

AUTOMATIC LICENSE PLATE RECOGNITION USING IMAGE PROCESSING AND NEURAL NETWORK

P. Surekha, Pavan Gurudath, R. Prithvi and V.G. Ritesh Ananth

Department of Electrical and Electronics Engineering, PES Institute of Technology, India

Abstract

In recent times, the number of vehicles on road has exponentially risen due to which traffic congestion and violations are a menace on roads. Automatic License Plate Recognition system can be used to automate the process of traffic management thereby easing out the flow of traffic and strengthening the access control systems. In this paper, we compare the efficiency achieved by morphological processing and edge processing algorithms. A detailed analysis and optimization of neural network parameters such as regularization parameter, number of hidden layer units and number of iterations is done. Here, a scheme is designed for implementation in real time and controlled using a graphical user interface suitable for the application of parking security in offices, institutions, malls, etc. The system utilizes image processing techniques and machine learning algorithms running on matlab and Raspberry Pi 2B to obtain the results with an efficiency of 97%.

Keywords:

License Plate Recognition, Edge Processing, Vertical Projection, Horizontal Projection, Neural Network, Back Propagation Algorithm

1. INTRODUCTION

Amidst the hassles of the bustling 'smart' life that we live in, the development of a smart transportation managing system which provides the data of vehicle numbers that can be used in follow up, analyses and monitoring is quite necessary. Vehicle tracking, recognition and management are major issues to be tackled in modern traffic control systems. Automatic License Plate Recognition (ALPR) systems have their use in several in several applications of traffic and security, such as parking management, toll fee collection, access and border control, tracking of stolen cars etc. ALPR system involves capturing the image from a digital camera, pre-processing of the image to remove noise and make the image suitable for character segmentation and thereby character recognition. The output of this system is the license plate of the vehicle and the system can be tuned so as to accommodate the end user's requirement. One such user requirement that is focused in this paper is the automation of parking security and management at malls, colleges, offices, etc. The system that is developed can be used to automatically detect a vehicle coming into a parking lot and/or toll booth. An image of the vehicle is captured and the license plate is extracted. Once the characters are recognized, the user is allowed to manage the system using a GUI. The GUI provides the timestamp of the vehicle entering the premises, thereby allowing the user to monitor all the vehicles passing through the premises.

2. LITERATURE REVIEW

Vehicle license plate recognition has been an active research for the past few years. This paper aims to automate the process of

license plate recognition, improve the optimization of the neural network, to achieve prediction accuracy greater than 95% and reduce running time to about 1 second. For the purpose of localizing the LP, several techniques have been addressed. Horizontal edge detection addressed in [4] and double edge detection in [5] are two such edge detection algorithms. The techniques in [1] [4] make use of morphological operations along with other technique, however [3] specifically used only morphological processing. There are several systems already in place however, varying illuminations have been tested in [1] [3] [6] [7] and [8], but have achieved efficiency of only 84.8%, 88.71%, 80.8% and 90% respectively. The highest efficiency being 90% in [8] has a dataset consisting of only 20 images. While the others have a large dataset, their accuracy is poor. If characters are incorrectly segmented, the Optical Character Recognition (OCR) would be affected and accurate results will not be achieved. Algorithms such as connected component method in [1][3][7][8] and horizontal projection in [4][5] are used for the purpose of segmenting. OCR technique is a widely used technology that translates printed text into machine encoded text. While in [6] and [10] neural networks has been implemented all the other papers listed in the references makes use of OCR based on template matching method in order to predict. These systems have been executed for specific font styles only. License Plate Recognition becomes easy if executed on standard plates, however with different countries, the stroke width and the size of plate vary. Therefore it becomes necessary to tackle such issues at places where there aren't any strict rules and/or regulations aren't followed. All theoretical studies have explored ALPR systems using standard processing algorithms and template matching method for character recognition. Only recently artificial intelligence technique such as Artificial Neural Network (ANN) is being employed. ANNs are widely used intelligent computing architecture for pattern recognition. Typically, the work done in [6] and [10] employs feature extraction and binary pixel values to organize the inputs of feed-forward neural network respectively. The processing techniques used in [1] and [3] take 0.5 seconds and 0.3 seconds respectively. However, papers with algorithms having accuracy greater than 90% have not mentioned the processing time. There is always a trade-off between the computational time and the efficiency with which the algorithm works, however we try to achieve maximum efficiency on a dynamic environment with a computational time of about 1.3 seconds and on a static image of 0.4 seconds. With the existence of a myriad image processing algorithms, the use of these techniques have been found amongst the recent literatures.

