Automated Car Park Management System

JP Fabros¹, D Tabañaɡ², A Espra³, OJ Gerasta⁴

Mindanao-State University-Iligan Institute of Technology
Iligan City, Philippines

johnpatrick.fabros@g.msuiit.edu.ph, arceile.espra@g.msuiit.edu.ph,
darlyn.tabanag@g.msuiit.edu.ph, olgajoy.labajo@g.msuiit.edu.ph,

Abstract. This study aims to develop a prototype for an Automated Car Park Management System that will increase the quality of service of parking lots through the integration of a smart system that assists motorist in finding vacant parking slots [1]. The research was based on implementing an operating system and a monitoring system for parking system without the use of manpower. This will include Parking Guidance and Information System concept which will efficiently assist motorists and ensures the safety of the vehicles and the valuables inside the vehicle. For monitoring, Optical Character Recognition was employed to monitor and put into list all the cars entering the parking area. All parking events in this system are visible via MATLAB GUI which contain time-in, time-out, time consumed information and also the lot number where the car parks. To put into reality, this system has a payment method, and it comes via a coin slot operation to control the exit gate. The Automated Car Park Management System was successfully built by utilizing microcontrollers specifically one PIC18F4550 and two PIC16F84s and one PIC16F628A.

1. Introduction

Fully automated systems are being adopted in industries across the world at a rapid rate. Control systems are replacing manual operators and fully automated machines are replacing human labor. Less personnel and smarter machines means less operating and labor costs while increasing the quality of the products or services offered.

This study will increase the quality of service of parking lots by integrating a smart system which assists motorists in finding vacant parking slots [1]. It has a Parking Guidance and Information system concept which will efficiently assist motorists and ensure the safety of the vehicles and the valuables inside the vehicle.

In this parking system, a database lists the plate numbers of users and records the duration of their use of the parking lot. An Optical Character Recognition (OCR) system is also implemented to recognize the plate numbers of the users as they enter the parking lot [2,3]. Though this limits the use of the parking lot, it increases the security of the lot which has been deemed as a reasonable trade-off.

JP Fabros, D Tabañaɡ, A Espra, OJ Gerasta
are with EECE Dept. of Mindanao-State University-Iligan Institute of Technology
Iligan City, Philippines
2. Parking System

Figure 1 shows the whole system of the project which is composed of PIC micro, sensors, PC, stepper motor, LED, and coin slot. Figure 2 is the behavioral simulations of the main circuit board using Proteus.

Two seven-segment displays are situated at the front of the car park prototype to inform motorists of parking lot availability as shown in Figure 3. One display shows the total number of unoccupied parking slots in the car park prototype. The other shows the parking slot that is assigned to the motorist immediately at the front gate of the car park. This is to help motorists avoid wasting time in locating an available parking slot. To complement these displays, there will be two sets of LEDs designated for parking slot availability in the car park. When there is at least one parking lot that is available, the first set of LEDs will be lit. When all the parking slots have been occupied, LEDs
designated for the car park being full will be lit. These LEDs are viewable outside of the car park prototype to inform motorists of parking lot availability.

Sensors underneath each lot as shown in Figure 4 determine if the lot is available or not. If the sensor for that particular parking lot is activated, it implies parking is done in that lot. This lot is then taken off the list of recommended parking lots which is displayed at the seven-segment display above the entrance.

Two stepper motors serve to control the entrance and exit gates. The operation of each stepper motor is controlled by separate PIC16F84A microcontrollers. The entrance microcontroller is in turn be activated by the main microcontroller and the exit microcontroller will be controlled by the payment system microcontroller. Once the LDR sensors at the gates are activated, the main microcontroller or the payment system microcontroller will send a signal to the gate microcontrollers to open or close the appropriate gate. An LCD will also be placed at the front of the car park prototype that displays the time and name of the car park prototype.

![Figure 3 Parking Lot Prototype](image1)

![Figure 4. LDR Position](image2)

### 3. Coin Slot Payment System

The coin slot payment system is created as an independent block before being integrated into the prototype as shown in Figure 5.

