A Review of Automatic License Plate Recognition System in Mobile-based Platform

L. Connie[1], C. Kim On[1], A. Patricia[2]
[1]Faculty of Computing and Informatics, Universiti Malaysia Sabah,
88400 Kota Kinabalu, Sabah, Malaysia
[2]Faculty of Environment, Society and Design, Lincoln University, Christchurch, New Zealand.
kimonchin@ums.edu.my

***Abstract*—Automatic license plate recognition (ALPR) is the process of retrieving license plate information from a captured image or video frames from a sequence of videos. ALPR can assist law enforcement officers to identify stolen vehicles or to capture vehicle information from those that violate traffic laws instantly. It is also commonly used as an electronic payment system for toll payment or parking fee payment. Traditionally, ALPR is installed in a PC-based platform to take advantage of its processing power to process high-quality images captured by high-resolution cameras. Most smartphones nowadays are equipped with a high-quality camera and faster processing system which can be used to develop portable ALPR system. Thus, this has encouraged many researchers to work on implementing ALPR technology for the mobile platform. In this paper, we reviewed several researches that have implemented ALPR in the mobile-based platform. We discuss the techniques used in the three main stages of ALPR namely localisation, segmentation and recognition. The advantages and disadvantages of each technique are summarised in a table.**

***Index Terms*—Automatic License Plate Recognition (ALPR); Car Plate Recognition System (CPRS); Mobile Platform; Smartphone; Vehicle License Plate Recognition (VLPR).**

I. INTRODUCTION

Automatic license plate recognition (ALPR) is also known as car plate recognition or vehicle license plate recognition. It utilises image processing technology in order to extract and recognise license plate information from an image or video frames. The extracted information can be used in various applications such as electronic payment gateway system, parking fee payment system, road monitoring system and traffic control system [1-3]. In real life application, ALPR's researchers have to cope with various challenges such as type of license plate, type of fonts, color and font of the plate, location of the number plate, and also the environmental conditions such as brightness, weather, and objects which will increase the difficulty in detecting and recognizing the license plates [16]. The formats of the license plate usually vary from country to another such as different colours, languages and fonts. Some plate might have borders with different colours from the background surrounding the plates, and some might have a single colour background which indirectly caused additional challenges in detecting and recognising the car plate. Variations of environmental conditions such as illumination and image background will also affect the license plate recognition rate [4].

ALPR system can be divided into four main stages which are image acquisition, license plate localisation, segmentation and character recognition. Figure 1 shows the general ALPR system flowchart. In order to gain a high accuracy rate of license plate recognition, the images must be with good quality. Several factors can affect the quality of an image. These include the type of camera used, camera resolution, light and orientation of the camera used for acquiring the input images.

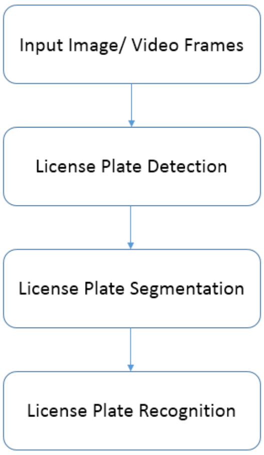

Figure 1: ALPR system general flowchart

The traditional ALPR system usually requires a high-resolution camera and a computer or notebook to process the complex ALPR algorithm. With the advances in the mobile technology, many mobile devices are now equipped with a strong CPU and high-resolution camera which meet the requirement of a good license plate recognition system. The quality of cameras in high-end smartphones is comparable to those highly expensive digital cameras. As hardware result, many complex applications for daily activities have been transferred from PC into the mobile platform. Thus, it is now possible to implement license plate recognition algorithm in the mobile platform.

Conventional ALPR system is usually bulky as it needs to communicate with a fixed-point camera, which is not portable. A portable ALPR will be highly beneficial to law enforcement officers as they can carry around the device and perform traffic monitoring anywhere and anytime. It includes identifying for stolen vehicles and retrieving information of vehicles that violated traffic rules.

Conventional ALPR system is usually very expensive to install and costly to program and maintain. In contrast, the implementation of ALPR in a mobile device is cheaper. However, there are still very limited researches that have been conducted thus far for ALPR on a mobile platform.