3. PROPOSED METHODOLOGY

The proposed system focuses upon the security of parking at any premise. Therefore, the crucial step is for the system to log

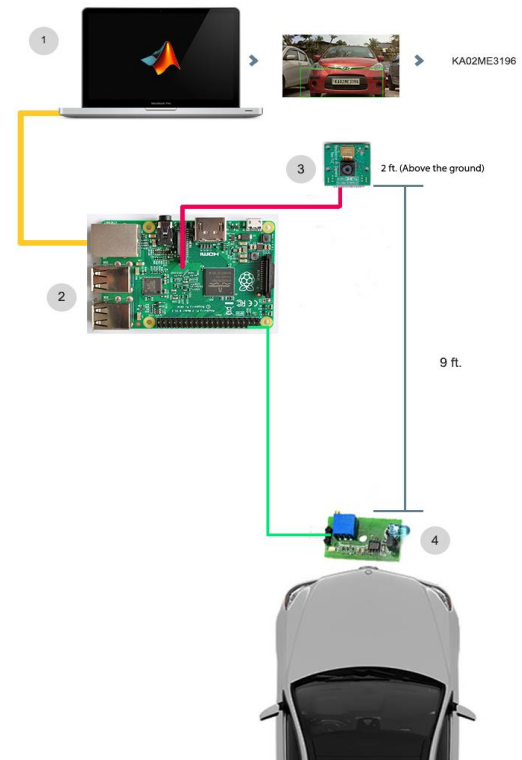
the time and details of the license plate when the vehicle enters the premises. While any ALPR system consists of an image acquisition system and a character prediction network, this system makes use of a sensor network to activate the camera and a graphical user interface (GUI) for the user to be able to control the entire setup. The system is built using Raspberry Pi 2 Model B and its camera, Raspberry Camera Module V1. TSOP 1738 IR sensor was used to detect the presence of a vehicle and the software for the system is built using MATLAB 2016b. With MATLAB Support Package for Raspberry Pi hardware, it is possible to remotely communicate with the Raspberry Pi and use it to control peripheral devices. This support allows the system to acquire data from sensors and imaging devices connected to the Raspberry Pi. The system is represented pictorially in Fig.1. Several preprocessing techniques such as cropping ROI, contrast enhancement, dilation, vertical and horizontal edge processing, and character extraction using connected component analysis are applied. Once the characters are extracted, they are passed as an input to the neural network, implemented to predict the characters. The neural network is trained using the feed-forward backpropagation algorithm. The network consists of 1650 input units, 500 hidden units and 36 output units. Initially the network was trained with 25 hidden layer units. 500 hidden layer neurons was chosen after optimizing the neural network with respect to regularization parameter, number of training steps and number of hidden layer neurons.

The following assumptions are to be taken care off while implementing the proposed system:

- Network has been trained for 26 English alphabets (A-Z) and 10 Numbers (0-9).
- A region of interest has been pre-defined depending upon the environment at which the system is implemented. Here, a flat surface environment is considered with the image of the vehicle captured from a distance of 9ft (margin of +2ft and -1ft) and the camera being mounted at a height of 2ft above the ground.
- Single vehicle entry. This system was designed for cars and can be extended to other vehicles by varying the Region of Interest.
- Speed of car above the sensor is between 10-20 km/h.

3.1 IMAGE ACQUISITION

This system assists the algorithm to run in a dynamic environment and aids the system in license plate localization and character extraction. An image acquisition system is considered appropriate if the image provided by it is a stable and balanced image under all of its working conditions. It comprises of the imaging system and the sensor network. Using this system, a dataset of 105 cars was created. The imaging system makes use of a Raspberry Pi 2 Model B connected to a Raspberry Camera Module V1 having a resolution of 5 MP (2592×1944). The sensor network comprises of a TSOP 1738 which has high immunity against ambient light [12]. The primary purpose of our sensor network is to detect the movement of the incoming vehicle and initiate a trigger for the Raspberry camera to capture the image.