![Figure 5 Coin slot Payment System Architecture](image3)

The payment system microcontroller manages the automation of payment and exit. The MATLAB GUI is responsible for observing the vehicles’ parking durations. When the program detects a vehicle leaving from its designated parking slot, the changes will be reflected in the GUI. The parking duration of that particular vehicle will be sent to the payment system microcontroller through the MAX232 interfacing. The microcontroller is deactivated until data is sent to it.
The parking charges are incremented by 5 at a fixed rate. The total charge incurred during the customer for his parking duration will be displayed on a dual seven-segment display. When the customer drops coins into the coin slot, the corresponding value of the coin will be deducted from the charge displayed in the dual seven-segment display. The microcontroller then sends a signal to the exit gate microcontroller signaling it to open. The stepper motor will then control the gate to open. An LDR sensor at the front of the exit gate will determine when the gate should be closed. When the LDR sensor is triggered by the exiting vehicle, the gate is then brought down. At the same time when the customer is able to pay his charges in full, the microcontroller is brought back to its deactivated state until another vehicle exits the car park.

4. Design of Management GUI in MATLAB

The MATLAB GUI will serve as the monitoring tool for the Automated Car Park System. It is used for convenient management and data logging of the events of the system. Also, this is used as plate number recognition function.

4.1 GUI as monitoring tool

The availability of the parking lot is shown along with the basic information of the motorist’s usage of that parking lot. The eight parking lots will be shown with their respective graphical axes as shown in Figure 6. These will show the screenshots of the vehicles front area which includes the plate number. Along with these screenshots will be the time-in, time consumed, and tentative parking fee of each vehicle.

![Figure 6 MATLAB GUI](image)

4.2 GUI as Plate Recognition

![Figure 7 Plate Number Recognition Function](image)
Figure 7 screenshot shows the operation of the plate number recognition function. The first image is the input image. The second image is the image darkened and converted to black and white. The small area connected white components are removed from the image and what is left is most probably the plate number area. The third image shows the image after the small white parts are removed. The coordinates of this large white area is then extracted and saved using the values of the vertical and horizontal histograms. Lastly, only the plate number region is left. This is shown on the last image. This is then processed through segmentation and template matching.

The label above the image shows the result of the image processing. The extracted string is shown to be “GTR798” which coincides with the plate number image. Therefore the plate number recognition has some degree of accuracy.

4.3 Design of OCR Function in MATLAB

The system is developed as a complement to the automatic smart parking system prototype as shown in Figure 8. When the ASPS system detects an incoming car, a front view snapshot of the car will be taken. This image is sent to the main program to be processed by the image processing system.

For ease of use, the system has been created in MATLAB as a function named processplate.m. The Main Program will call for the function to process the webcam image. After processing the image, the output which is the plate number will then be available in string format. The string will be used for identification of individual cars and will be easily usable for the GUI of the Main Program.

4.4 MATLAB Algorithm

4.4.1 Grayscale/BW image
The input image is taken from the webcam, and this will first be in RGB color format. An RGB color image is understood by MATLAB as an m-by-n-by-3 array consisting of the individual values for the red, green, and blue components.
4.4.2 Find Largest White Area

After the image is darkened and converted into black and white, the image becomes an image of white blobs on a dark background. The most probable plate number region is the area that has the highest number of interconnected white pixels. The small white regions are removed from the image using the bwareaopen function:

\[
\text{testimage} = \text{bwareaopen(testimage, 15000)};
\]

We are left with a patch of white that has the highest area. The row and column coordinates of this region is determined using a scanning algorithm. Each pixel is scanned by row until the logical value 1 is found which represents the white region.

4.4.3 Histogram Analysis

Histograms representing the sum of the differences of neighboring pixels column-wise and row-wise will be used to indicate the number of transitions from black to white and vice versa along a row or column as shown in Figure 10. The distinguishing characteristic of a plate number is that it is a dark row of letters across a white background, that is, it represents quick transitions from light to dark to light. For an image with no other structure with this property, an area with a high sum of pixel differences is a very likely candidate for the number plate area.

