

An Automatic Number Plate Recognition System for Car Park Management

Mutua Simon Mandi
Faculty of Information
Technology,
Strathmore University
Nairobi, Kenya

Bernard Shibwabo
Lecturer, Faculty of Information
Technology,
Strathmore University,
Nairobi, Kenya

Kaibiru Mutua Raphael
Lecturer, Department of
Computing and Information
Technology,
Machakos University, Kenya

ABSTRACT

Automatic Number Plate Recognition (ANPR) is an internationally recognized methodology that is used in vehicle identification. ANPR systems allow for real time recognition of a vehicle's number plate. Vehicle parking is an important component within any transportation system, whereby vehicles are often parked at destinations. With an increased number of motor vehicles on roads especially in developing countries, there is need for a vehicle identification mechanism that is effective, affordable and efficient. There are also increased insecurity challenges including terrorism which call for increased surveillance. In most academic institutions and car parks, the ongoing car park entry registration process for visitors, staff or students entering the institution involves a security guard having to confirm membership details by checking for membership sticker on the windscreen of the vehicle or by checking the driver's identification card. This process of writing is tedious and time consuming and is prone to inaccurate recordings, furthermore the backup and sharing of this vehicle information is difficult because the data is hard copy. We propose the adoption of a mobile based software solution that has ANPR capabilities to aid in vehicle identification and vehicle registration. The software application that was developed adopted an object oriented analysis and design methodology, the software developed implements Optical Character Recognition (OCR) using the mobile device camera to detect and capture the vehicle number plate. The proposed solution reduced registration time from 30 seconds to 6 seconds in addition to other benefits. It was recommended that the system be adopted and implemented to address the current challenges in vehicle registration and surveillance.

Keywords

ANPR (Automatic Number Plate Recognition), Vehicle surveillance, Vehicle Parking, Optical Character Recognition

1. INTRODUCTION

Parking is an important component within any transportation system, whereby vehicles must be parked at every destination. Most vehicles are parked most of the time during the day. Parking convenience is the ease of accessing a safe and preferable parking position, hence affecting the ease of reaching destinations [1, 2].

Automatic number plate recognition (ANPR) is a mass surveillance method that was first developed in 1976 at the Police Scientific Development Branch in the United Kingdom with the aim of combating crime. ANPR is a process where vehicles are identified or recognized using their number plate or license plate. ANPR uses image processing techniques in order to extract the vehicle number plate from digital images [3].

As [4] state, ANPR systems generally comprises of a camera, processor and application capable of performing sophisticated optical character recognition (OCR), to transform the image of the plate into alphanumeric characters. Application software to compare the transformed license plate characters to databases of license plates of interest to law enforcement; and a user interface to display the images captured, the results of the OCR transformation, and an alert capability to notify operators when a plate matching an agency's wanted list is observed.

Successful implementations of ANPR systems have resulted in faster and easier vehicle identification. This has also resulted in faster and easier search and retrieval of vehicle information mostly done by law enforcers in identifying vehicles that are uninsured, stolen, or driven by someone without a license or prohibited from driving. Law enforcement use it to control access to restricted areas, to patrol international borders, and in some parts of the world, as a counter-terrorist measure [5].

2. LITERATURE REVIEW

2.1. Automatic Number Plate Recognition

Automatic Number Plate Recognition is a process where vehicles are identified or recognized using their number plate or license plate. ANPR uses image processing techniques so as to extract the vehicle number plate from digital images [3].

ANPR systems normally comprises of two components: A camera that used in capturing of vehicle number plate images, and software that extracts the number plates from the captured images by using a character recognition tool that allows for pixels to be translated into numerical readable characters [6]. A license plate recognition system generally works in four main parts namely image acquisition, license plate detection, characters segmentation, and lastly character recognition [7]. Fig 1 shows a typical ANPR process.

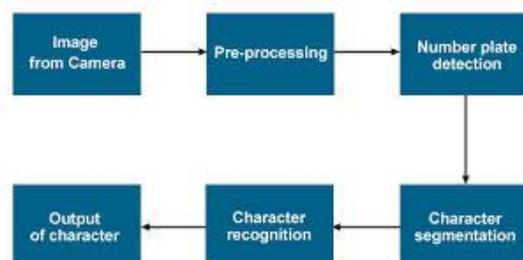


Fig 1: Typical ANPR Process [8]

2.1.1. Image Acquisition

The first step is the image acquisition stage. The image of the vehicle is captured using a photographic camera. The constraint is that the image of the vehicle should be captured

in such a way that the selected input image contains rear or front view of the vehicle with the number plate [9].

