**Indonesian Journal of Electrical Engineering and Computer Science**
Vol. 5, No. 1, January 2017, pp. 99 ~ 108
DOI: 10.11591/ijeecs.v5.i1.pp 99-108

# Design of Automatic Number Plate Recognition on Android Smartphone Platform

**Teddy Surya Gunawan*[1], Abdul Mutholib[2], Mira Kartiwi[3]**
[1,2]Department of Electrical and Computer Engineering, Kulliyyah of Engineering
[3]Department of Information Systems, Kulliyyah of ICT, International Islamic University Malaysia
Jalan Gombak, 53100 Kuala Lumpur, (+603) 6196 4521
*Corresponding author, e-mail: tsgunawan@gmail.com

***Abstract***

*Automatic Number Plate Recognition (ANPR) is an intelligent system which has the capability to recognize the character on vehicle number plate. It is a combination of hardware and software designed to offer the optimum reliability. Since the past decades, many researchers have been proposed to recognize the vehicle number plate and implemented it in various access control, law enforcement and security, including parking management system, toll gate access, border access, tracking of stolen vehicles and traffic violations (speed trap, illegal parking, etc). However, previous researches implemented ANPR system on personal computer (PC) with high resolution camera and high computational capability. On the other hand, not many researches have been conducted on the design of ANPR in Android smartphone platform which has limited camera resolution and limited computational power.The main challenges of implementation ANPR algorithm on smartphone are higher coding efficiency, lower computational complexity, and higher the scalability. The objectives of this research is to design ANPR on Android smartphone, including graphical user interface (GUI) design, process design, and database design. First, a comprehensive survey on the pre-processing, segmentation, and optical character recognition is conducted. Secondly, proposed system development and algorithm implementation is explained in more details. Results show that our proposed design can be implemented effectively in Android smartphone platform.*

***Keywords***: *ANPR, GUI design, database design, OCR, Android*

## 1. Introduction

In the current innovative technology era, the significant tool for traffic law enforcement, access control, automated tolls and automated parking is needed. The automatic number plate recognition (ANPR) considerably can be used for those reasons. As an intelligent tool, ANPR has the ability to detect and recognize the vehicle’s number plate and provide the information regarding the vehicle properties.The ANPR is an image processing technique to extract the image of license plate on vehicle taken by camera or taken by either a color, a grayscale camera or an infrared camera and identified the vehicles by their number plate through optical character recognition (OCR) [1, 2].

The ANPR system recognizes characters on license plate through the combination of various techniques and algorithms, namely image pre-processing, object detection, and character recognition. It consists of a camera to detect the number plate object and processing unit to process and extract the characters and interpret the pixels into numerically readable characters [3]. It became much exciting in the last decade along with the improvement of digital camera technology and the computational processing [4]. Nowadays, the ANPR system has been used in traffic law enforcement, including speed trap, stolen car detection and border monitoring. It can be used also for the building management, such as parking lots and gate control [5].

Many researchers have proposed and published various methods and algorithms on ANPR. ANPR system commonly deployed by one of two basic methods: one accomplished the entire process real-time at the track location and the other transmits all the images from many paths to a remote computer location then performs the OCR process at some later point in time. For the process at the track location, it requires high resolution camera to capture the number

*Received September 12, 2016; Revised December 20, 2016; Accepted December 11, 2016*

plate image. In the transmitting and processing at server, it needs a reliable network and a large number of PCs to handle high workload, high bandwidth connection and processing the number plate image.

Although many researches on ANPR has been conducted, but not many researches are focused on the smartphone platform. In this paper, ANPR algorithm is designed and implemented in the low resolution camera and low processing power of smartphone. The following sections will discuss our design and implementation in more details.

## 2. Automatic Number Plate Recognition

This sectionwill provide a comprehensive literature reviewon ANPR algorithms, previous researches conducted so far in this area and techniques employed in the development of an ANPR system. The comparisons of available algorithms of ANPR along with the idea of ANPR on Android smartphone will be presented.

Automatic Number Plate Recognition (ANPR) is acquainted with many other terms, namely Automatic Vehicle Identification (AVI), Automatic License Plate Recognition (ALPR), Car Plate Recognition (CPR) and License Plate Recognition (LPR). The whole terms are referring to the recognition system using the camera to read the character on vehicle number plate. In this research, the ANPR term was used due to its popularity.

