Presented to the Department of Electronics and Computer Engineering
De La Salle University - Manila
Term 2, A.Y. 2022-2023

In partial fulfillment
of the course Signals, Spectra, and Signal Processing Laboratory
In Course LBYEC4A-EK2

Final Project:
"Recreating an Automatic Number Plate Recognition System (ANPR) through MATLAB"

CHUA, Carl R.
DEVERA, Nathan H.
OBCIANA, Michael Angelo A.

Submitted to:
Ramon Stephen L. Ruiz

April 11, 2023

I. ABSTRACT

As society in the Philippines depend more on road transportation in order to go from one place to another, road officials are in ever need to supervise road safety and road management, thus Automatic Number Plate Recognition (ANPR) Systems are used to easily detect vehicles for tracking and databasing. In this project, the proponents plan on using the programming language known as MATLAB that will read and detect a plate number of a given car or motor vehicle, which consists of a combination of letters and numbers and will store it in its own personal database; thus creating an ANPR System. In this program, it will consist of the following features: Plate Number Detection and Removal of Background Images for clarity. After the successful completion of the system, an additional feature was added wherein the ANPR system would also detect the last digit of the plate number in order to determine if the vehicle would follow the Number Coding Rule for the day.

II. INTRODUCTION

Human transportation is considered to be one of the most important yet underestimated necessities in a persons' daily life. Whether it be from the comfort of their own private vehicle, or sharing with other strangers in a public transport, one thing's for certain, it is necessary for these vehicles to be held accountable to the lives of the drivers, passengers, other drivers, and especially pedestrians. In accountability, each vehicle must be publicly registered in a database for officials to monitor the said vehicle. To do this, each vehicle must be given a plate number, which becomes the vehicle's form of identification, with the countless possible combination that the Land Transportation Office has standardized in recent years, in which the format for the plates of motor vehicles would be a three letter then four number combination [1]. Briefly mentioned earlier these plate numbers must be stored in a database for officials from the Land Transportation Office (LTO) to be observed, this system of detecting and storing said plate number is called an Automatic Number Plate Recognition (ANPR) System.

To put it briefly, ANPR systems is a technology that utilizes optical character recognition (OCR) on vehicle registration plate images in order to recognize and to read the vehicle's registration number. This system uses several image processing techniques that will allow the user to identify vehicles automatically and quickly in real-time video or images captured via camera [2].

Figure 1.1: Representation of a Working ANPR System [2]

These systems are becoming more important as the years progress in order to monitor public safety, enhance law enforcement, and improve traffic management. Regarding the first two factors, ANPR systems would support law enforcement in improving their ability to enforce and investigate, gather more substantial and precise data toward the vehicle, and speed up the tedious process of matching license plate numbers to lists of wanted, stolen, or other vehicles of interest [3]. In terms of traffic management, ANPR technology offers the possibility of collecting precise traffic data from different locations across a road network, the data gathered would then be able to generate detailed insights into traffic patterns (traffic volumes, average traffic speeds, and dependable origin-destination data). It can also provide valuable solutions in constructing and adjusting traffic models, which can hopefully decrease traffic congestion, which is an ever-amounting problem in Metro Manila [4].

For the project, the students plan to create a fixed ANPR system that focuses mainly on plate number detection, plate number enhancement through the use of filters and background removal, and reading the said license plate. All of these factors would be be implemented within the programming language known as MATLAB, allowing the proponents to use their understanding and knowledge in image processing by recreating the said system, while also making it more efficient.

III. THEORETICAL CONSIDERATION

In the creation of the program, specific functions were used that either applied the necessary applications of image processing suitable in creating an ANPR system or supported the said system. Listed below are these said functions:

Loading Image

- ***imread*** - reads an image from the file specified by its filename; the images' format must also be included as well [5].

- ***imshow*** - displays the image stored specified by its filename [6].

Changing the Original Image Color to Morphological

- ***rgb2gray*** **-** converts the original image to its grayscale form, this is done by removing the images' hue and saturation while retaining its luminosity [7].
- ***imbinarize*** - generates a binary image from either a two or three-dimensional grayscaled image by setting the values of a determined threshold to 1's while the remainig data bits will have a value of 0 [8].
- ***strel(****'disk', r)* **-** represents a flat morphological structure, in this case, representing a shape of a disk wherein the value *r* would be the value of the radius it encompasses [9].
- ***imclose*** **-** performs a morphological closing on either a grayscaled or binary image. Essentially takes away the unwanted parts of the image while preserving the important information (in this case, the license plate) intact [10].

Scanning and Displaying Filtered Text

- ***ocr*** - returns an object that contains information regarding to optical character recognition (OCR) from the filtered image or the region of interest (ROI) [11].
- ***strrep*** - replaces occurrences from the old text and changes it into the new specified text. In context of this project, it removes any unwanted spaces in order to generate a clearer license plate [12].
- ***strtrim*** **-** removes any possible leading or trailing whitespace characters from the recognized string.