The image is usually captured in a RGB (Red, Green and Blue) color model. The captured image is affected by many factors like: optical system, distortion, system noise, lack of exposure or excessive relative motion of camera or vehicle thus resulting in a degradation of a captured vehicle image hence adversely affecting the results of the overall image processing. As a correction mechanism, an image pre-processing stage is introduced to take care of any errors that may have occurred during the image acquisition stage. Image pre-processing mainly involves converting the RGB image into gray color, noise removal, and border enhancement for brightness. Image pre-processing is usually done through image filtering [10].

2.1.2. Number Plate Detection

The next stage that follows is the number plate recognition phase that does several functions such as resizing of the image to a feasible aspect ratio. As well as converting the colored image into a grey scale, image [9].

Number plate detection searches an input image in order to identify specific features that contain the number plate. The number plate can be found anywhere within an image, it is impractical to check all the pixels of the image in order to locate the number plate. Therefore, we only focus on those pixels that have the number plate.

2.1.3. Character Segmentation

Character segmentation can be defined as a technique, which partitions images of lines or words into individual characters. It is an operation that seeks to decompose an image of a sequence of character into sub images of individual symbols [11]. Character segmentation is an operation that seeks to decompose an image of a sequence of characters into sub-images of individual symbols [12].

Character segmentation is the process through which the text component within an image is isolated from the background. In order for proper text recognition to take place the line of text is first segmented, then from the segmented line the words are segmented and then from that the characters are segmented [13].

2.1.4. Character Recognition

Character recognition is process of detecting and recognizing characters from input image and converting it into meaningful text in ASCII (American Standard Code for Information Interchange) or other equivalent machine editable form [14]. Character recognition is the process to classify the input character according to the predefined character class [15].

Template matching, or matrix matching, is one of the most common classification methods. In template matching, individual image pixels are used as features. Classification is performed by comparing an input character image with a set of templates from each character class [14]. A very basic description of a functional ANPR process is:

Step1: The camera takes a picture of the vehicle containing the number plate (Image acquisition).

Step2: The camera isolates the plate, adjusts the brightness and contrast and segments it into characters (Number plate detection and Character segmentation).

Step3: The pattern of each character is analyzed to convert the picture into text (Character recognition).

2.2. Application of Optic Character Reader

There are widespread appearances of commercial OCR products over the last few years meeting different user requirements. There are two main application areas of OCR these include process automation and data entry:

Process automation: This area is not concerned with reading what was printed but rather to control a specific process. For instance automatic address reading for sorting mail thus the aim is to direct each mail (letter) into the appropriate bin automatically by just reading off the mail address [16].

Data entry: This process deals with entering of large amounts of text. This process typically involves the use of camera or scanners to obtain digital images of the original documents, and then the OCR software extracts the text from the images, thus returning the text that is contained in the images [16].

2.2.1. Optic Character Reader on Mobile Devices

Existing mobile applications that implement the use of OCR include:

Nokia Multi-scanner: A freeware application designed for cell phones with Symbian OS. The application supports picture taking and consequently sending it through MMS, Bluetooth or via infrared. It is possible to transfer the image into a text and save it and at the same time the selection of certain area can be made by dragging. Another possibility is to send the image for business card recognition. This option automatically recognizes contact details on the business card and fills in the details for adding a new contact. The OCR engine supports post-processing based on language. However, this solution do not support real virtual keyboard or clipboard copy and paste features furthermore the application runs only on Symbian Operating System [17].

Cam-Card - Business Card Reader: Cam-Card is an application specialized on reading business cards. It is targeted at cell phones which run on OS Android, iOS (OS of iPhone cell phones), or Windows Mobile and BlackBerry phones. Furthermore, the Cam-Card is an extensively automated business card reader with detection of a rotation and a language. The whole recording takes just couple of presses. The main disadvantage is the narrow specialization on business cards and its price [18].

2.3. Elements of ANPR Systems

Automatic Number Plate Recognition Systems typically consist of hardware and software units.

