compares the plate number with a built in database, and decides whether a vehicle is allowed to enter the designated area or not. The general algorithm involves the following steps:***

- ***Image capturing which can be achieved by a digital camera.***
- ***Plate localization and extraction to obtain the vehicle plate sub image.***
- ***Character segmentation to determine exactly where characters exist inside the plate.***
- ***Recognition which identifies the numbers contained in the plate.***
- ***Evaluating the performance of the algorithm.***
- ***Designing a database to store the numbers of authorized vehicles that are allowed to enter the parking.***
- ***Designing a graphical user interface (GUI) to simplify the interaction with the software.***

I. INTRODUCTION

A. Background

Vehicle license plate recognition is an image processing system whereby it is used to recognize the vehicles by identifying the license plate. It can be used for traffic control, security purposes, and parking access control. The license plate system works as follows:

- Firstly, the vehicle will stop at the car gantry. The cycle will start when the vehicle steps over the detector. It will activate a signal to the vehicle license plate system for the presence of the vehicle.

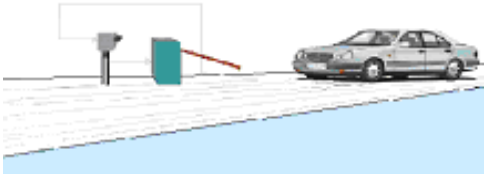


Fig 1: Step one: Car stops at the entrance of parking lot.

- Secondly, illumination will be activated and images of the front picture of the vehicle will be taken. Then the system will read the information pixels of the vehicle and run the recognition process.

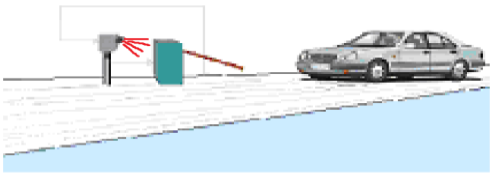


Fig 2: step two: An image is captured for the vehicle.

- Thirdly, the system will apply certain algorithm to analyze the vehicle image. If the registration plate number is inside the predefine list, the barricade will be lifted for the vehicle to go through. Otherwise, the vehicle will be denied entry.

Fig 3: step three: The vehicle is allowed to enter if it is authorized.

The steps of operation that the software follows are shown in the block diagrams below:

ISSN: 2277-3754
ISO 9001:2008 Certified
International Journal of Engineering and Innovative Technology (IJEIT)
Volume 4, Issue 4, October 2014

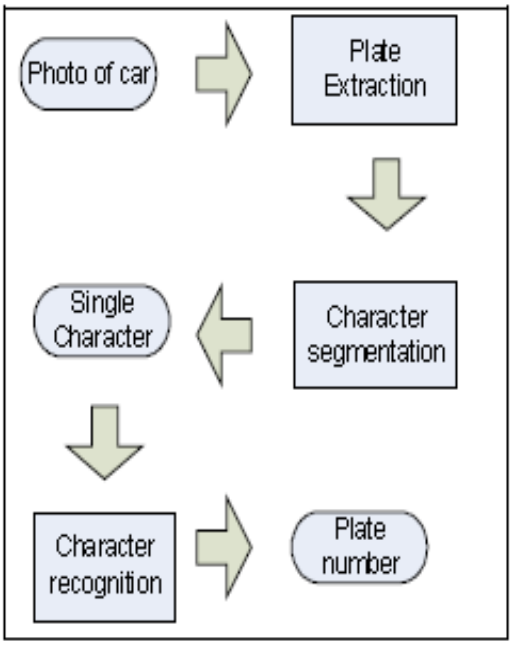


Fig 4: Software sequential operation.