- 1 → Computer running MATLAB®
 2 → Raspberry Pi 2 model B
 3 → Raspberry Camera Model V1
 4 → TSOP 1738 IR Sensor

Fig.1. Proposed Methodology

3.2 IMAGE PRE-PROCESSING

Pre-processing an image plays a crucial role in enhancing the quality of an image so that necessary features can be extracted and used for further analysis. The image undergoes a series of pre-processing techniques as follows. A region of interest is pre-defined based on the location i.e. flat, inclined or declined surface, at which the system is being implemented. The interested region is cropped and converted to grayscale format since it would reduce processing time as well as the complexity of the algorithm. It is then enhanced in order to make the edges more distinct. When the image is converted from RGB to grey image, certain important parameters like difference in colour, lighter edges of object, etc. may be lost. The process of dilation nullifies such losses and enhances the edges. The Fig.2 shown below gives us the histogram of the images before and after the process of dilation. The difference between the histograms before and after dilation is shown in Fig.3 thereby illustrating the requirement for enhancing edges.

3.3 LICENSE PLATE LOCALISATION

License plate localization is the step where we bound the exact license plate from the larger scene. In this paper, we compare the two processes of localising the license plate i.e. morphological processing and edge processing.

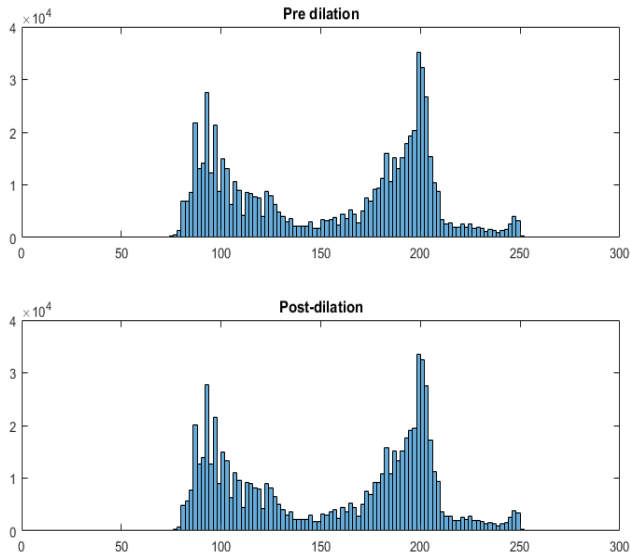


Fig.2. Image dilation histogram

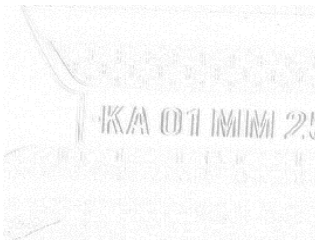


Fig.3. Effect of dilation

3.3.1 Morphological Processing:

Morphology is defined as a particular shape or structure and when employed in image processing, it is a collection of non-linear operations related to the shape in an image. Morphological techniques are applied on an image with a small template known as a structuring element. This structuring element is centred on all the pixels of the image and mathematical operations are carried with its corresponding neighbourhood of pixels. A structural element of diamond was used to dilate the image. This process enhances the bright regions surrounded by dark regions and reverse is true for dark regions surrounded by bright regions [1]. Further, a disk was used to erode the image. A morphological algorithm was employed on the database by applying canny edge detection, and henceforth being able to localize the plate. However, it was found that this algorithm executed with a relatively lower percentage of 83%. It was found that this algorithm did not work effectively on white cars having white background number plates since the license plates did not appear to have a definite edge. The edge of the number plate blends with the colour of the car, thereby not giving a distinct edge when canny edge detection is applied. The algorithm failed for different illuminating environments. Therefore, this algorithm was discarded and edge projection technique was employed, which improved the efficiency of the algorithm significantly.

3.3.2 Edge Processing:

Edges are boundaries that indicate sudden changes of discontinuities. In a license plate there is a transition from relatively dark to light regions, i.e. characters and white spaces,

and these transitions are considered to be edges. Edge processing is the technique where histograms of the edges processed are plotted. These edges are processed in both horizontal and vertical directions in order to provide a probable region of the license plate.