2.3.1. Hardware Components of ANPR System

Camera(s) – Digital cameras are used in the image acquisition stage. They are primarily used in obtaining images or video footage of vehicles.

Infra-Red – Light source at night maybe provided by Infra-red in order to provide illumination for the camera.

Frame Grabber – This is a hardware interface between the digital camera and the computer. The Frame Grabber is tasked with getting the digital image from the camera storing it temporarily and then submits it to the computer for processing.

Computer – This is the central processing unit in ANPR systems. The computer has the ALPR application installed in it [19].

2.3.2. Software Components of ANPR System

ANPR Software – This software has the OCR capabilities, which results in the extraction of the number plate from the digital image.

Database – This provides storage for the data (number plates that have been read by the ANPR).

Back End Software - The back end software is located on a server and provides many functions such as:

- a. Data collection from cameras.
- b. Data mining of previously collected data for investigations and pattern analysis.
- c. Allowing for sharing of data with other agencies [19].

3. METHOD

3.1. System Development Methodology

This research adopted an Object-oriented System Development methodology. An Object-oriented software design is a design strategy where system designers think in terms of 'things' instead of operations or functions.

Object-oriented development methodology ensures that the system being developed is refined and transformed through analysis, design, code and test phases. Details and modifications are added in successive iterations (changes and improvements are introduced as needed) and incremental releases of software modules are delivered [20].

Object-oriented software development life cycle is an iterative process that has five key phases. Fig 2 is an image detailing the key stages in this methodology. These phases include:

- i. Requirements analysis: - This phase is critical to the success of the project. Expectations need to be fleshed out in great detail and documented. This is an iterative process with much communication taking place between stakeholders, end users and the project team [21]. The key stakeholders and users included the University's Head of Security as well as the Security Personnel manning the entry points of the University, the Researcher interacted with them so as to fully understand their day to day processes as well as to collect/gather the desired system features.
- ii. Design: - During this phase, the technical design requirements are prepared. The User requirements are used to define how the application will be written; the technical requirements are specified to detail for instance; database to be used, features and functionalities, security processes and hardware and system requirements [22].
- iii. Code: - At this stage, the design is translated into a machine-readable form. Programming tools like compilers, interpreters, debuggers are used to generate the code. Different high level programming languages like, C, C++, Java, PHP, HTML are used for coding.
- iv. Test: - This stage occurs after the application has been developed; different types of testing will be performed including performance and integration testing. User acceptance testing is the last part of testing and is performed by the

end users to ensure the system meets their expectations. At this point, defects may be found and more work may be required in the analysis, design or coding [22].

- v. Maintenance: - This phase confirms the software passed the user acceptance stage and now is operational. If required, the users are trained on, or aided with the documentation on how to operate the software and how to keep the software operational. The software is maintained timely by updating the code according to the changes taking place in user end environment or technology. This phase may face challenges from hidden bugs and real-world unidentified problems.

Fig 2 shows the system development cycle used in development of ANPR system.

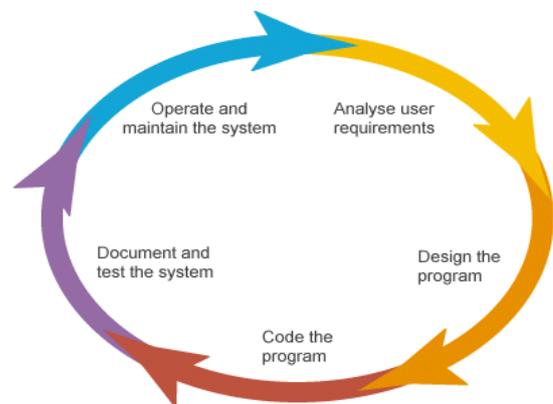


Fig 2: System Development Life Cycle

4. SYSTEM DESIGN AND ARCHITECTURE

4.1. Requirement Analysis

A requirement is any function, constraint, or property that the system must provide, meet, or satisfy in order to fulfill its purpose. The goal of analysis is to produce essential requirements [23].

The main requirements of the system were obtained at the data collection stage. The major stakeholders of the system were interviewed and observed as they performed their day to day task in order for the researcher to fully understand the current process.