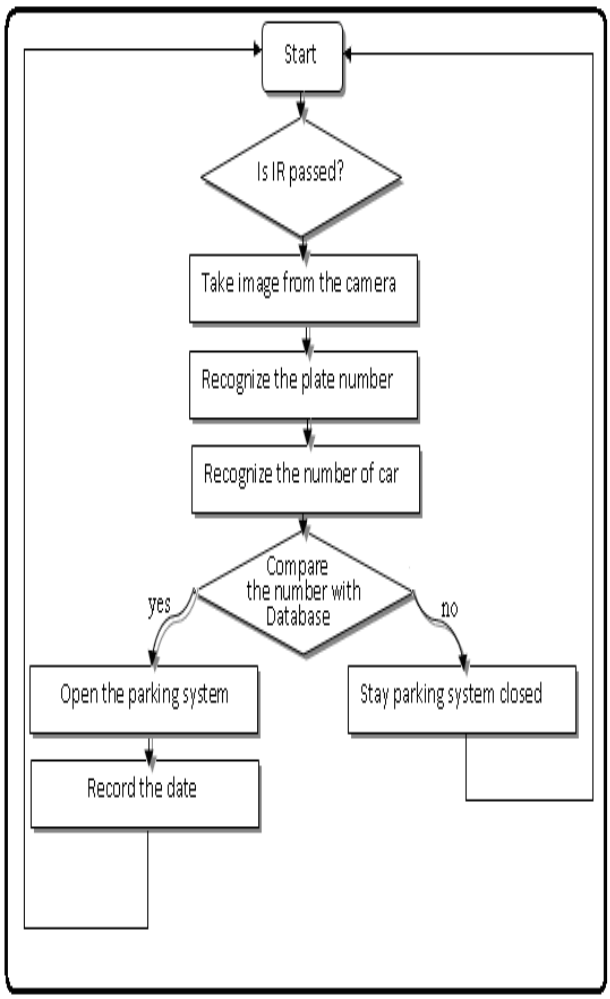


Fig 5: The flow chart of the recognition process

Over the recent years, there were quite number of methods for character/numeric and vehicle registration recognition developed. However, identification of vehicle registration plate number with respect to various natural viewing conditions still remains a challenging task. This is because:

- Vehicle registration plate number acquired is of different front sizes.
- Color of the vehicle registration plate number affected by varying illumination.
- Vehicle registration plate character might be chipped off.
- Algorithms must be able to be used in real time implementation.

There are difficulties for vehicle license plate recognition in which it will affect the efficiency and accuracy of the system. It is essential and important to determine the facts which will able to influence the operations and recognition proficiency. We also need to look into other facts of variables that are not constant. The following are the non-constant variables which affect the accuracy of recognition:

- Weather condition.
- Type of Vehicle.
- Distance between vehicle license plate and the camera.
- Type of plate (rectangular, bent type).
- Vehicle license plate orientation.

B. Objectives

The overall objective of this paper is to develop a system that recognizes vehicle license plate from a car at a gate entrance of a parking lot. The software could lead to a cheaper and faster way of enhancing and determined the performance of the recognition system. The system will be based on a personal computer which will generate a report on the vehicle license plate that has been captured. Once the vehicle license plate is captured, the characters will be recognized and displayed on the graphical user interface.

This algorithm development should be able to meet the requirements and goals as stated below:

- Able to detect the vehicle registration plate at a faster speed.
- Able to accurately recognize the vehicle registration plate.

There is definitely a lot more room for further improvement on this paper.

II. THEORETICAL PRINCIPLES

The following flow chart simplifies the operation of the system:

ISSN: 2277-3754
ISO 9001:2008 Certified
International Journal of Engineering and Innovative Technology (IJEIT)
Volume 4, Issue 4, October 2014

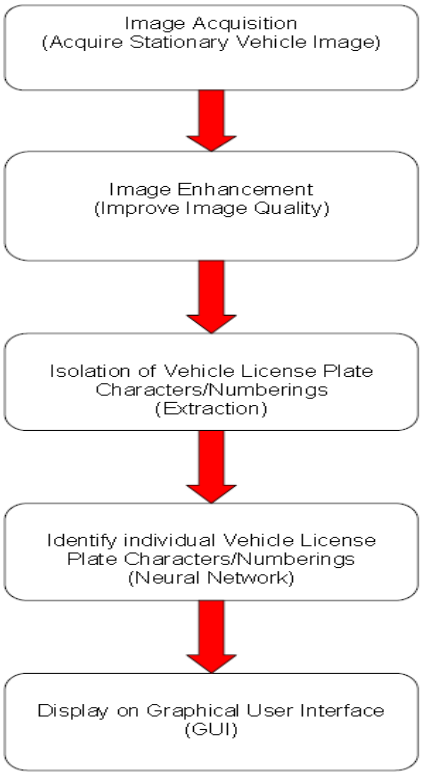


Fig 6: The flow chart of the system

B. Image Acquisition

The initial phase of the image processing for vehicle license plate recognition is to obtain images of vehicles. Electronic devices such as optical (digital/video) cameras, webcams, etc. can be used to capture the acquired images.

The images will be stored as color JPEG format on the camera. Next, we might proceed in using the Matlab function to convert the vehicle JPEG image into gray scale format.

Fig 7: Captured image

B. Plate Localization

1. Converting the Image to a Gray Scale

We now convert the image to a gray scale, which is important as we will take the edge that needs this form of image. Then we enhance the image, this step produces good results in some cases and approximately provides the same results in other cases.