UK's Police Scientific Development Branch was noted as the inventor of ANPR system in 1976. The ANPR system has the proficiency to extract and recognize a license plate's characters from an image. It consists of a camera to capture a number plate, detect the characters' location in the image and excerpt the characters to deduce the pixels into numerically readable characters [3].

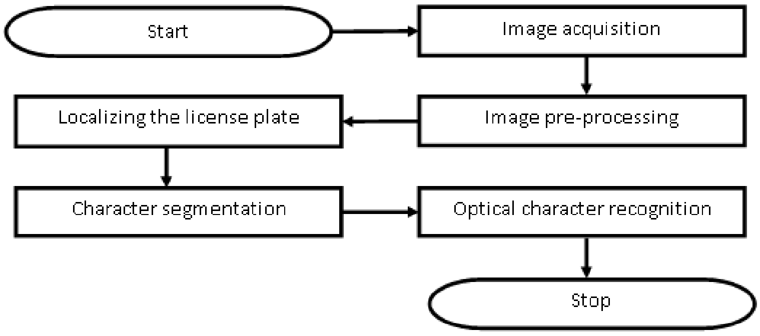

Figure 1. General Flow Chart of ANPR System

The ANPR system, as seen in Figure 1, generally can be composed of the following five stages [6]. The first stage is image acquisition which capture the image of license plate using a camera by considering the camera resolution, orientation, shutter speed and lighting condition. The second stage is image pre-processing, such as normalization, brightness and contrast adjustment, and skewness correction of the captured image. The third stage is localizing the license plate to extract the license plate from the whole vehicle image based on some features, such as the boundary, the color, or the presence of the characters. The fourth stage is character segmentation to segment the characters on the license plate by locating and identifying the individual character on the license plate image. The final stage is optical character recognition to recognize the segmented characters by pattern matching or classifiers, such as artificial neural networks (ANN), fuzzy logic, Hidden Markov Model (HMM), and template matching [7]. The overall performance of an ANPR system is depending on the performance of each individual stage.

The output of ANPR system, i.e. recognized plate number, can be connected to other applications or databases. ANPR used a sequence of image manipulation methods to detect, normalize and improve a license plate image and optical character recognition (OCR) to recognize the alphanumeric of license plate. During the last decade, it became much interesting alongside due to the enhancement of digital camera technology and computational processing [4]. It also can be used to detect and prevent criminal activities and parking control on a restiction parking area [8].

### 2.1. Image Acquisition

Image acquisition is an important stage to the ANPR system because it provides the input data for the subsequent processes. There are numerous ways to acquire the license plate’s images. The existing literature discusses the different image acquisition approaches used by various researchers. For example, a digital camera with or without embedded infrared lighting have been used to acquire license plate in [2, 9]. Kim et al [10] has utilized video camera to acquire the license plate image. Another method is using an image acquisition card as use in [11] to convert video signals into digital images based on some hardware-based image. In summary, the license plate image could be captured either by digital camera or video camera.

### 2.2. Image Pre-Processing

The goal of this step is to improve image quality captured previously. In [12], the acquired color image was converted to gray scale, and non-linear medium filter was used to enhance the image. Sobel filters and histogram equalization were used in [13], while power law transformation was used in [14]. It can be summarized that the choice of pre-processing algorithms depends on the complexity and the level of noise of the acquired image. Due to low processing power of smartphone, we need to carefully select the preprocessing algorithm so as not to burden the processor but producing a reasonable enhanced image.

### 2.3. Localization of the License Plate

Localization of the license plate is the process to detect the rectangular area of the number plate in a captured image. In human descriptions, a number plate is a small plastic or metal plate pasted to a vehicle for official identification purposes. However, the machines do not understand well this description. Therefore, we need to find an alternative description of a number plate based on descriptors that will be understandable for machines. In [10], two neural-network based filters and post processor were used with high reliability but it has rather complex computationally. Hough transform and contour algorithm were used in [13], while Sobel vertical edge detection, threshold and vertical projection were utilized in [2]. In our design, the license plate image was captured by smartphone, in which a relatively proper image could be obtained by using rectangular guide during capturing. Hence, localization algorithm requirement is minimized.

### 2.4. Character Segmentation

Many different methods of character segmentation have been proposed in the literature and. In [12], connected-component method which is simple, straight forward, and robuts was used. However, if there are joined or broke characters, the segmentation might fail. Horizontal projection and vertical projection were employed in [13], in which the algorithm can handle character with some rotation. In [9], thin window scanning 56x1 pixels was utilized which is straightforward and fast, but is more susceptible to the noise.