The results that were obtained from data collection mainly included the current vehicle entry registration processes into car parks and some of its challenges. This information has been summarized in the steps that follow:

- i. Vehicle arrives at the entrance/gate with aim to gain entrance.
- ii. The security guard first checks if the driver has a valid ID (i.e. staff or student ID) and/or sticker.
- iii. If the ID is valid the gate is opened.
- iv. There is another security guard who hands out the ticket number while noting down on an occurrence book the vehicle details. These details include vehicle make, model and color, time of entry, as well as the ticket number issued to the vehicle.
- v. On vehicle exit, the ticket issued is returned back to the security guard at the gate and the time of exit is

recorded for that particular vehicle.

4.2. System Architecture

Fig 3 is an architectural design for the Automatic Number Plate Capture and Vehicle Registration System. This architecture can be broadly categorized into two:

- Client Application (Front-End)
- Server side (Back-End)

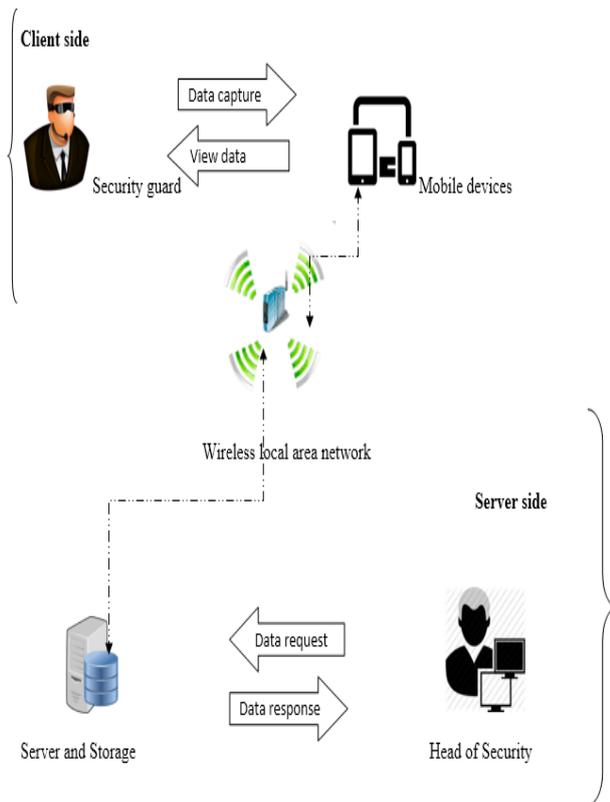


Fig 3: System Architecture

4.2.1. Client Application

The solution comprises the mobile application that has the Optical Character Recognition functionality required in the scanning of vehicle number plate, as well as inputting other vehicle description details. The mobile device also allows for additional services like capturing the vehicle time in by using the mobile phones' clock functionalities, as well as storage of the vehicle information. The mobile application is communicating with the server side via wireless local area network (WLAN).

4.2.2. Server Side

This consists of a server used for hosting the scripts that will be involved in the transmitting and receiving of the vehicle information. There is also a database used for storage of vehicle details as well an administrator's portal that is used in viewing of the reports of the vehicles that have been recorded.

With regards to the architectural design of the Automatic Number Plate Capture and Vehicle Registration System. The Security guards are the primary users of the mobile application, while the Head of Security at the Institution will be a secondary user whereby He will be receiving the information of the vehicles entering and exiting the University's premises from an online portal.

4.3. Use Case Diagram

A use case diagram at its simplest is a representation of a user's interaction with the system and depicting the specifications of a use case. A use case diagram can portray the different types of users of a system and the case and will often be accompanied by other types of diagrams as well. Fig 4 shows a use case diagram describing the actors and the activities involved in the new vehicle registration process.