Fig 8: Gray scale image

2. Taking the Complement of the Image

In this step, we take the complement of the image to enhance it, and will use the canny method to find the edge. The Canny method finds edges by looking for local maxima of the image's gradient. The gradient is calculated using the derivative of a Gaussian filter. The method uses two thresholds to detect strong and weak edges, and includes the weak edges in the output only if they are connected to strong edges. This method takes sigma as the standard deviation of the Gaussian filter, which controls the amount of details of the image; if sigma increases the details will decrease.

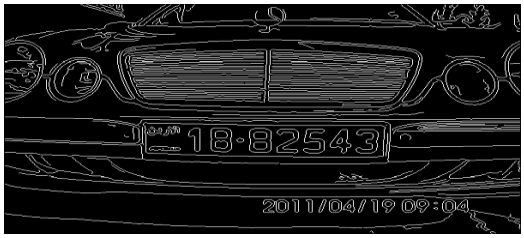


Fig 9: Complement of the image

In our practical part, we choose sigma of 0.95. However, if the process cannot get the plate then we change it until we reach the correct result.

3. Filtering the Image from Small Objects

We now convert the image into a double class because the functions that we use need this class. Then, we filter the image from any object of less than 280 pixels and close the image.

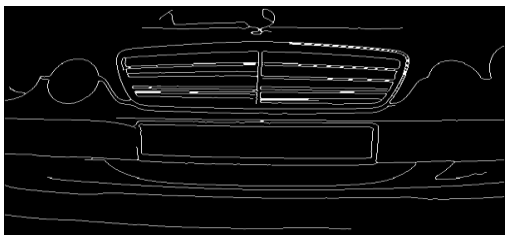


Fig 10: Filtered image

ISSN: 2277-3754
ISO 9001:2008 Certified
International Journal of Engineering and Innovative Technology (IJEIT)
Volume 4, Issue 4, October 2014

4. Labeling the Image to Objects

We now have an image that consists of objects. In order to decide the correct plate from these objects, we label them and find the characters of each object.

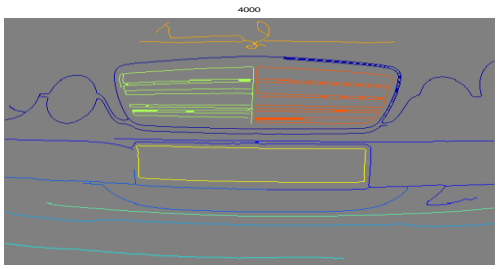


Fig 11: Labeling image

5. Separating Every Object

We then separate, fill, and rotate every object to give it specific characteristics that help in knowing the plate like the picture below:

Fig 12: Separating every object.

6. Recognizing the Plate

By a trial and error we could specify the suitable description solidity, the ratio between the height and width, convex area and bounding box. The last step is to display the plate by multiplying the plate object with the original image.

Fig 13: Recognizing the plate

C. Character Segmentation

Character Segmentation is an important step in the vehicle registration plate recognition system. However, with the influence factors such as image noise, spacing, brightness, different sizes and so on, there are quite a lot of difficulties faced. Character segmentation is basically the process of separating out the individual digits from a stretch of numbers images capture.

1. Threshold of Numeral Images

Before any character segmentation to be carried out, we have to convert the images captured into grayscale. This is basically changing all the pixels to either just black or white pixels depending on whether the pixels are above or below a defined threshold. This threshold selection has to be a function of the intensity range of the pixels in the image. The average of the minimum and maximum threshold values is normally enough to optimize the conversion.

2. Extracting Connected Components

To begin the segmentation, we have to extract out each of the connected components by finding black pixels and then using a search algorithm to find all the pixels which are connected with them. This process is carried out until every black pixel in the image is being classified as part of a connected component. If any of the captured digits is broken/ defaced, then any of the connected components could represent a single character.

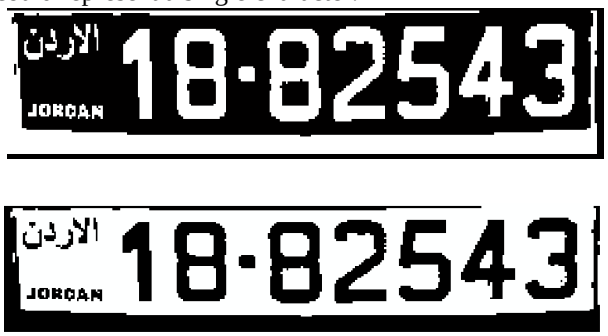


Fig 14: Converting black & white

3 Taking Horizontal and Vertical Projections for Each Number

We then take the horizontal and vertical projections (number of pixels versus its position) for each number to identify it. We take every number after filling it to increase the properties to describe any number.