### 2.5. Optical Character Recognition

Optical character recognition (OCR) is the process to convert the images of handwritten or typewritten into machine encoded text. In previous researches, there are numerous methods such as simple Euclidean distance [15], Hidden Markov Model (HMM) [13], Artificial Neural Network (ANN) [12, 16, 17], Support Vector Machine (SVM)[18] and template matching [7, 19]. Table 1 shows the various algorithms for OCR along with its strengths and weaknesses. The Tesseract OCR library [20] was used as it can be integrated into Android SDK as well as it can provide two methods, including ANN and template matching.

Table 1. Various Algorithms for Optical Character Recognition

| Method | Strength | Weakness | Reference |
|---|---|---|---|
| Euclidean distance | Straight forward by taking the distance as an indication for the amount of distortion with respect to a certain prototype character. Accuracy 87% of overall process. | Need huge and representative training set | [15] |
| Markov model (HMM) | Can deal with multiple plate in the same image. The OCR accuracy 97. 52% | Require large seed, slow and need much memory and processing time. | [13] |
| Artificial Neural Network (ANN) | Can extract striking features, robust to any distortion, fast recognition, the accuracy about 97.30%; 98.79%; and 93.2% in sequent. | Feature extraction takes time, non-robust features will degrade the recognition | [16], [17], [12] |
| Support Vector Machine | Average time recognition 91 ms. Overall time 284 ms. Recognition accuracy 97.88% and Overall process 93.54%. | Cannot precisely detect the frame, misclassifications occur when image lower than 40×80 pixels | [18] |
| Template Matching | Simple and straightforward, accuracy 98.8% and 92% in the order. | The non-important pixels also be processed, weak to noise, rotation, thickness change | [7], [19] |

## 3. Android Platform

Android is a software environment made for mobile devices such as smartphone and tablet. It has to be stated that Android is not a hardware platform. Android consists of a Linux kernel-based OS, a rich UI, end-user applications, code libraries, application frameworks, multimedia support, and much more. Android was founded by Open Handset Alliance (group of over 30 companies led by Google) in July 2005. In the next sections, Leptonica and Tesseract libraries will be discussed.

### 3.1. Leptonica

Leptonica is an open source C library which is useful for efficient image processing and image analysis applications [21]. The library is developed since 2001 by a Google employee, Dan Bloomberg, and it is licensed under a Creative Commons Attribution 3.0 United States License. This library can be downloaded from http://www.leptonica.org.The library supported following operations, including rasterops, binary and grayscale morphology, convolution, scaling, rotation, affine transformations, seed filling and connected components, and various image enhancement algorithms.

### 3.2. Tesseract

Based on the information from the paper written by Ray Smith [20], Tesseract was developed between 1985 and 1994 at Hewlett Packard Laboratories Bristol and Hewlett Packard Co, Greeley Colorado. It appeared for the first time in 1995 at University of Nevada Las Vegas (UNLV) Annual Test of OCR Accuracy. It was a PhD research project conducted in HP Labs, Bristol, and it had the momentum as a possible software and hardware add-on for the HP's line of flatbed scanners. After ten years, due to lacking any further development, HP released the Tesseract as open source in late 2005. Currently, Tesseract library is available at https://github.com/tesseract-ocr. Tesseract is suitable for use as the backend and can be utilized for more complicated OCR tasks.

## 4. Proposed Design of ANPR System on Android Smartphone

To implement ANPR algorithm on Android smartphone, it requires system development environment and system design. System development environment used as the development tools which consist of hardware and software. In this section, hardware and software used for the development are presented. The system design explains the prototyping design of all ANPR subsystems including graphical user interfaces (GUI) design, process design and database design. It provides advantages for the developer as guidance to develop and implement the ANPR system on any Android platform.