IV. METHODOLOGY (application in sigdisc)

For an ANPR system to function properly, this system must encompass most, if not all, of these image/video processing applications. An ideal ANPR system would consist of the following:

- Video Input & Image Acquisition
- Plate Number Detection & Cropping
- Plate Number Extraction & Reading
- Using of Number Plate Information

First would be video input and image acquisition. The ANPR camera captures either an image or video that includes one or more license plates. Frequently, during nighttime as the visibility of the camera would be greatly hindered, infrared illumination is integrated to enable the cameras to still recognize the said plate numbers. Next would be plate number detection and cropping. With the aid of machine learning, the system

would be able to identify the license plate itself in the image/video. One common practice of this application is locating the license plate through its bounding box, which is accomplished by identifying the areas of contrast between the plate and the background. After locating the license plate, it is then cropped and normalized. Proceeding this would be extracting and reading the license plate. OCR software is then utilized to extract the said plate number in text format from the identified plate area. Further, developing said software can allow the system to identify the country based on the given plate number. Lastly it can be used for additional plate information. After transforming it into plain text format, the plate number is stored in a database which can determine if the said vehicle is authorized or not [2].

V. RESULT

Table 5.1: Clear and Close

```
clc
clear all
close all
```

Table 5.2: Loading Image

```
pic = imread('Plate_Number_Sample_2.PNG');
imshow(pic);
INFO = imfinfo('Plate_Number_Sample_2.PNG');
title('Original Image');
```

Figure 5.1: Generated Image (Original)

Table 5.3: Changing the Original Image Color to Morphological

```
figure;
pic_gray = rgb2gray(pic);
pic_binary = imbinarize(pic_gray);
```

```
pic_flipBW = ~pic_binary;
g = strel('disk',5);
bw = imclose(pic_flipBW,g);
imshow(bw);
title('Morphological Image');
```

Figure 5.2: Generated Image (Morphological)

Table 5.4: Scanning and Displaying the Text Founded in the Plate Number

```
translation = ocr(bw,'TextLayout','Line','characterSet', ...
   'ABCDEFGHIJKLMNOPQRSTUVWXYZ01234567890')
str_ocr = translation.Text;
str_ocr_no_space = strrep(str_ocr,' ', '');          % Remove Spaces
str_ocr_trim = strtrim(str_ocr_no_space);      % Remove New Line (Enter)
```

```
translation =
  ocrText with properties:

                      Text: 'PHR 938↵↵'
    CharacterBoundingBoxes: [9×4 double]
      CharacterConfidences: [9×1 single]
                     Words: {2×1 cell}
        WordBoundingBoxes: [2×4 double]
          WordConfidences: [2×1 single]
```

Figure 5.3: Generated Output from the Code (variable ***translation***)

Table 5.5: Detecting Last Digit

```
Plate_Number = str_ocr_trim                % Obtaining the string.
Last_Digit=Plate_Number(end)               % Get the last digit.
                                           % 4 = position of a character
                                           % in a string.
```

```
Plate_Number = 'PHR938'

Last_Digit = '8'
```

Figure 5.4: Generated Output from the Code (variable ***Plate_Number*** & ***Last_Digit***)

Table 5.6: Number Coding System

```
t = datetime;
DAY = day(t,'name')

strMo = 'Monday';
strTu = 'Tuesday';
strWe = 'Wednesday';
strTh = 'Thursday';
strFr = 'Friday';
strSa = 'Saturday';
StrSu = 'Sunday';

LD = Last_Digit;

if contains(strMo,DAY) || contains(strWe,DAY)
   if contains(LD,'1') || contains(LD,'2')
      violation_MW = 'Has VIOLATED the Number Coding Rule'
   else
      followed_MW = 'Has followed the Number Coding Rule'
   end
elseif contains(strMo,DAY)||contains(strTh,DAY)
   if contains(LD,'3') || contains(LD,'4')
      violation_MTh = 'Has VIOLATED the Number Coding Rile'
   else
      followed_MTh = 'Has followed the Number Coding Rule'
   end
elseif contains(strTu,DAY) || contains(strTh,DAY)
   if contains(LD,'5') || contains(LD,'6')
      violation_TuTh = 'Has VIOLATED the Number Coding Rule'
   else
      followed_Tuth = 'Has followed the Number Coding Rule'
   end
elseif contains(strTu,DAY) || contains(strFr,DAY)
   if contains(LD,'7') || contains(LD,'8')
      violation_TuFr = 'Has Violated the Number Coding Rule'
   else
      followed_TuFr = 'Has followed the Number Coding Rule'
```

```
   end
elseif contains (strWe,DAY) || contains (strFr,DAY)
   if contains(LD,'9') || contains(LD,'0')
      violation_WeFr = 'Has violated the Number Coding Rule'
   else
      followed_WeFr = 'Has followed the Number Coding Rule'
   end
end
```

```
DAY = 1×1 cell array
    {'Monday'}

followed_MW = 'Has followed the Number Coding Rule'
```