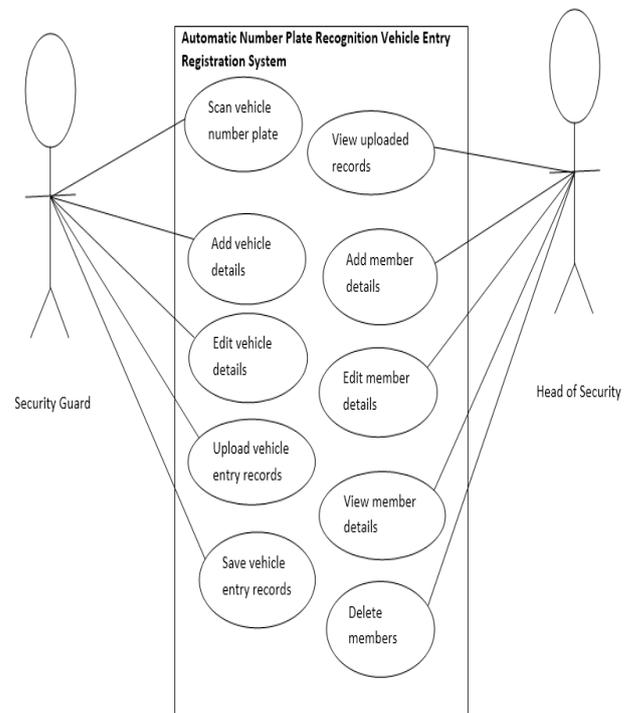


Fig 4: Use Case Diagram for Automatic Number Plate Recognition System

4.4. Context Diagram

A Context Diagram is a single picture that has the system of interest at the center, with no details of its interior structure or function, surrounded by those elements in its environment with which it interacts [24].

Fig 5 is a Context Diagram for the Automatic Vehicle Entry Registration System detailing the interactions with the users of the system that include the Security Guard and the Head of Security Officer.

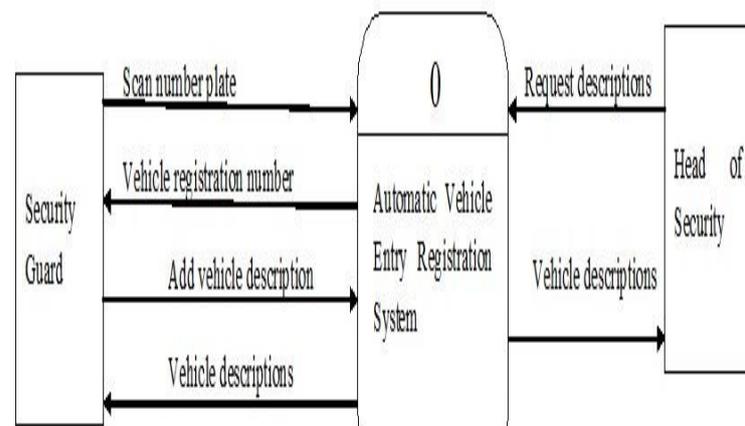


Fig 5: Context Diagram

5. SYSTEM IMPLEMENTATION AND TESTING

5.1. System Development Tools

This system architecture is split into two main categories namely the client side and the server side. The tools required in the development of the client side application include, Android SDK (Software Development Toolkit) version 4.0.3, Abby FineRead OCR library, Eclipse IDE (Integrated Development Environment).

As for the server side the main development tools include the use of MySQL database for the storage of vehicle data, HTML (Hyper Text Markup Language) for creation of web-pages, PHP (Hypertext Pre-processor) used in the writing of scripts that are used to interact between the web-pages and the database.

5.2. Application Screenshots

The developed application had the following representations. Fig 6 is the application home page having four buttons i.e. Entry, Logs, Members, Records. When the user clicks on the Members button, the members' page is displayed as shown in Fig 7. On the members' page, the user has the options to add, delete, view, or modify any of the member details. Some of the member details include vehicle number plate, make of the vehicle, and color of the vehicle as well as the name of the member.



Fig 6: Application Home Page

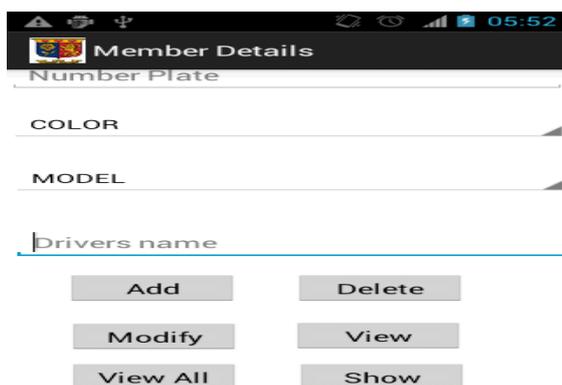


Fig 7: Members Page

Once the user clicks on the Entry button on the home page the application opens the device camera. Once the number plate has been captured as depicted in Fig 8, it is displayed to the user on the left side of the screen. Once the user clicks on the number plate a dialog box appears with two buttons search and cancels as show in Fig 9. When the user clicks on search, the captured number plate is queried against the members' list if the number plate exists in the members' list then the stored vehicle details are displayed as shown in Fig 10.