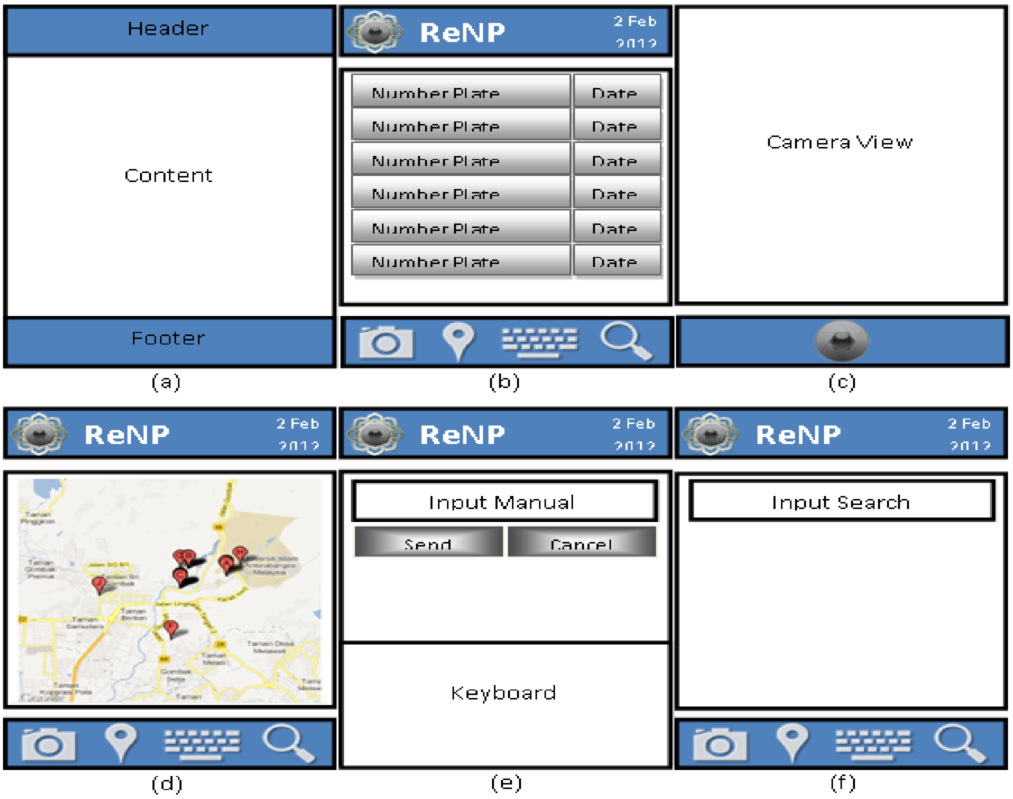

Figure 2. GUI, (a) Master layout of ANPR system, (b) Main page layout, (c) Capture page layout, (d) Map page layout, (e) Input manual page layout (f) Search page layout

### 4.1. Hardware and Software Environment Development

The Android smartphone used in this research is Samsung GT-S5830 code name Galaxy Ace also known as Galaxy Cooper which is mid-end smart phone family released by Samsung. It has 800 MHz 1 Core ARM v6 of processor with 278.34 MB of RAM. The screen resolution is HVGA (480 x 320) ~165 ppi pixel density and 5 mega pixels of camera resolution.

The ANPR system is constructed using Java Development Kit (JDK), Eclipse IDE (Integrated Development Environment), Android SDK (Software Development Kit) and Android Developer Tool (ADT), i.e. an Eclipse plug-in. The minimum Android API (Application Programming Interface) used for the constructing of the ANPR system is Android API 16 to develop application for Android 4.1 version (Jelly Bean).

### 4.2. System Design

In order to develop any application or system, it needs the guidance what work the developer must do to deliver the good application. The design leads the developer to the creation of various representation of the system which serve as a guide for the construction activity [22]. In this research the system design will be divided into user interface design, process design and database design.

### 4.2.1. Graphical User Interface Design

User interface design helps the developer to the concern on how the system presents the information to user. It is very important since user need the easiest way and simple to operate the system. Such as the navigation and the information’s presence need to show to the user. In the proposed system, the ANPR system only have five windows layout, as shown in Figure 2, including main window, capture window, location window, manual input window and search data window.

### 4.2.2. Process Design

The process design of the ANPR system in this research instead of having process inside the smartphone, there exists other processes outside the smartphone. Figure 3 shows the whole processes of the ANPR system in this research.