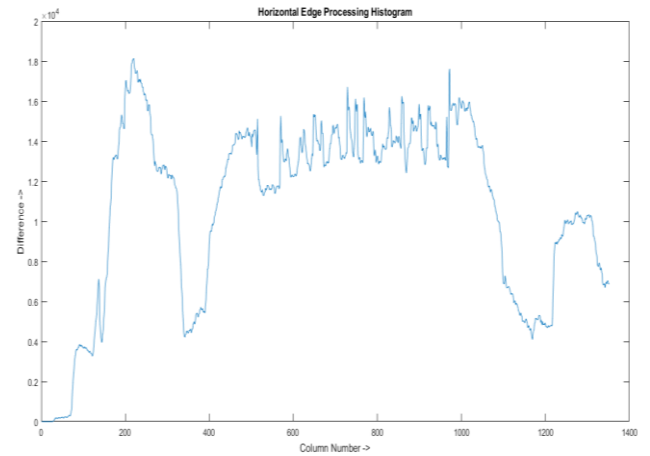


Fig.4. Horizontal edge histogram

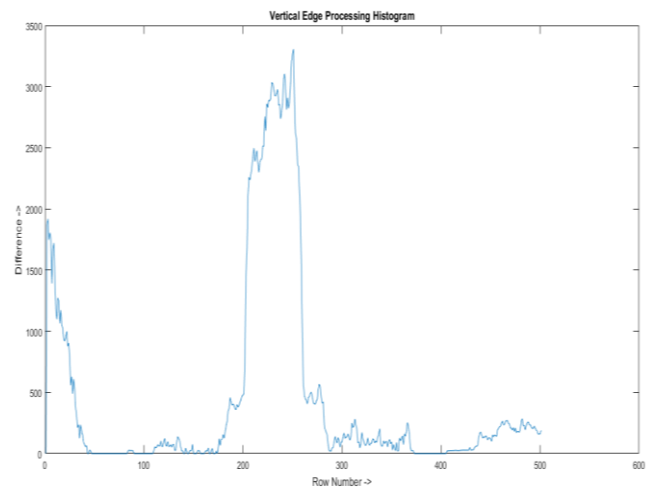


Fig.5. Vertical edge histogram

As shown in Fig.4, a histogram for the edges processed in the horizontal direction is plotted and it is then passed through an averaging filter to smoothen the varying transitions. The smoothened histogram is passed through an averaging process that highlights the regions that could contain the license plate. The search is further narrowed down by incorporating the largest column in the histogram. Furthermore, a histogram is plotted, as shown in Fig.5, for the edges processed in the vertical direction and a similar process is followed as described above to obtain the license plate. In order to avoid boundary clipping effects, i.e. situations wherein the process of region clipping could remove certain characters from the license plate that are useful information, the image region obtained is dilated as shown in Fig.6.



Fig.6. Dilated License plate to avoid boundary clipping

3.4 CHARACTER EXTRACTION

The extracted characters from the license plate are the input layer units of the neural network. Most Indian license plates contain an 'IND' hologram as per the High Security Registration Plates. To segment the characters, the image is converted to binary by applying a global threshold obtained from Otsu's method [11]. These irrelevant features are removed by area opening and border clearing.