In this paper, we reviewed several researches and approaches that have implemented ALPR in the mobile-based platform. We discuss the techniques used in three main stages of ALPR namely localisation, segmentation and recognition.

The remainder of this paper is organised as follows. Section II describes the license plate detection and extraction method. The license plate segmentation is discussed in Section III. Section IV discusses the license plate recognition, and finally, Section V summarises and concludes the findings.

II. License Plate Detection/Extraction

The first stage in ALPR is the license plate detection or extraction. The input for this stage is an image or a video frame. The image or the video frame is usually captured from a smartphone device. The output for this stage is then used for license plate location. This stage is very important as the result of the license plate recognition success rate is highly dependent on the image or the video frame. Note that license plate can exist anywhere in the image. The license plate can be distinguished by its feature in an image, such as edge boundary, size, colour and existence of character feature. With the knowledge of these features, the processing time for detecting a license plate can be reduced by only processing the pixels that contain these features. This step is very important to prevent bottlenecking or crashes in the mobile apps, as the mobile device has limited computational resources compared to desktop PC.

The earlier ALPR systems in the mobile platform have been started by [5] and [8]. The localisation process was irrelevant in [5] and [8] as the license plate image was captured from the device. This approach is not suitable for the real-world situation as license plate normally needs to be detected under various conditions such as different background, illumination and light condition.

Wavelet transformation was used in Android device to detect and locate license plate in [6]. The wavelet transform is a mathematical based approach, and it is suitable for mobile-based ALPR system due to its low complexity development and simple algorithm. In this paper, wavelet transformation was used to identify the horizontal edges by scanning the ROI of the license plate. Next, by using the knowledge of the license plate aspect ratio, the horizontal edges were scanned with a block of the aspect ratio size to get the license plate location.

In [9], the morphologically based method was used for license plate detection. As license plate is normally designed in rectangular shape, this method used the geometrical attribute of a license plate shape for detection. After applying morphology operations (erosion followed by dilation), the input image will have to undergo binarisation in order to find the contours. This is to allow the process of acquiring a rectangular region in the image. This method had shown a high localisation rate of up to 98% detection rate in 200 images. The morphologically based method is ideal for implementation in mobile due to its less complex algorithm compared to other methods such as edge detection method. By using the less complex algorithm, it is much more efficient, and it helps to reduce the processing time in the mobile device.

In [7], the research combined Android mobile patrolling robot with license plate recognition system in an Android-based smartphone for license plate searching, detection and recognition. For the localisation part, the system performed overall license plate detection. When one or more license plate is detected, then the Android-based robot will evaluate the nearest license plate distance and move towards it. Upon approaching the nearest plate, the vertical edge was enhanced by deconvoluting the scene image with a horizontal-direction Wiener filter followed by applying the AdaBoost cascade classifier. This approach was able to detect license plate accurately under various complex background.

In [10], the colour feature and the edge information were combined in order to extract the license plate. In the localisation based on colour segmentation, HSV colour space model was used as license plate colour threshold interval for binarisation purpose. After removing noises in the binary image, the morphological operation was used to fill the gaps of certain disconnected areas. Then, the license plate region was extracted based on the contours rectangular shape ratio and the contours area size which meet the characteristics of a license plate. The second method used the Sobel operator as edge detection method. The candidate region obtained from the two methods was intercepted to get the final license plate location. Table 1 summarises the advantages and disadvantages of methods used in license plate detection, particularly in the mobile-based platform.

Table 1
Advantages and Disadvantages of Methods used in License Plate Detection in Mobile-based Platform

Methods	Advantages	Disadvantages	References
Capture license plate image by directly pointing the camera towards license plate	Reduce processing computational power required in a mobile device to perform image processing on license plate localisation	Not feasible for real-world usage, cannot be applied to complex background and environment	[5], [8]
Wavelet transformation method	Suitable for less complex background image. Less complex algorithm which is ideal for limited computational power mobile device	When the background is more complex and has many edges, it will become computationally complex	[6]
Morphology-based method	Less computationally complex algorithm which is ideal for limited processing power in mobile device	When background is complex, might contain more false license plate detection	[9]
AdaBoost cascade classifier	More accurate license plate detection rate and able to detect license plate with complex background	Requires much processing power and might result in slower processing time	[7]
Combining colour information and edge information feature	More reliable and license plate detection accuracy is higher	Combining two features are more computationally complex which will result in slower processing time	[10]