Fig 8: Vehicle Scan

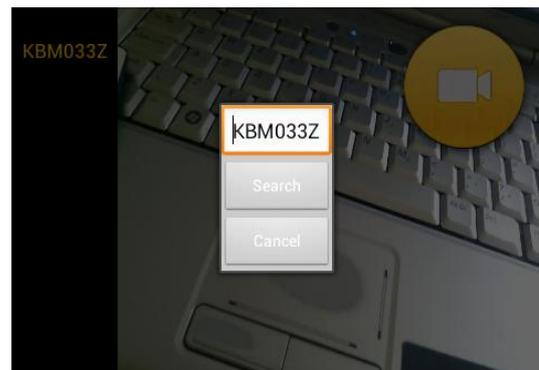


Fig 9: Search Dialog

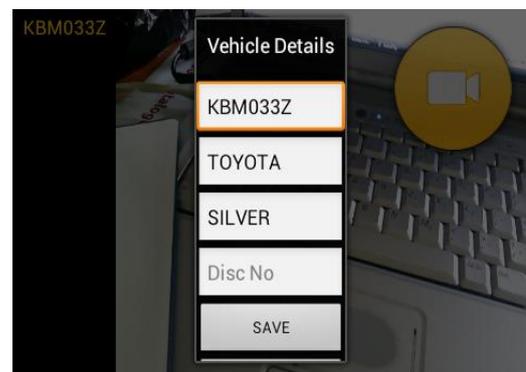


Fig 10: Details Dialog

Fig 11 is a sample screen shot if the number plate captured does not exist in the members' list then application displays a different dialog box, allowing the security guard to capture the additional details such as vehicle type, color as well as ticket number assigned to the vehicle.

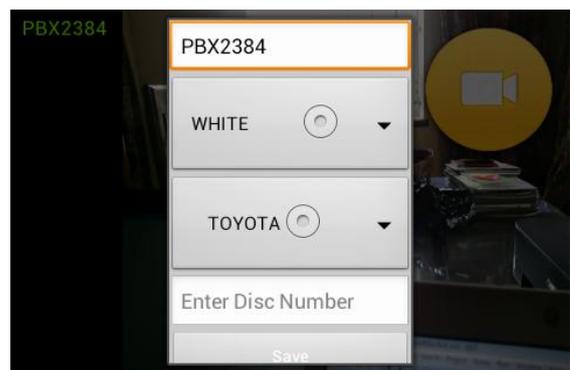


Fig 11: Details Dialog

When the vehicle is about to exit, the user gets a pop-up dialog prompting him or her to confirm that the vehicle is about to exit as shown in Fig 12.

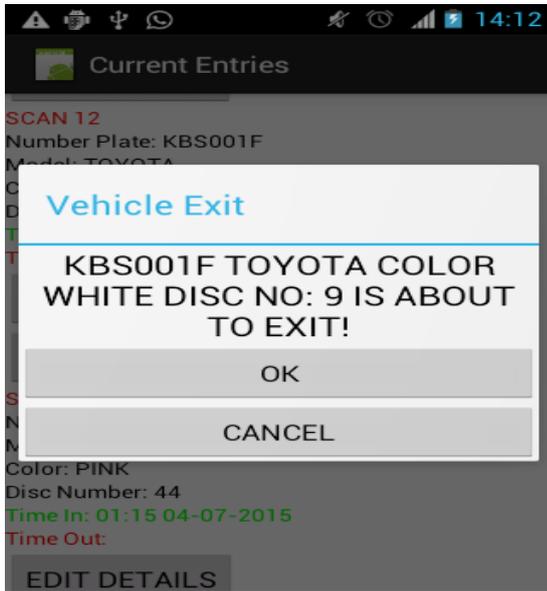


Fig 12: Vehicle Exit Dialog

Fig 13 shows a section of the backend functions which include a reports and analytics page. The backend provides this and statistics.