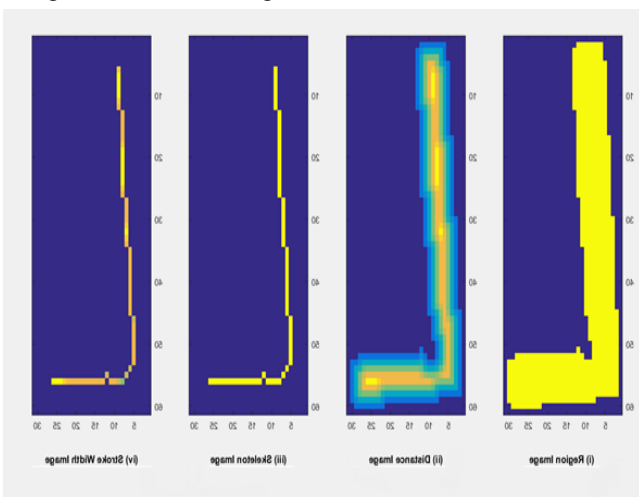


Fig.7. Stroke width variation of a character

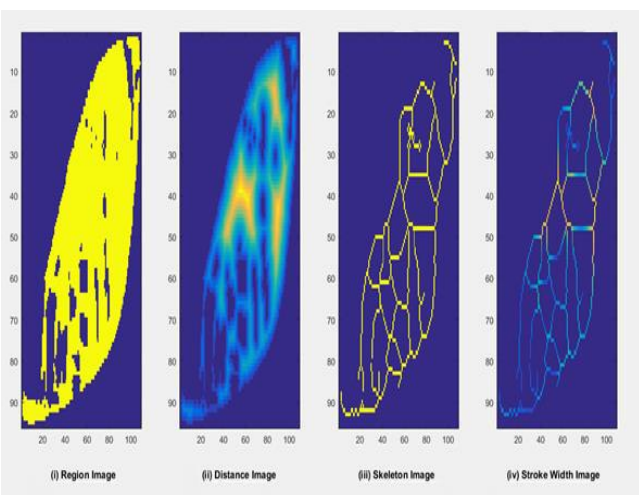


Fig.8. Stroke width variation of a fog lamp

The characters are extracted by applying connected component analysis after undergoing constraints such as aspect ratio and stroke width property. The Fig.7 and Fig.8 represents the

stroke width constraint that is applied on a relevant character and an irrelevant feature (fog lamp) respectively. The license plate before and after undergoing aspect ratio constraint is as shown in Fig.9.

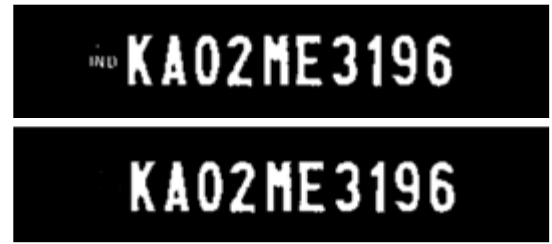


Fig.9. Aspect ratio constraint on a license plate

3.5 NEURAL NETWORK

Amongst the several techniques of training the neural network, this paper has been implemented based on supervised learning. In order to predict the characters, the training set has an input vector 'X' and a corresponding desired output value 'Y'. This network makes use of three layers i.e. an input, hidden and output layer. Feed- forward backpropagation algorithm has been used to train the network. The training set was created from the license plate database that was used for plate localization. Also, a variety of fonts that are inbuilt in Microsoft Word have been incorporated into the training set. The database consisted of 1391 characters. The neural network was trained using 70% of these characters and remaining 30% of these characters were used for cross-validation. The dataset was created by extracting the characters from the license plate and resizing them into a 55x30 binary image. At first, the network had a single hidden layer having 25 units. The input layer consisted of 1650 units while the output layer had 36 units. The neural network was further optimized based on certain parameters as explained in the following section.

3.5.1 Neural Network Optimization:

Different learning settings such as regularization, number of training steps and number of hidden layers for the network can impact the performance of the neural network. After training a neural network for more iterations or varying the regularization parameter λ , it is possible to get the neural network to perfectly fit the training set.

In order to obtain an optimal value for λ , The Fig.10 shown below plots a graph of average accuracy of the cross-validation set versus λ and the corresponding cost function value versus λ respectively. From the plot, it is clear that as the regularization parameter is increased more features are discarded resulting in a decrease in prediction accuracy. At the same time the cost of the neural network goes on increasing which implies that the mapping of training samples to the target output is poor. $\lambda = 0$, the network accuracy of 97% is obtained and the cost function value is zero. While this may be ideal, it is a case of high variance which means the stability of the network to respond to new samples is poor. When $\lambda = 5$ the network accuracy obtained is close to 95%, and hence this value of λ is chosen for training the neural network. The number of training steps was also used to optimize the network. By increasing the number of iterations, the network can be trained to achieve minimal cost which is an indicator that the

algorithm is working correctly. The result of this analysis is shown in Fig.11.