Fig 15: Segmentation by projection

ISSN: 2277-3754
ISO 9001:2008 Certified
International Journal of Engineering and Innovative Technology (IJEIT)
Volume 4, Issue 4, October 2014

4. Segmenting the Numbers by Taking the Boundary Box

Now we have an image that consists of only the numbers that we want. So, we segment the number by taking the boundary box for every one as shown below:

Fig 16: Final stage of segmentation

D. Numbers Recognition

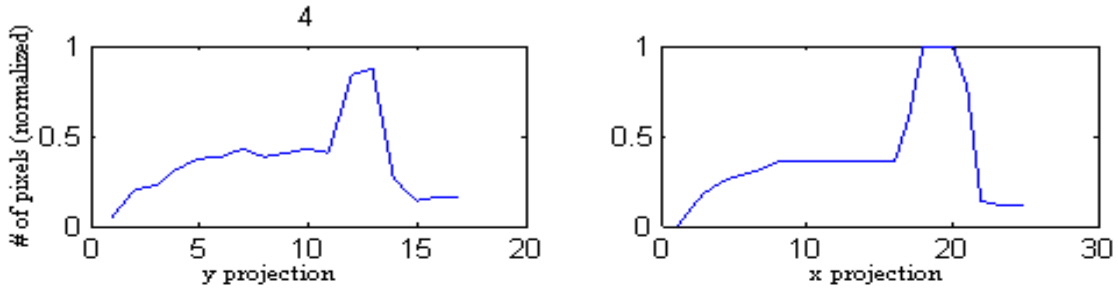
1. Normalization

In this phase, the extracted characters are resized to fit the characters into a window. In our practical work, each character is normalized to the size of 50x30 binary image then reshaped to standard dimensions before sending the data set to a neural network for training. It is very important to expnad the training database size for neural network because of the fact that increasing the database size results in an improved efficiency and accuracy of the network.

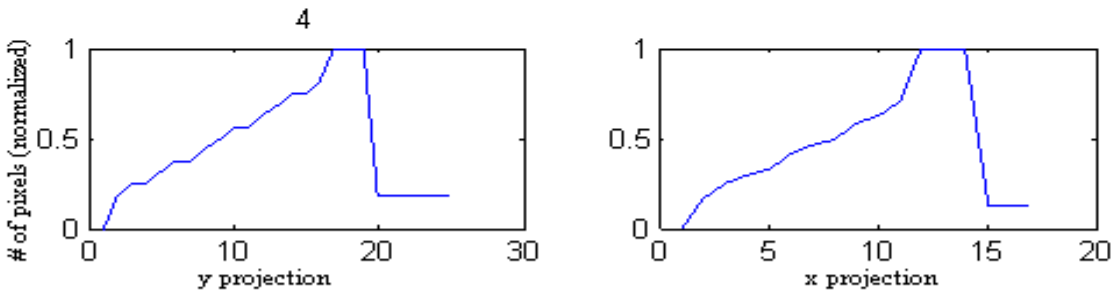
Segmented characters have very much variation in size. In this phase, all the characters are normalized to predefined height (vertical length) in pixels. As the characters always have variable width (horizontal length), each character image is normalized to a size of 32x32 using an image mapping technique.

2. Features Extraction

The goal of the feature vector is to define distinguishing features of the characters. Selecting the most relevant feature of each character cannot only facilitate data visualization and data understanding, but also reduce the measurement, storage requirements, training, and utilization time, particularly when the features are redundant. Initially, the centroid of the character image is determined. With respect to centroid, the number of transitions along the axes, 0 to 1 and 1 to 0, up to the boundary of character are counted. Transitions are specified for axes with predetermined angles.