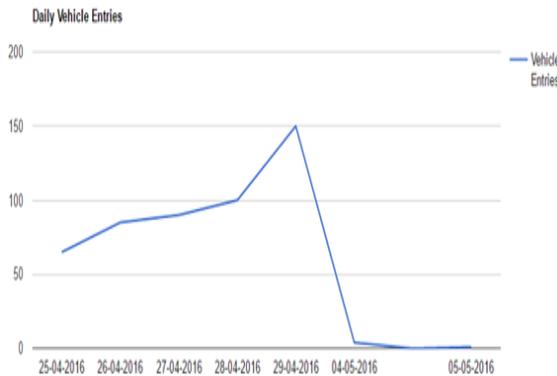


Fig 13: Daily Vehicle Records Report

5.3. Testing Results

In order to understand the problem, the researcher investigated the current vehicle entry registration process at a learning institution and found out that, it took the security guards approximately 30 seconds to register a vehicle.

Due to the challenges with the current system the researcher was able to propose a solution for the registration of vehicles as they entered the Institution. The proposed solution involved the use of Optical Character Recognition in order to automatically capture, scan and display a vehicle's number plate in a timely and user-friendly manner.

A user acceptance test was carried out on the first version of the application. This test was performed in order to check whether the application was able to meet the user's expectation in a real world with regards to the vehicle entry registration process. The user acceptance test was successful

due to the reason that the mobile application is able to successfully detect, scan and display the vehicle number plate in an average time of six (6) seconds (which is even lower without human interaction with the solution).

Table 1: Comparison between current and the Proposed System

Vehicle No.	Time taken to write down vehicle details (T-1)	Time taken to scan number plate and add vehicle details (T-2)
1.	30 sec	7 sec
2.	25 sec	7 sec
3.	35 sec	5 sec
4.	40 sec	6 sec
5.	20 sec	5 sec
6.	25 sec	5 sec
7.	35 sec	8 sec
8.	45 sec	8 sec
9.	40 sec	7 sec
10.	40 sec	6 sec
11.	30 sec	5 sec
12.	25 sec	6 sec
13.	35 sec	5 sec
14.	30 sec	5 sec
15.	30 sec	6 sec

6. CONCLUSION

Problems and challenges associated with car park vehicle identification details formed the basis for this research. The main aim of this research was to develop an automatic number plate recognition system for car park management, using Optical Character Reader (OCR) on a mobile device.

The OCR process forms the basis of the entire system that was proposed by the researcher as a means of tackling the challenges faced by the security guards during the vehicle entry registration process. The completion of the proposed system resulted in the following benefits:

- i. Elimination of the hard copy occurrence book and the need to have to physically write onto the book, because all the vehicle details records will be digitized.
- ii. Hastening of the car park vehicle identification process including the entry and exit process, thus shortening the time duration.
- iii. Accurate recording of vehicle information.
- iv. Provides a means of easy information sharing and

information backup.

- v. Real time information sharing of the vehicles entering and exiting the Institution to the Head of Security.
- vi. Easier analysis of the vehicle information captured.

In future, an implementation an ANPR system can be useful in: Identification of vehicles that either have been reported as stolen or are wanted for having committed traffic offenses, or have fake registration certificates. This can be even be more useful in enhancing border patrol and border surveillance.

7. ACKNOWLEDGEMENTS

Thank you Strathmore University for the support in this study, especially thanks to the Faculty of Information Technology.