Figure 5.5: Generated Output from the Code (variable ***DAY*** & ***followed_MW***)

VI. DISCUSSION

The objective for this project is for the MATLAB program to be able to detect the plate number of a certain vehicle, and decide whether the vehicle follows the rules and regulations of the number coding scheme implemented by the Metro Manila Development Authority (MMDA). Given that there is a picture of the plate number, it must be in black and white, in such a way that the texts are colored white. Afterwards, the OCR (Optical Character Recognition) reads the text. It should be limited to detecting alphabets and letters only. Furthermore, the string created by the OCR should be extracted to get rid of spaces and newlines. However, there are limitations of this MATLAB code such as the OCR may not be able to read the texts of the plate number properly if the image includes the back of the car. There are some texts from the plate number that are not read by the OCR properly. This is due to a certain font that is not familiar with the OCR. Hence, it is more likely that machine learning must be taken into consideration when building the code. Furthermore, the given pictures are not taken in an angle, and they are near to the camera. Such additional parameters needed more advanced programming such as machine learning. Fortunately, this code was able to demonstrate the basics for text detection of the plate number. Moreover, the detection of text can be integrated to another system such that an additional code can be added as a decision whether the last digit of the plate number of a certain vehicle violated the rule or not with the manipulation of the if then statements. It is suggested that those who want to develop a more sophisticated plate detection system must consider as many significant parameters as possible and study a more advanced course in programming to improve the accuracy of the system.

VII. CONCLUSION

During the coding completion and its programming process, the students were able to use their knowledge and understanding of the lessons learned through the laboratory by creating a functional ANPR system. The methodologies mentioned earlier were adequately met as the program was able to read and acquire the said image, focusing only on the vehicles plate number by first detecting its location, disregarding its background image to only attain the license plate, extracting the plate number itself through the process of digital filtering, and lastly using the extracted information to determine whether or not the vehicle is suitable to drive on the given day by the implementation of a number coding system. All of these applications on image processing as an umbrella term were all done through the programming language known as MATLAB, with the help of its database of functions.

VIII. AUTHOR'S CONTRIBUTION

Abstract -
Introduction - Nathan Devera
Theoretical Consideration - Nathan Devera
Methodology - Michael Angelo Obciana
Result - Carl Chua
Discussion - Carl Chua
Conclusion - Nathan Devera

IX. REFERENCES

[1] "DOTC: New LTO license plates available by September 2013," *The Summit Express*, Aug. 15, 2013. https://www.thesummitexpress.com/2013/08/dotc-new-lto-license-plates-available-by-september-2013.html

[2] N. Klingler, "Automatic Number Plate Recognition (ANPR) - Overview," *viso.ai*, Jun. 26, 2021. https://viso.ai/computer-vision/automatic-number-plate-recognition-anpr/

[3] "Automated License Plate Recognition," *Theiacp.org*, 2020. https://www.theiacp.org/projects/automated-license-plate-recognition

[4] A. Robinson and A. Van Niekerk, *Uses of ANPR data in traffic management and transport modelling*. 2014. Accessed: Apr. 12, 2023. [Online]. Available: https://repository.up.ac.za/handle/2263/45559

[5] “Read and Display Image,” *Mathworks.com*, 2009. https://www.mathworks.com/help/matlab/ref/imread.html

[6] “Display Grayscale, RGB, Indexed, or Binary Image,” *Mathworks.com*, 2016. https://www.mathworks.com/help/images/ref/imshow.html

[7] “Convert RGB image or colormap to grayscale - MATLAB rgb2gray,” *www.mathworks.com*. https://www.mathworks.com/help/matlab/ref/rgb2gray.html

[8] “Binarize 2-D grayscale image or 3-D volume by thresholding - MATLAB imbinarize,” *www.mathworks.com*. https://www.mathworks.com/help/images/ref/imbinarize.html

[9] “Morphological structuring element - MATLAB,” *www.mathworks.com*. https://www.mathworks.com/help/images/ref/strel.html

[10] “Morphologically close image - MATLAB imclose,” *www.mathworks.com*. https://www.mathworks.com/help/images/ref/imclose.html

[11] “Recognize text using optical character recognition - MATLAB ocr,” *www.mathworks.com*. https://www.mathworks.com/help/vision/ref/ocr.html

[12] “Find and replace substrings - MATLAB strrep,” *www.mathworks.com*. https://www.mathworks.com/help/matlab/ref/strrep.html (accessed Apr. 12, 2023).