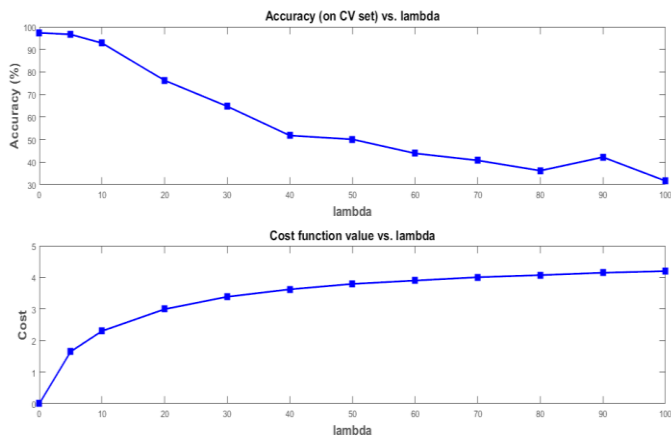


Fig.10. Regularization parameter - Analysis

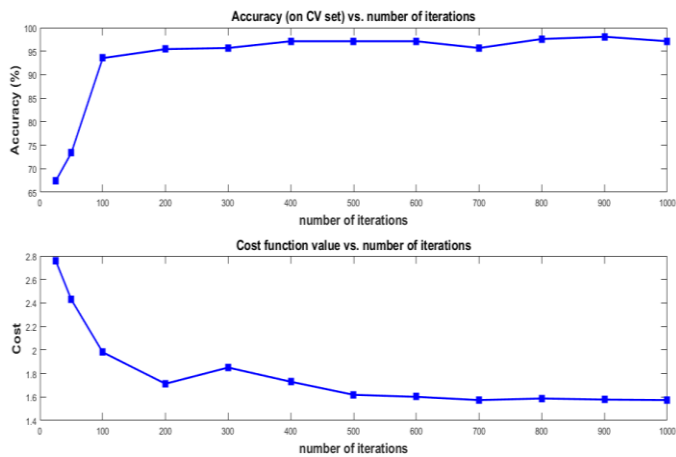


Fig.11. Number of iterations - Analysis

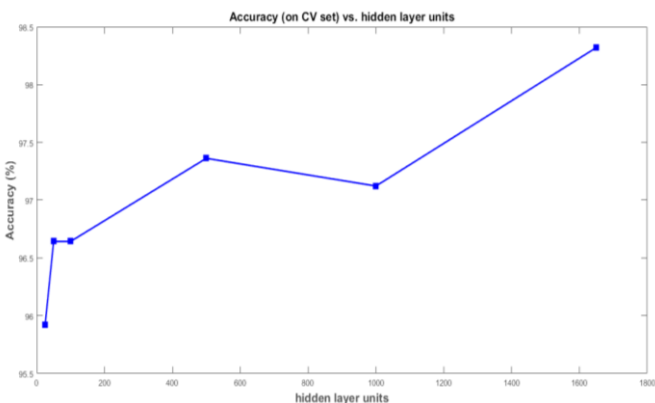


Fig.12. Number of hidden layer units - Analysis

From this plot, it can be observed that, for number of iterations greater than 500, the cost remains almost constant. Likewise the accuracy also remains constant at nearly 95%. Therefore an optimal value of 500 iterations is considered because the time required for training the network increases with increase in number of training steps. The number of hidden layer neurons is the final parameter that was varied to optimize the neural network.

This number can vary from one up to the number of input units. For the network to generalize, the number of nodes in the hidden layer has to be kept as minimal as possible. As the number of nodes is increased, the network predicts samples from the training set to perfection but does not perform well on samples that were not part of the training set. The result of this analysis is shown in Fig.12.

It can be observed that as the number of hidden layers is increased, the accuracy increases. When the number of neurons in the hidden layer is same as the number of neurons in the input layer, the accuracy of the network is 98.25%. But the time required for training increases with increase in number of hidden layer neurons. Hence, an optimal value of 500 neurons was chosen for the hidden layer.