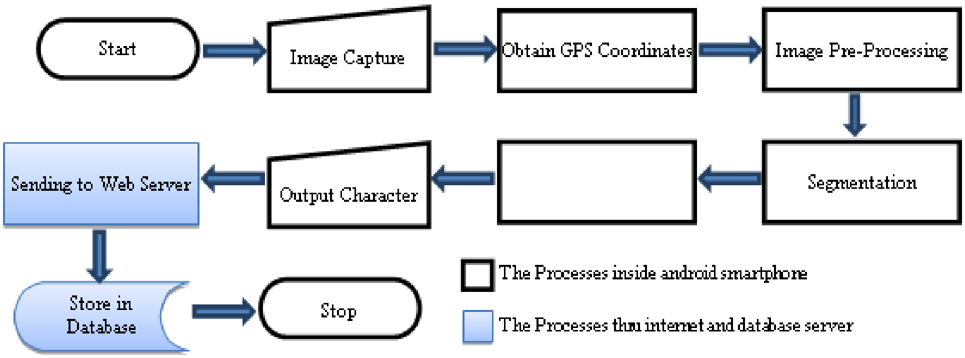

Figure 3. Flow Chart of ANPR in Android Smartphone and Database Server

The process design inside the smartphone consists of image acquisition, GPS coordinates acquisition, image pre-processing, character segmentation, character recognition and the output will consist of the number plate in characters and alphanumeric form, GPS coordinates and smartphone identity, in the form of police officer's email. The process outside the smartphone includes storing the result data into the database server; send back the information to the police officer's smartphone and the last process is sending notification's email to the vehicle's owner. Figure 4 shows the scenario process out of the smartphone through the internet connection and database server.

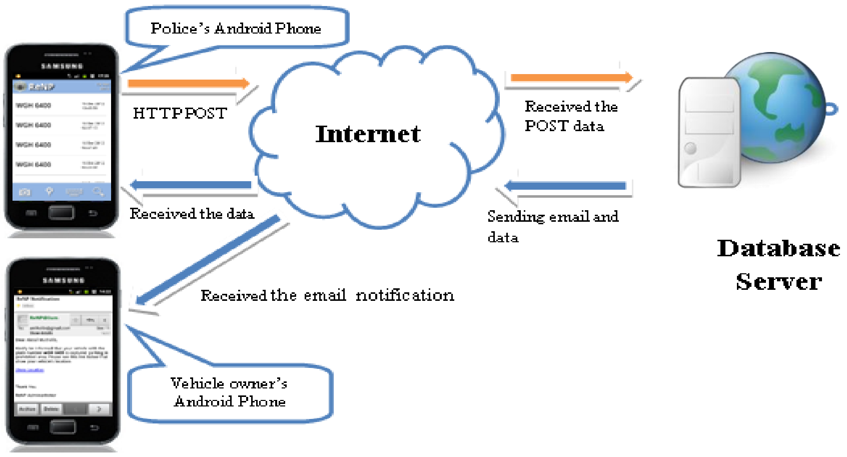

Figure 4. Scenario of Design Process on Android Smartphone and Database Server

### 4.2.3. Database Design

In this research, database is designed to support the ANPR system. It aims to record the data collection of vehicle number plate. The database is required by ANPR system in order to record and retrieve some information related to the number plate such as owner information, vehicle information, and position information of the vehicle. Figure 5 shows the design of database and its relation between two tables namely renp_master and renp_collection tables.

The table design in database has relation of one to many. It can be seen in Figure 5 that the connector shows one (1) in renp_master table and many (∞) in renp_collection table.

The description of this relation is that renp_master table set as master table for the vehicle information. It does not need to be changed frequently and the data will remain the same as the data during vehicle registration except if there is new update like changing ownership. The renp_collection table will be updated frequently as the data received from the ANPR system in Android smartphone. It has many relations due to the required capability to record many data for the same number plate as foreign key which represents the number plate as the primary key in renp_master table.

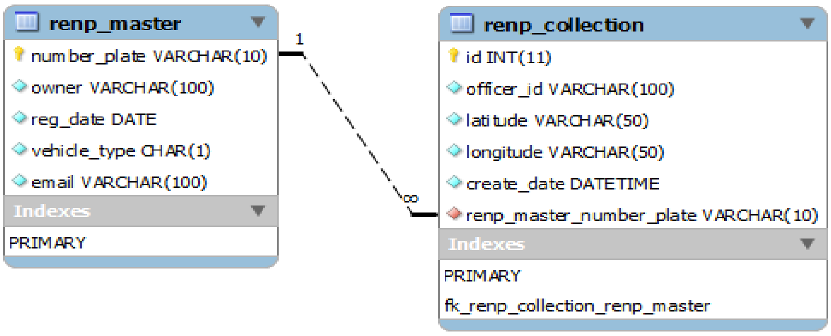

Figure 5. Database Design and Its Table Relation

The first table is renp_master table. It stores the master data of the vehicle and the owner. Table 2 shows that the table consist only five fields, including number_plate, owner, reg_date, vehicle_type and email. Those five fields are presenting the minimum information about vehicle and its owner as the prototype for ANPR system in Android smartphone. This table is only recorded at the first time of vehicle registration.