III. LICENSE PLATE SEGMENTATION

After the extraction/detection of the license plate area, the next process is the segmentation of the license plate. During this process, the number plate image character is split repeatedly until every character is separated into individual sub-images. This phase is important in any ALPR system as each segmented characters will determine the rate of character recognition accuracy. If the segmentation of a license plate fails, the result will cause a character to be divided improperly, or two characters are mistakenly merged as one segmented character. There are some cases where the license plate extracted from the localisation step have some problems such as dots, tilting, stains on plates and different illumination condition across the license plate. For example, unwanted stains that are present on the license plate might cause the system to mistakenly recognise it as characters which lead to recognition error. However, these problems can be easily solved by applying preprocessing processes such as mathematical morphology and vertical edge projection histogram used in [7], which will be discussed further in the next section.

In [5] and [6], both researches used a projection profile method for their license plate character segmentation process which is horizontal projection and vertical projection respectively. In [5], the segmentation process was done by using a method known as a horizontal projection of the license plate. Basically, it is a process of finding the maximum peak or also known as the space between characters by the vertical projection of the graph iteratively. When the algorithm meets a certain condition such as the height of the peak, it will zeroizes the peak iteratively until no spaces are found. A vertical projection method was also used in [6] for a license plate character segmentation process in a mobile phone. In this method, the vertical axis is scanned for a black point in the images in order to detect the character. If the black point is found within the vertical axis, the vertical line is marked. By using the projection profiles method for license plate segmentation, it has recorded a very high accuracy rate with fast processing speed at 99.2% and 10-20ms respectively for 30000 images [15].

In [10], a combination of horizontal projection was used for character segmentation. It was used to remove upper and lower borders of the extracted license plate image. Meanwhile, the vertical projection was used for characters segmentation. This implementation process is almost similar to the method used in [6].

In [8], connected pixels information was used for the license plate segmentation process. Each group of connected pixels were labelled as one single character. The labelled pixels were analysed against aspect ratio of characters and those that contained pixels number less than the threshold value were considered as noise and were ignored. The character connected pixels were cropped out by searching for its left, right, upper and lower edge and the cropped out portion was resized and scaled for the character recognition process. However, this method assumed that all characters in the license plate were separated. It may not work best for a license plate with characters that appear connected or joined together.

There are two preprocessing techniques were used to enhance extracted license plate before character segmentation in [7]. The techniques are mathematical morphology and vertical edge projection histogram segmentation. Mathematical morphology was used to remove stains and dirt on the license plate. It is less computationally complex and consumed fewer resources on the Android device which makes it ideal to be implemented on a mobile platform. The vertical edge projection histogram segmentation was used to remove larger unwanted elements such as border stains. It is a method with a combination of fill-in and connecting labelled component for border removal and character segmentation. An aspect ratio of the border was used in order to differentiate characters from borders to avoid false removal of characters component. However, this character segmentation method is only useful for license plate containing borders.

In [9], each character was segmented by finding rectangle region in the extracted license plate. The algorithm involves binarisation and contours finding on the extracted license plate before detecting the rectangle region. Each individual contour was bounded with a rectangular bounding box as a region. Then, it was analysed under a certain aspect ratio. Only detected region that met the criteria was kept for character recognition, while the rests were removed. However, this method might perform slowly on certain mobile devices if the necessary processing power is limited. Table 2 summarises the advantages and disadvantages of methods used in character segmentation, particularly in mobile-based platform

Table 2
Advantages and Disadvantages of Each Method Implemented in Character Segmentation in Mobile based Platform