3.6 GUI

The graphical user interface is developed on matlab using 'guide' for a better user experience for the end user operating the software. The GUI developed is specific to a parking lot application. The GUI consists of two tables, an alert table and an entry table along with options to start, stop and open camera as shown in Fig.15. The interface also provides an option to export the tables to an excel sheet. The entry table displays the region at which the camera captures, the license plate number along with the confidence with which it predicts and the time at which the car enters the premises. It also provides a link to the image of the car that was captured. The alert table comprises of similar columns as explained above. The number of characters on a vehicle license plate in India varies between to 8-10. The algorithm checks for certain conditions and if it does not satisfy any one amongst them, then the data is logged into the alert table. The user who's monitoring has the ability to edit the alert table entries as and when the license plate is logged in.

The conditions are as follows:

- Confidence of individual characters must be greater than 40%.
- Mean confidence of all the license plate characters must be greater than 75%.
- Number of characters that are predicted must be between 8 and 10.

4. RESULTS AND DISCUSSION

The system was tested on a set of 105 static images as well as in a dynamic environment. The following sections describe the results based on the test conducted on the static images. The execution time of the system on a dynamic environment is 1.3s and the license plate of the vehicle can be skewed up to an angle of 10 degrees.

4.1 LICENSE PLATE LOCALIZATION EFFICIENCY

The Fig.14 shown below illustrates the way in which license plate localization is carried out. At first, the motion of the vehicle is detected over the sensor and is displayed on the command window. After all the processing is done, the image of the license plate is displayed. Two types of processing takes place in morphological processing that localizes the license plate from the

vehicle at an efficiency of 83%, at the rate of 0.7 seconds. Whereas, edge processing technique localizes the license plate at an efficiency of 100% and at the rate of 0.15 seconds. The results are presented in Table.1.

Table.1. License Plate Localization Efficiency

Method	Total Samples	Plate Localization	Efficiency (%)
Morphological Processing	105	87	83
Horizontal and vertical edge projection	105	105	100

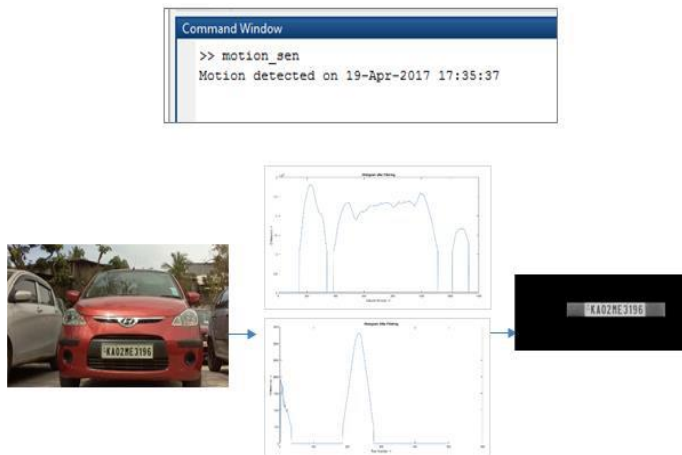


Fig.14. License Plate Localization

4.2 NEURAL NETWORK PREDICTION EFFICIENCY

The 105 static images of the vehicle were tested and the exact license plate was detected on 102 of those images with an efficiency of 97%.

4.3 GRAPHICAL USER INTERFACE

The graphical user interface displays the output of the entire system. It shows the license plates that have been detected successfully and also displays those that have been moved into the alert table.

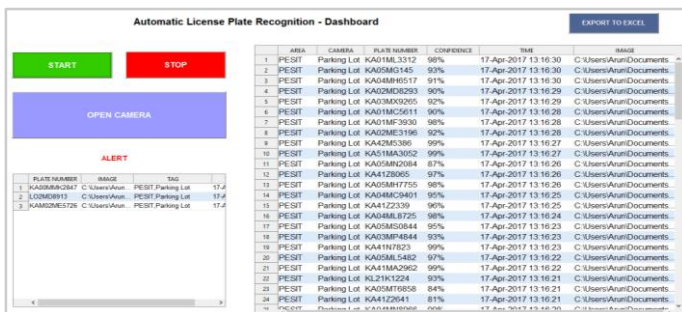


Fig.15. Graphical User Interface

The Fig.15 illustrates the list of static license plates that were detected. From a set of 105 static images, 102 vehicles' license

plates were successfully predicted while 3 of them were moved into the alert table.