Table 2. Detail of Table renp_master

| Column Name | Data Type | Length | Description |
|---|---|---|---|
| number_plate | VARCHAR | 10 | Primary key |
| owner | VARCHAR | 100 | Vehicle owner name |
| reg_date | DATE | | Vehicle registration date |
| vehicle_type | CHAR | 1 | 1: Car; 2: Motorcycle |
| email | VARCHAR | 100 | Owner’s email |

The second table is renp_collection table as shown in Table 3. It records the data which are sent from the ANPR system in Android smartphone. For the purpose of this research it contains only six fields, including id, officer_id, latitude, longitude, create_date, and renp_master_number_plate. The same as the table renp_master, this renp_collection is also presenting the minimum information of the vehicle number plate collection. It includes the officer who captures the number plate, position of the vehicle, date when the number plate was taken and the number plate in string as the string format.

Table 3. Details of Table renp_collection

| Column Name | Data Type | Length | Description |
|---|---|---|---|
| id | INT | 11 | Primary key |
| officer_id | VARCHAR | 100 | Police officer id |
| latitude | VARCHAR | 50 | Latitude of taken number plate image |
| longitude | VARCHAR | 50 | Longitude of taken number plate image |
| create_date | DATE | 100 | Date of taken number plate image |
| renp_master_number_plate | VARCHAR | 10 | Foreign key |

## 5. ANPR Algorithm Implementation

In this section, the implementation of ANPR system on Android smartphone are presented. There are various methods for every step in ANPR algorithm. The selected algorithm in every step of the development of ANPR will be discussed in more details.

### 5.1. Acquisition of Number Plate Image

The first phase of ANPR system is acquisition of number plate image. The number plate image is captured using camera that is available in Android smartphone. The Android smartphone used in this research has a camera with 5 mega pixel resolution, which has capability to produce maximum of 2592x1944 pixels image and minimum of 640x480 pixels. In this research, the captured images sizes were 640x480 pixels to reduce the use of storage resources and also to reduce computational complexity. Figure 6 shows the sample of image captured by Android smartphone's camera.

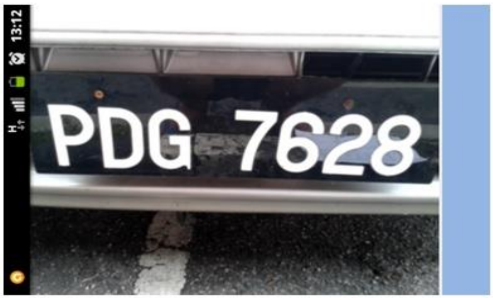

Figure 6. Image Acquired by Android Smartphone's Camera (640x480 pixels)

### 5.2. Image Preprocessing

The next step of ANPR system is image preprocessing. This step is a significant process for the ANPR system. This step will produce an image that should be easily used to perform character segmentation. It means that the captured images need to have minimumnoises and high contrast between character and its background. It is important also to select a suitable threshold of gray level to properly extract the objects from their background.

In this step, edge enhancement method using unsharp masking algorithm, color inversion method and image binarization using Otsu thresholding method are applied.Before those methods performed, the image needs to be converted into grayscale 8 bpp (bits per pixels) in order to reduce computational complexity. To convert the acquired image (640x480, 16-color palette image), the convert To 8 function provided by the Leptonica library will be utilized. Each pixel is stored in 4 bytes and each channel is stored with 8 bits of precision (256 possible values).