Methods	Advantages	Disadvantages	References
Projection profiles method (vertical and horizontal projection)	Able to segment character regardless of character position and less complex. The algorithm can perform faster in mobile devices	Extra noises in the plate will affect the results of the segmentation	[5], [6], [10]
By using connected pixels information in extracted license plate	Less complex algorithm, able to segment character regardless of license plate rotation and position	Characters must be separated, cannot extract joined characters	[8]
Combining mathematical morphology and vertical projection edge histogram projection method	Able to remove noise, dirt and borders in extracted license plate. Resulting in much higher accuracy segmentation and more reliable	Combining features resulting in more computationally complex algorithm. It is not ideal for limited computational resource mobile devices	[7]
Detecting rectangle region by using characters contours	Able to get characters exact boundaries	Contours finding may be slow and not suitable for real-time usage as it may generate incomplete or distorted contours when license plate angle is different	[9]

IV. LICENSE PLATE RECOGNITION

In this phase, all the extracted characters from the segmentation stage will undergo recognition process. There are a number of methods used for the recognition process.

Artificial Neural Networks (ANN) is one of the commonly used approaches for license plate recognition due to its ease of code, less complexity, fast recognition, and highly promising performance in various areas. Some of the commonly used ANN has combined with Backpropagation Neural Network (BPNN), Radial Basis Function Network (RBF), and Ensemble Neural Network (ENN) [12].

Recently, researchers have started to implement the Convolution Neural Network (CNN) in ALPR localisation and character recognition phase [17-20]. However, CNN requires usage of GPU processing power for its training model for faster processing time, which might not be ideal for implementation in mobile devices. The main reason why ANN is among the popular methods for license plate recognition is that the training and classification phases are done separately. Since the training process might take a long time, it can be done separately on the desktop.

Later it performs a classification process in the mobile device which has limited computational resource. Apart from ANN, existing Optical Character Recognition (OCR) engine such as Tesseract OCR is often used for the recognition process [21]. In this section, we will review character recognition methods that have been applied to existing research in license plate recognition in a mobile device. The summaries for the advantages and disadvantages of methods used for character recognition is shown in Table 3.

In [5] and [8], both researches used the ANN approach for the character recognition process. In [5], the OCR algorithm was used to compare the segmented image with a set of the alphanumerical database that was trained with ANN. The OCR also used correlation method to counterpart individual character by determining the degree of similarity of data patterns stored in the alphanumerical database. In [8], the segmented character was resized into 14x10 pixels image single character before feeding into NN. The training process was done on a desktop computer instead of the mobile device. The NN was trained with MATLAB using its backpropagation algorithm. The weights resulting from the training process was then exported in the form of fixed weights matrix in order to be implemented in the mobile device with the same NN architecture used during the training process.

Another commonly applied character recognition method is the template matching, and it was implemented in [6] for recognition purpose. Basically, template matching measures the similarity between a character and the template itself. The template is the sample of license plate character image which could be a set of same size individual alphanumerical image that is stored in a database. In [6], the template contained a set Taiwanese standard font which consisted of a total of 35 alphanumerical fonts. The segmented character was resized into 30x30 pixels image in order to match the extracted character with the templates. However, this method might be slow when applied in a mobile device because it would need to process all the unnecessary pixels in the image and it was limited to recognise only the fonts that were available in the template.

In [7] and [9], an open source OCR library known as Tesseract OCR engine [13] was used to perform the character recognition process. Tesseract OCR as built for OCR in general documents with common character fonts, it was not designed specifically for license plates recognition. Therefore, in order to make it usable for license plate recognition, [7] retrained the character dictionary in Tesseract OCR with Taiwanese license plate font characters. Then, the customised Tesseract OCR was ported to the Android smartphone through the Java Native Interface (JNI) technique. In [9], the authors implemented two methods for character recognition which were Tesseract OCR engine and Kohonen neural networks [11]. In order to compare the recognition rate and processing time of each method, the Tesseract OCR used was also retrained with a collection of real license plate images. The results from [9] showed that Tesseract OCR have a better recognition rate which was 3% higher compared to Kohonen network, while Kohonen network has a slightly faster average processing time which was 0.3% faster compared to Tesseract OCR engine.

In [10], Chen implemented Support Vector Machine (SVM) to recognise Chinese license plate character. The segmented character was normalised into a 40x20 image before feeding it into the training network. A multiclass classifier for recognising multiple characters was used by designing SVM between two types of character. The training process was done using OpenCV CvSVM.train function [14]. OpenCV also provides an extension for implementation of multiple image processing algorithms in mobile platforms such as iOS and Android. The results showed that the proposed method performed well with the alphanumerical character, but it was not ideal for Chinese character recognition.