First, the vertical histogram is created by running the algorithm through each row from end to end. The code starts at the leftmost value of a row and subtracts this with the value exactly next to it. The absolute value is taken and added to the sum of pixel differences for that row. This process repeats with the 2nd, 3rd, 4th value and so on until it reaches the second rightmost value.

This process is also repeated with the succeeding rows and the final histogram will consist of the pixel differences of each row. To get the horizontal histogram, the same process used to get the vertical histogram need only be changed to consider the columns instead of rows. Both horizontal and vertical histogram length will have equal number of rows and columns respectively.

![Figure 10 Sum of differences histogram](image)

4.4.4 Plate Extraction

After getting the histograms, the region indicated by the histograms to have the highest number of transitions from white to black to white is most probably the number plate region. The coordinates of this region is saved. From a copy of the original and unedited black and white image, the image found on the coordinates indicated by the histogram will be extracted.

4.4.5 Segmentation

After the plate region has been extracted, we are left with an image of consecutive letters. An essential step in character recognition is separating these letters from one another as shown in Figure 11b. The algorithm used to separate the letters from one another assumes that the plate number is horizontal. With this, the algorithm traverses through each column of the image and looks for black pixels which are the letters themselves. When a black pixel is recognized in one column, it signals the starting edge
of a letter. The algorithm continues until it reaches a column in which there are no more black pixels. This signals the rightmost edge of that letter and this letter is saved separately. The process continues until the last column is reached and all the letters saved separately. This process is then repeated for the separate letters, but this time the code is run vertically.

(a) G  (b) GTR798

**Figure 11** (a) Horizontal Scan, (b) Segmented Letters

### 4.4.6 Template Matching

The final step in character recognition involves recognizing the individual segmented letters. A template matching technique is employed to recognize these characters. Since template matching involves comparing a test image with a template, a database of templates has been created. Every letter of the alphabet and the numbers 0 to 9 each have their own template. Each segmented letter then of the plate number is scaled to equate the templates and compared one by one to the templates. The comparison with the highest match is considered to be the equivalent character of the image. To determine how good a match between the test image and the template is, a correlation function is used. The algorithm uses the normalized cross correlation of the matrices of the test image and the template to determine the accuracy of the match. The process is done for each of the segmented letters and the results are combined to acquire the string format of the plate number.

(a) F  (b) F

**Figure 12** (a) Character Template, (b) Segmented Letter

### 5.0 Summary

The practicality of the design is evident in that it lessens the stress of the motorists in finding a parking lot. It acts the way it is supposed to and shows motorists even from afar if the car park has vacant parking lots or not. It then recommends to the entering vehicle which parking lot is vacant. Since the whole system is automated, the need for manual operators is lessened, ergo; the labor cost is also lessened.

The automated car park prototype was successfully built by utilizing microcontrollers specifically the one PIC18F4550, two PIC16F84s and one PIC16F628A. The main microcontroller which is the PIC18F4550 successfully controls functions of the whole prototype. The prototype is put through a series of tests to identify the efficiency of the system. Despite its limitations, the prototype does the jobs it was meant to do excellently.

### References

[1] M. M. Rashid, A. Musa, M. Ataur Rahman, and N. Farahana, A. Farhana, M. M. Rashid, A. Musa, M. Ataur Rahman, and N. Farahana, A. Farhana, “Automatic Parking Management System and Parking Fee Collection Based on Number Plate Recognition”, International Journal of Machine Learning and Computing, Vol. 2, No. 2, April 2012

[2] Sulaiman, N., Mustafa, M., Hawari, K., “Development of Automatic Vehicle Plate Detection System”, in proc. Conf. System Engineering and Technology (ICSET), 2013 IEEE 3rd International Conference.

[3] Ayer Keroh ; Idris, A., “Study of Car Park Control System Using Optical Character Recognition”, Computer. Vision & Robot. (CoVisBot) Lab., Tech. Univ. of Malaysia Melaka.