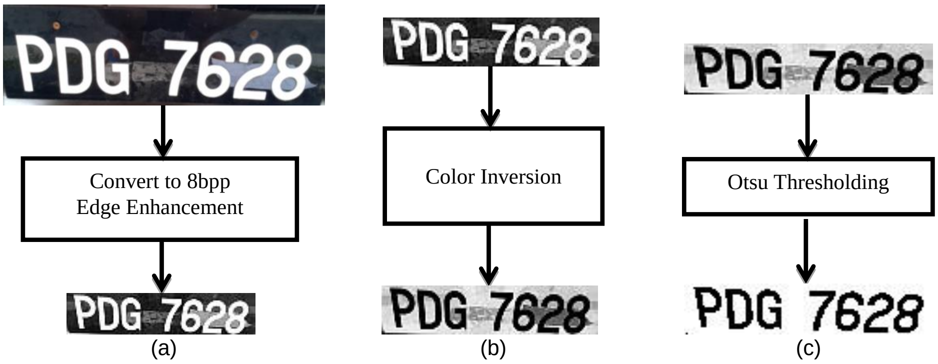

Figure 7. Image Preprocessing Results: (a) Edge enhancement, (b) Color inversion, (c) Otsu thresholding

### 5.2.1. Edge Enhancement

The edge enhancement method works by classifying sharp edge boundaries in the image. It is improving the image contrast in the area proximately around the edge between character on number plate and number plate’s background. It has the effect of producing delicate bright and a dark highlight on either side of any edges in the image. The edge enhancement method is available in Leptonica library under the enhance.c file. Figure 7(a) shows the result of edge enhancement.

### 5.2.2. Color Inversion Method

Color inversion is a method to reverse the color in an image object. It is also well-known as the negative image. In this research it has the purpose to invert between character on number plate color and number plate background color. The color inverting method is produced by deducting each RGB color value from the maximum possible value which is ordinarily 255.The result of this color inversion can be seen in Figure 7(b).

### 5.2.3. Image Binarization

A binary image is a digital image which has only two possible black and white values for each pixel. The convertion to monochrome from a gray scale image is a rudimentary image processing assignment.Otsu’s method is the most popular method to accomplish clustering-based image thresholding. The thresholding can be used to reduce a gray level image into a binary image.The Otsu adaptive thresholding algorithm basically calculates a global threshold for each tile and performs the threshold operation, producing in a binary image for each tile [23]. The idea is to find the threshold that minimizes the weighted within-class variance and it turns out to be the same as maximizing between-class variance [24]. The result of image binarization is shown in Figure 7(c).

### 5.3. Character Segmentation of the Number Plate Image

Generally, the number plates contain the characters which are having the equal width or fixed pitch. It will be easier to perform the character segmentation. The method that we used in this step is fixed pitch detection and chopping as proposed by Ray Smith for the Tesseract OCR engine [20]. The fixed pitch method tests the number plate’s text lines to determine if they are fixed pitch. If fixed pitch text found in number plate, the fixed pitch method chops the number plate into characters using the pitch. Those charactersare ready for the recognition [20]. Figure 8 illustrates the fixed pitch text and how it might slice it.

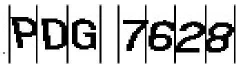

Figure 8. A Fixed Pitch Chopped Method

### 5.4. Character Recognition of Numbers Plate Image

The final step in ANPR system is recognition of numbers plate. A main task of this step is to identify the characters’ image. Based on Table 1, there are numerous of methods have been proposed for the recognition process. ANN is dominating the feature extraction techniques for pattern recognition, while template matching is among the most selected method for number plate recognition. In this research, template matching algorithm is chosen due to its better recognition rate compare to other algorithms and its simplicity. This arrangement is selected due to its impression on the overall performance of the ANPR systems.

## 6. Conclusions

This paper has discussed comprehensive literature review on ANPR algorithms, including preprocessing, localization, character segmentation, and optical character recognition. Leptonica and Tesseract library was used for efficient image processing and character recognition. The system design, including GUI, process, and database design, has been discussed in details. The proposed ANPR algorithm was implemented and results show the effectiveness of our proposed design. More thorough performance evaluation including the recognition rate of ANN versus template matching algorithms need to be conducted. Moreover, the proposed design could be further optimized to balance the processing requirement with the accuracy. Further research includes performance evaluation of the proposed ANPR algorithms using both ANN based and template matching based OCR.

## References

[1] MY Aalsalem and WZ Khan. Campus Sense-A Smart Vehicle Parking Monitoring and Management System using ANPR Cameras and Android Phones. *Journal Editorial Board*. 2016: 809.

[2] A Badr, MM Abdelwahab, AM Thabet AM, and Abdelsadek. Automatic Number Plate Recognition System. *Annals of the University of Craiova, Mathematics and Computer Science Series*. 2011; 38: 62-71.

[3] CJ Lakshmi, AJ Rani, KS Ramakrishna and MK Kiran. A Novel Approach for Indian License Plate Recognition System. *International Journal Of Advanced Engineering Sciences And Technologies (IJAEST)*. 2011; 6: 010 - 014.