Table 3
Advantages and Disadvantages of each Method Implemented in Character Recognition in the Mobile-based Platform

Methods	Advantages	Disadvantages	References
Artificial neural network (ANN)	Training and classification process can be separated, which will not consume too much mobile device processing power.	Huge number of training data is required in order to achieve high recognition accuracy.	[5], [8]
Template Matching	Simple and straightforward application especially when using foreign characters.	Application is only limited to sets of templates available. It cannot work well when license plate character contains different design or fonts	[6]
Tesseract OCR engine	High recognition accuracy and it is built in to support mobile platform	Have to retrain character datasets in order for character recognition to be applicable to license plate character fonts	[7], [9]
Support Vector Machine (SVM)	High recognition accuracy	Does not works well when the license plate is tilted or under poor light condition	[10]

V. CONCLUSION

In this paper, we discussed the existing research that has implemented the ALPR system in the mobile-based platform. The methods used by each existing work in different ALPR stages are summarised in Table 4. Based on the review, it can be seen that by using a mobile device as a platform for ALPR system, most of the researches were able to achieve high accuracy of recognition rate of more than 90%. However, the processing power of a mobile device is still less superior than the PC based platform. There are still lots of room for improvements in terms of processing time. Currently, there are only a few existing researches that have implemented the ALPR system on a mobile platform. But with the availability of more mobile-optimised image processing libraries, further development of ALPR application for a mobile platform is possible.

Table 4
Summary of Existing Research in ALPR Mobile-based Platform

Ref.	Year	Methods			Problem Solved
		Localization	Segmentation	Recognition	
[5]	2012	Direct capture license plate	Horizontal projection method	neural network	Design and Implementation of LPR on Android mobile phone platform with NN
[6]	2012	Wavelet Transform	Scanning black point for character with vertical axis	Template matching	Create LPR system that can recognise license plate real time
[7]	2012	AdaBoost plus Vertical Edge LP detection	Vertical Edge Projection Histogram Segmentation (VEPHS)	Tesseract OCR with retrained data	Create LPR system featuring Android-based robot that can recognise license plate in real time
[8]	2012	Direct capture license plate	Connected pixels method	Backpropagation Feedforward neural network	NN training is separated from mobile device to speed up process and license plate information is saved in mobile database
[9]	2016	Morphological based	Contour finding & morphological operation	Tesseract OCR & Kohonen networks	Compare Tesseract OCR and Kohonen network and determine which method is more efficient in mobile platform
[10]	2017	Combination of edge-based method and colour information	Vertical projection method	Support Vector Machine (SVM)	Implementing ALPR system using SVM method for recognising Chinese car plate

ACKNOWLEDGEMENT

This project is partially sponsored by e-Science Fund 01-01-10-SF0235, Knowledge Technology Research Unit (KTRU) and Artificial Intelligence Research Unit (AiRU).

REFERENCES

[1] M. Fang, C. Liang, and X. Zhao, "A method based on rough set and SOFM neural network for the car's plate character recognition, " in Intelligent Control and Automation, WCICA 2004, Fifth World Congress on, 5, 2004, 4037-4040. IEEE.

[2] N. A. Jusoh, J. M. Zain, and T. A. A. Kadir, "Enhancing Thinning Method for Malaysian Car Plates Recognition, "in Innovative Computing, Information and Control, ICICIC'07, Second International Conference on IEEE, 2007, 378-378. IEEE.

[3] E. R. Lee, P. K. Kim, and H. J. Kim, "Automatic recognition of a car license plate using color image Processing, " in Image Processing, ICIP-94., IEEE International Conference, 2, 1994, 301-305.

[4] S. Du, M. Ibrahim, M. Shehata, and W. Badawy, "Automatic License Plate Recognition (ALPR): A State-of-the-Art Review," in IEEE Transactions on Circuits and Systems for Video Technology, vol. 23, no. 2, pp. 311-325, Feb. 2013.