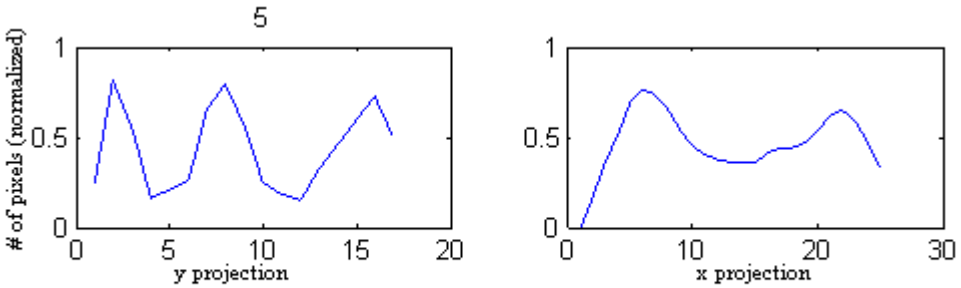


when number 4 is not filled

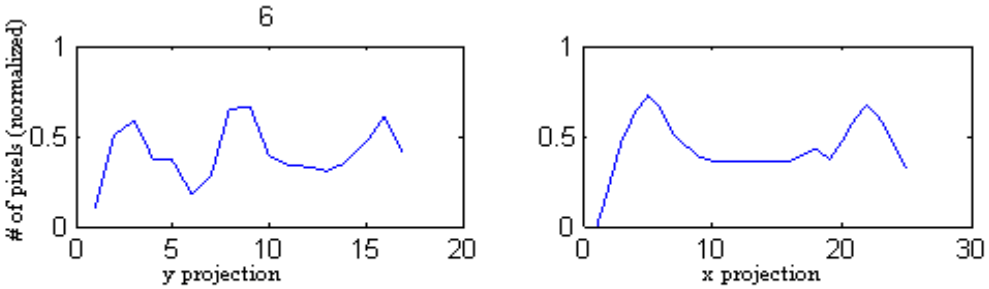


when number 4 filled

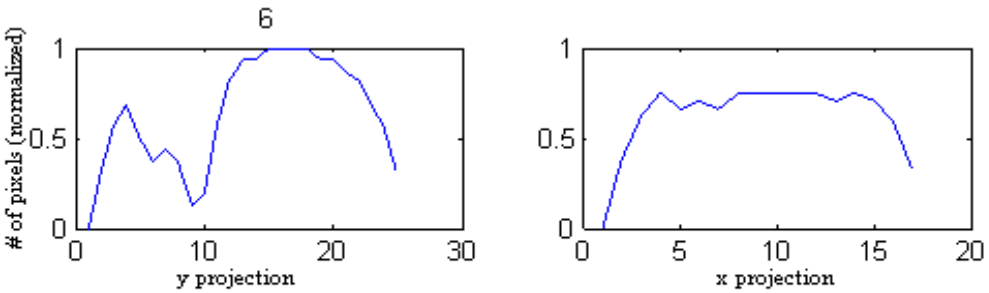
ISSN: 2277-3754
ISO 9001:2008 Certified
International Journal of Engineering and Innovative Technology (IJEIT)
Volume 4, Issue 4, October 2014



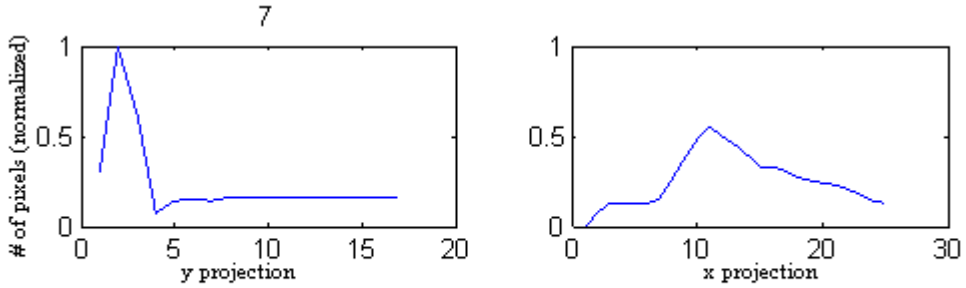
number 5



when number 6 is not filled



when number 6 is filled



number 7

ISSN: 2277-3754
ISO 9001:2008 Certified
International Journal of Engineering and Innovative Technology (IJEIT)
Volume 4, Issue 4, October 2014