[4] MT Qadri and M Asif. Automatic Number Plate Recognition System for Vehicle Identification Using Optical Character Recognition. *International Conference on Education Technology and Computer*. 2009: 335 - 338.

[5] LGC Hamey and C Priest. Automatic Number Plate Recognition for Australian Conditions. *Proceedings of the Digital Imaging Computing: Techniques and Applications*. 2005: 8.

[6] A Roy and DP Ghoshal. Number Plate Recognition for use in different countries using an improved segmentation. in *2011 2nd National Conference on Emerging Trends and Applications in Computer Science (NCETACS)*. 2011: 1-5.

[7] S Ozbay and E Ercelebi. Automatic Vehicle Identification by Plate Recognition. *World Academy of Science, Engineering and Technology*. 2007; 9: 778-781.

[8] H Jun-Wei, Y Shih-Hao and C Yung-Sheng. *Morphology-based license plate detection from complex scenes*. in Proceedings. 16th International Conference on Pattern Recognition. 2002; 3: 176-179.

[9] HAQ Maarif and S Sardy. Plate Number Recognition by Using Artificial Neural Network. *Prosiding Semiloka Teknologi Simulasi dan Komputasi serta Aplikasi*. 2006: 176-182.

[10] KK Kim, KI Kim, JB Kim and HJ Kim. *Learning-based approach for license plate recognition*. in Proceedings of the 2000 IEEE Signal Processing Society Workshop Neural Networks for Signal Processing X. 2000; 2: 614-623.

[11] Y Dai, H Ma, J Liu, and L Li. *A high performance license plate recognition system based on the web technique*. in Proceedings. 2001 IEEE Intelligent Transportation Systems. 2001: 325-329.

[12] O Khalifa, S Khan, R Islam and A Suleiman. Malaysian Vehicle License Plate Recognition, *The International Arab Journal of Information Technology*. 2007; 4: 359-364.

[13] TD Duan, TLH Du, TV Phuoc and NV Hoang. Building an Automatic Vehicle License-Plate Recognition System. *Intl. Conf. in Computer Science – RIVF'05, Vietnam*, 2005: 59-63.

[14] H Yo-Ping, L Shi-Yong and C Wei-Po. *A template-based model for license plate recognition*. in 2004 IEEE International Conference on Networking, Sensing and Control. 2004; 2: 737-742.

[15] HA Hegt, RJ de la Haye and NA Khan. *A high performance license plate recognition system*. in 1998 IEEE International Conference on Systems, Man, and Cybernetics. 1998; 5: 4357-4362.

[16] Z Xiaojun, F Bensaali and R Sotudeh. *OCR-based neural network for ANPR*. in 2012 IEEE International Conference on Imaging Systems and Techniques (IST). 2012: 393-397.

[17] SA Radzi and M Khalil-Hani. Character Recognition of License Plate Number Using Convolutional Neural Network. *International Visual Informatics Conference, IVIC*. 2011; 2: 45-55.

[18] W Ying, L Yue, Y Jingqi, Z Zhenyu, KM von Deneen and S Pengfei. An Algorithm for License Plate Recognition Applied to Intelligent Transportation System. *IEEE Transactions on Intelligent Transportation Systems*. 2011; 12: 830-845.

[19] SH Kasaei, SM Kasaei and SA Kasaei. New Morphology-Based Method for Robust Iranian Car Plate Detection and Recognition. *International Journal of Computer Theory and Engineering*. 2010; 2: 264-268.

[20] R Smith. *An Overview of the Tesseract OCR Engine*. in Ninth International Conference on Document Analysis and Recognition. ICDAR 2007. 2007: 629-633.

[21] D Bloomberg. Leptonica: An open source C library for efficient image processing, analysis and operation. 2007.

[22] RS Pressman. *Software Engineering: A Practitioner's Approach*, 7 ed. Singapore, Mc Graw Hill. 2010.

[23] Z Ningbo, W Gang, Y Gaobo and D Weiming. *A Fast 2D Otsu Thresholding Algorithm Based on Improved Histogram*. in CCPR 2009. Chinese Conference on Pattern Recognition. 2009: 1-5.

[24] Y Zhang and L Wu. Fast Document Image Binarization Based on an Improved Adaptive Otsu's Method and Destination Word Accumulation. *Journal of Computational Information Systems*. 2011; 7: 1886-1892.