8. REFERENCES

- [1] Litman, T. 2013. *Parking Management Strategies, Evaluation and Planning*. Retrieved from http://www.vtpi.org/park_man.pdf
- [2] Subraman T. 2012. Parking Study on Main Corridors in Major Urban Centre. *International Journal of Modern Engineering Research (IJMER)* ISSN: 2249-6645. Retrieved from http://www.ijmer.com/papers/vol2_issue3/AE23742748.pdf
- [3] Cornwall. 2009. Drivers on police files for life. Retrieved from <http://www.westbriton.co.uk/Drivers-police-files-life/story-11398048-detail/story.html>
- [4] Roberts & Casanova 2012. *Automated License Plate Recognition Systems: Policy and Operational Guidance for Law Enforcement*. Retrieved from https://www.aclu.org/files/FilesPDFs/ALPR/federal/NH_TSA/15948-16075DOJ-IACP%20report.pdf
- [5] ACPO (2013). The police use of Automatic Number Plate Recognition Retrieved from <http://www.acpo.police.uk/documents/crime/2013/201303CBA-ANPR.pdf>
- [6] Friedrich, M., Jehlicka, P. & Schlaich, J. 2008. *Automatic number plate recognition for the observance of travel behavior*. Retrieved from http://www.isv.uni-stuttgart.de/vuv/publication/downloads/200805_Fr_PJ_JS-ANPR.pdf
- [7] Lotufo, R., Morgan, D. & Johnson, S. 2013. Automatic license plate recognition (ALPR) a state-of-the-art review. *Journal of IEEE transaction on circuits and system for video technology*, vol. 23, no, 2013, pp. 311-325 DOI:10.1109/TCSVT.2012.2203741
- [8] Camera-sdk Retrieved from http://www.camera-sdk.com/p_89-how-to-implement-number-plate-recognition-in-c-onvif.html
- [9] Reshma, P. 2012. *Noise Removal and Blob Identification Approach for Number Plate Recognition*. Retrieved from <http://research.ijcaonline.org/volume47/number8/pxc3879992.pdf>
- [10] Dhiraj, Y., Pramod, G. & Borole, B. 2014. *A Review Paper on Automatic Number Plate Recognition (ANPR) System*. Retrieved from <http://www.scribd.com/doc/229724157/A-Review-Paper-on-Automatic-Number-Plate-Recognition-ANPR-System>
- [11] Kumar, R. & Singh, A. 2011. *Character Segmentation in Gurumukhi Handwritten Text using Hybrid Approach*. Retrieved from <http://www.ijcte.org/papers/357-G831.pdf>
- [12] Casey, G. & Lecolinet, E. *A Survey of Methods and Strategies in Character Segmentation*. Retrieved from <http://perso.telecom-paristech.fr/~elc/papers/pami96.pdf>
- [13] Saha, S., Basu, S., Nasipuri, M. & Dipak, B. 2010. *A Hough Transform based Technique for Text Segmentation*. Retrieved from <http://arxiv.org/ftp/arxiv/papers/1002/1002.4048.pdf>
- [14] Shah, K. & Sharma, A. 1998. *Design and Implementation of Optical Character Recognition System to Recognize Gujarati Script using Template Matching*. Retrieved from <https://books.google.co.ke/books?id=vdCeBQAAQBAJ&pg=PA1&lpg=PA297&dq=#v=onepage&q&f=false>
- [15] Dedgaonkar, G., Chandavale, A. & Sapkal, M. 2012. *Survey of Methods for Character Recognition*. Retrieved from http://ijeit.com/vol%201/Issue%205/IJEIT1412201205_36.pdf
- [16] Eikvil, L. (1993). *Optical Character Recognition*. Retrieved from <http://www.nr.no/~eikvil/OCR.pdf>
- [17] Daily mobile 2008. Retrieved from <http://www.dailymobile.net/2008/10/24/symbian-application-nokia-multiscanner/>
- [18] iMore 2012. *CamCard vs. WorldCard vs. Business Card Reader: card scanner for iPhone app shootout*. Retrieved from <http://www.imore.com/camcard-worldcard-business-card-reader-card-scanner-iphone-app-shootout>
- [19] Competitive Survey and White Paper of Automated License Plate Recognition Vendors (n.d). Retrieved from https://www.aclu.org/files/FilesPDFs/ALPR/arizona/alpr_pra_PhoenixPD_PhoenixAZ_4.pdf
- [20] Leszek A. M. 2007. *Requirements Analysis and Systems Design*. Pearson Education Canada.
- [21] Murch R. (2012). *The Software Development Lifecycle - A Complete Guide*. Amazon Digital Services LLC
- [22] Alwan M. 2016. Getting even more context for errors & exceptions. Retrieved from <https://airbrake.io/blog/category/insight>
- [23] Kristensen T. (2016) *Computational Intelligence, Evolutionary Computing and Evolutionary Clustering Algorithms*. Bentham Science Publishers.
- [24] Burge, S. 2011. *The Systems Engineering Tool Box* <http://www.burgehugheswalsh.co.uk/uploaded/document/s/CD-Tool-Box-V1.0.pdf>