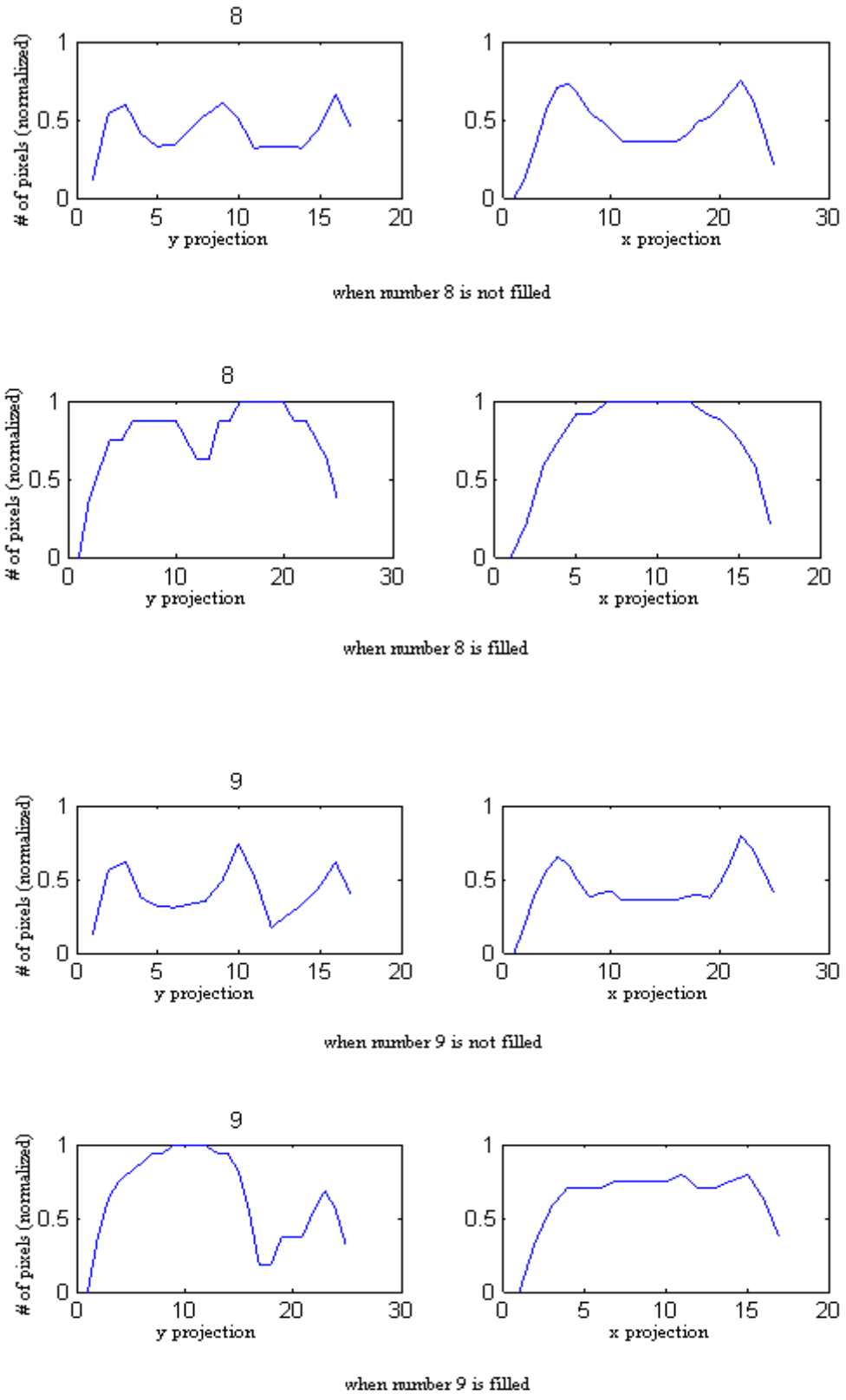


Fig 17: Curves of normalization

ISSN: 2277-3754
ISO 9001:2008 Certified
International Journal of Engineering and Innovative Technology (IJEIT)
Volume 4, Issue 4, October 2014

E. Database System

The database system is a collection of information or data which is being orderly organized, thus it can easily be accessed and updated. The database can be in the form of text, contents, and images.

A simple database system can be created, which contains the information of vehicles that are allowed to access a designated area for this access control application. The basic functions like "Add", and "Delete" should be also available to make it simple for the user to organize the database easily.

III. HARDWARE DESIGN AND IMPLEMENTATION

A. Electric Gates

Electric gates provide many benefits to the home owner, and they have become very popular in the recent years because of their low cost nowadays. There are two main types of gate operators: Slide and swing. Both types prevent access by a vehicle until the telephone entry system or the access control system enables entrance. Slide and swing gate operators come in a range of AC or DC powered models with battery and solar panel options.

B. IP Camera

IP stands for Internet Protocol, which is a protocol for transmitting data across a network. An IP Camera is a camera that plugs directly into a network router, and is not reliant on a PC to work. Data from the IP camera is transmitted through the network, and can be securely viewed at a remote location. An IP camera requires a high speed connection (such as DSL), a router, and an Ethernet cable. A computer is needed to view the data; however the camera works independently of the computer.

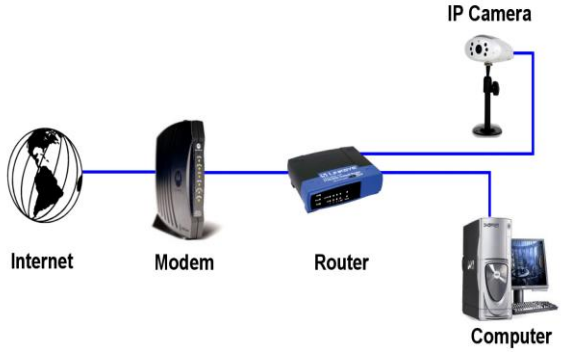


Fig 18: The connection of an IP camera

C. Motion Detectors

A motion detector is a piece of equipment that can be used alone or as part of a complete business or home security system. Infrared detectors, passive infrared detectors, and outdoor motion detectors are intended to sense any movement that occurs and notify the owner of the location when a motion is detected. Some motion sensors are small parts of a more complicated security system, while others are simple and basic, giving the owner of the location an audible sound during motion detection.