[5] A. Mutholib, T. S. Gunawan, and M. Kartiwi, "Design and implementation of automatic number plate recognition on android platform," 2012 International Conference on Computer and Communication Engineering (ICCCE), Kuala Lumpur, 2012, pp. 540-543.

[6] Chuin-Mu Wang, and Ching-Yuan Su, "Fast license plate location and recognition using wavelet transform in android," 2012 7th IEEE Conference on Industrial Electronics and Applications (ICIEA), Singapore, 2012, pp. 1035-1038.

[7] H. F. Chen, C. Y. Chiang, S. J. Yang, and C. C. Ho, "Android-based patrol robot featuring automatic license plate recognition," 2012 Computing, Communications and Applications Conference, Hong Kong, 2012, pp. 117-122.

[8] R.K. Romadhon, M. Ilham, N.I. Munawar, S. Tan, and R. Hedwig," Android-based license plate recognition using pre-trained neural network," in Internet Working Indonesia Journal, 2012, 15-18.

[9] H. N. Do, M. T. Vo, B. Q. Vuong, H. T. Pham, A. H. Nguyen, and H. Q. Luong, "Automatic license plate recognition using mobile device," 2016 International Conference on Advanced Technologies for Communications (ATC), Hanoi, 2016, pp. 268-271.

[10] J. Chen, "Chinese license plate identification based on Android platform," 3rd International Conference on Computational Intelligence & Communication Technology (CICT), Ghaziabad, 2017, pp. 1-5.

[11] T. Kohonen, "Self-Organizing Maps, Springer-Verlag, " Berlin, Heidelberg, New York, 1995.

[12] C.K. On, T.K. Yao, R. Alfred, A.A.A. Ibrahim, W. Cheng, and T.T. Guan," A Comparison of BPNN, RBF, and ENN in Number Plate Recognition" in Berry M., Hj Mohamed A., Yap B. (eds) Soft Computing in Data Science. SCDS 2016. Communications in Computer and Information Science, vol 652. Springer, Singapore

[13] R. Smith, "An Overview of the Tesseract OCR Engine," Ninth International Conference on Document Analysis and Recognition (ICDAR 2007), Parana, 2007, pp. 629-633.

[14] G. Bradski, "The OpenCV Library", Dr. Dobb's Journal of Software Tools, 2000.

[15] C. Busch, R. Domer, C. Freytag, and H. Ziegler, "Feature based recognition of traffic video streams for online route tracing," Vehicular Technology Conference, 1998. VTC 98. 48th IEEE, Ottawa, Ont., 1998, pp. 1790-1794 vol.3.

[16] P. Hurtik, and M. Vajgl, "Automatic license plate recognition in difficult conditions — Technical report," 2017 Joint 17th World Congress of International Fuzzy Systems Association and 9th International Conference on Soft Computing and Intelligent Systems (IFSA-SCIS), Otsu, 2017, pp. 1-6.

[17] S. Li, and Y. Li, "A Recognition Algorithm for Similar Characters on License Plates Based on Improved CNN," 2015 11th International Conference on Computational Intelligence and Security (CIS), Shenzhen, 2015, pp. 1-4.

[18] D. N. T. How, and K. S. M. Sahari, "Character recognition of Malaysian vehicle license plate with deep convolutional neural networks," 2016 IEEE International Symposium on Robotics and Intelligent Sensors (IRIS), Tokyo, 2016, pp. 1-5.

[19] D. Yao, W. Zhu, Y. Chen, and L. Zhang, "Chinese license plate character recognition based on convolution neural network," 2017 Chinese Automation Congress (CAC), Jinan, 2017, pp. 1547-1552.

[20] M. Z. Abedin, A. C. Nath, P. Dhar, K. Deb, and M. S. Hossain, "License plate recognition system based on contour properties and deep learning model," 2017 IEEE Region 10 Humanitarian Technology Conference (R10-HTC), Dhaka, Bangladesh, 2017, pp. 590-593.

[21] R. R. Palekar, S. U. Parab, D. P. Parikh, and V. N. Kamble, "Real time license plate detection using openCV and tesseract," 2017 International Conference on Communication and Signal Processing (ICCSP), CHENNAI, India, 2017, pp. 2111-2115.