IV. SOFTWARE DESIGN AND SIMULATIONS

Matlab programming is used as it is a user friendly and the image processing can be easily performed via it. We also use the graphical user interface (GUI) to build up a graphical display to run the system. The GUI is a graphical display that contains devices or components which enable the user to perform interactive tasks.

A. Setting up the Graphical User Interface

1) A quick start of the GUI at the MATLAB: We are able to create either a blank GUI or open an existing GUI by typing "guide" in the Matlab command.

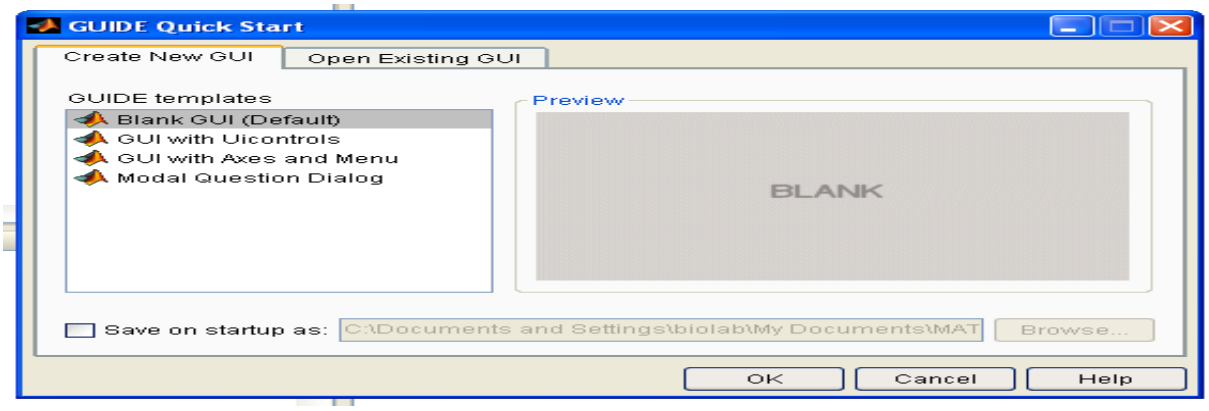


Fig 19: GUI quick start

ISSN: 2277-3754
ISO 9001:2008 Certified
International Journal of Engineering and Innovative Technology (IJEIT)
Volume 4, Issue 4, October 2014

2) After selecting the blank GUI, it will bring us into the images below. At the stages, users are able to insert whatever components that are wanted into the program. At the same time, users are able to determine how they are going to display the program.

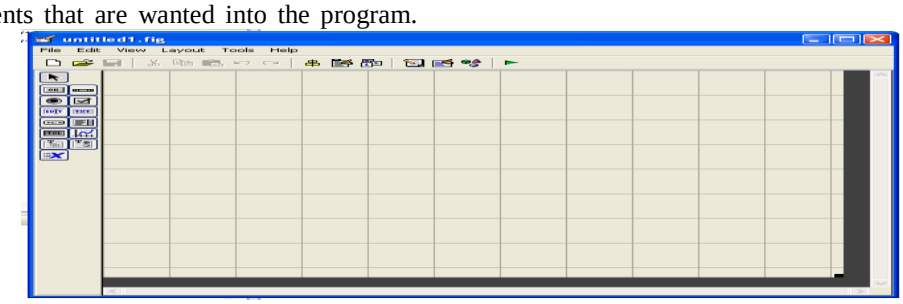


Fig 20: A blank GUI

3) This is the design output of the GUI:

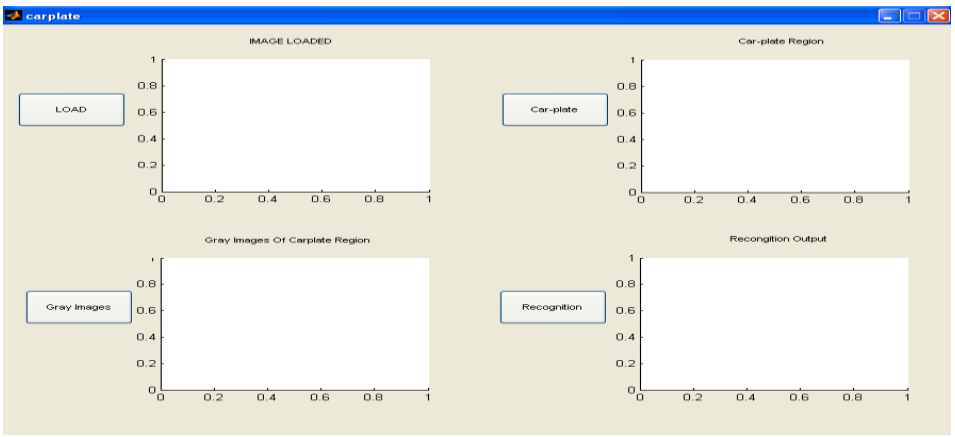


Fig 21: GUI prototype

4) After which, we program the individual pushbutton and will be able to display the individual function at the axis assigned to it. The below diagram shows the output of the GUI after programming:

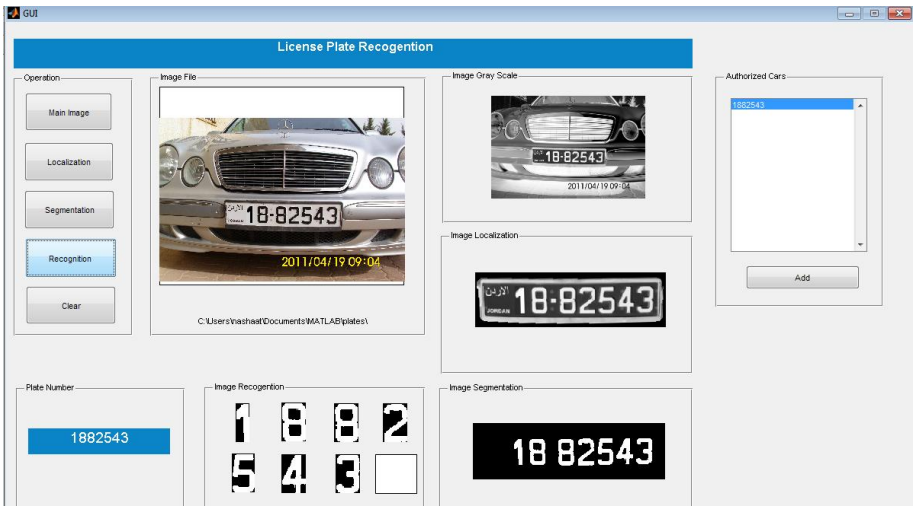


Fig 22: Final GUI

ISSN: 2277-3754
ISO 9001:2008 Certified
International Journal of Engineering and Innovative Technology (IJEIT)
Volume 4, Issue 4, October 2014

We now consider the three cases that may occur while examining our practical work:

Input	Description	System's Status	Result
Plate 1	The plate is found in the database	Available	Access is granted
Plate 2	The plate is not found in the database	Available	Access is denied
Plate 3	The plate is in the database, but it is not recognized	Available	No change

Table 1: Tests

We have three cases: The first case happens when the plate is recognized and also found in the data base (access is granted). The second case occurs when the plate is not found in the data base (access is denied). The third case takes place when the program cannot recognize the plate due to many factors such as whether, light intensity, and so on (no change). In our practical part, we have performed this test for about 100 vehicles and resulted in an average percentage of success of about 91%.

V. CONCLUSIONS AND FUTURE WORK

In this paper, we presented practical application software designed for the recognition of car license plates. Firstly, we extracted the plate location. We then separated the plate characters individually by the segmentation. Finally, we applied a template matching with the use of correlation for recognizing the plate characters. This system was successfully tested over a number of images.

The purpose of this paper was to investigate the possibility of making a system for automatic recognition of license plates to be used in parking access control system. We actually have proved that the development of the vehicle license plate recognition system is possible using Matlab.

The process of the vehicle number plate recognition requires a very high degree of accuracy. Our setup has been tested for 100 vehicles containing different numbers for different vehicle models, which resulted in an average percentage of success of about 91%.

As a future work, this system can be redesigned for multi-shape car license plates. Also, we may build a motion detection system for triggering the camera to capture a photo and send it to the software. Moreover, an interface between the software and the electric gate may be implemented as well to command the gate to open for a predetermined period.

REFERENCES

[1] Rafael C. Gonzalez, Richard E. Woods, and Steven L. Eddins, "Digital Image Processing Using MATLAB," Gates mark Publishing, 2nd edition, 2009.

[2] Rafael C. Gonzalez and Richard E. Woods, "Digital Image Processing," Pearson Education, Inc., 3rd edition, 2008.

[3] Sorin Draghici, "A Neural Network Based Artificial Vision System for License Plate Recognition," International Journal of Neural Systems, Vol. 8, No. 1, pp. 113-126, March 1997.