4.4 COMPARATIVE ANALYSIS

The work done in this system can only be judged upon by comparing it with systems that are aiming to achieve a solution to a common end user problem. Using the most recent journals and conference papers from the year 2015, Table.2 shows the efficiency of the proposed system with other recent approaches in literature.

Table.2. License Plate Localization Efficiency

Method	Total Samples	Predicted Samples	Efficiency
Proposed	105	102	97%
[1]	500	424	84.8%
[3]	150	138	92%
[7]	Unknown	Unknown	80.8%
[8]	20	18	90%

5. CONCLUSION

The implementation of ANPR system for parking lot application was achieved using MATLAB and Raspberry PI hardware. At first, morphological processing was employed for license plate localization and the results were analyzed in terms of plate localization efficiency. It was found that the algorithm poorly performed under brightly illuminated environments. This was followed by a careful study and implementation of edge detection algorithm which gave a drastic improvement in the plate localization efficiency. This was followed by an implementation of neural network using the feed-forward backpropagation algorithm. The parameters of the neural network were optimized to enhance the performance of the neural network. The system was tested in a dynamic environment, wherein it automatically detected the movement of the vehicle above the sensor and plate localization and character recognition was performed successfully in a time span of 1.3 seconds. This system works for skewed angle up to 10 degrees.

6. FUTURE SCOPE

The system could be capacitated to capture images from a large distance and at wider skewed angle. ALPR system should have the ability to predict multiple license plates when multiple vehicles are present on a single frame. Deep learning could be enforced in order to predict the presence of a vehicle.

REFERENCES

[1] N. Saleem, H. Muazzam, H.M. Tahir and U. Farooq, "Automatic License Plate Recognition using Extracted

- Features”, *Proceedings of 4th International Symposium on Computational and Business Intelligence*, pp. 221-225, 2016.
- [2] K. Makaoui, Z. Guennoun and M. Ghogho, “Improved License Plate Localization”, *Proceedings of IEEE International Conference on Electrical and Information Technologies*, pp. 402-405, 2016.
- [3] R. Islam, K.F. Sharif and S. Biswas, “Automatic Vehicle Number Plate Recognition using Structured Elements”, *Proceedings of IEEE International Conference on Systems, Process and Control*, pp. 44-48, 2015.
- [4] P. Prabhakar, P. Anupama and S.R. Resmi, “Automatic Vehicle Number Plate Detection and Recognition”, *Proceedings of IEEE International Conference on Control, Instrumentation, Communication and Computational Technologies*, pp. 185-190, 2014.
- [5] J. Chong, C. Tianhua and J. Linhao, “License Plate Recognition based on Edge Detection Algorithm”, *Proceedings of 9th IEEE International Conference on Intelligent Information Hiding and Multimedia Signal Processing*, pp. 395-398, 2013.
- [6] K.M. Hung and C.T. Hsieh, “A Real-Time Mobile Vehicle License Plate Detection and Recognition”, *Tamkang Journal of Science and Engineering*, Vol. 13, No. 4, pp. 433-442, 2010.
- [7] A. Puranic, K.T. Deepak and V. Umadevi, “Vehicle Number Plate Recognition System: A Literature Review and Implementation using Template Matching”, *International Journal of Computer Applications*, Vol. 134, No. 1, pp. 12-16, 2016.
- [8] P. Sai Krishna, “Automatic Number Plate Recognition by using Matlab”, *International Journal of Innovative Research in Electronics and Communications*, Vol. 2, No. 4, pp. 1-7, 2015.
- [9] M.S. Pan, J.B. Yan and Z.H. Xiao, “Vehicle License Plate Character Segmentation”, *International Journal of Automation and Computing*, Vol. 5, No. 4, pp. 425-432, 2008.
- [10] X. Zhai, F. Bensaali and R. Sotudeh, “OCR-based Neural Network for ANPR”, *Proceedings of 9th IEEE International Conference on Imaging Systems and Techniques*, pp. 393-397, 2012.
- [11] N. Otsu, “A Threshold Selection Method from gray-Level Histograms”, *Automatica*, Vol. 11, No. 2, pp. 23-27, 1975.
- [12] Photo Modules for PCM Remote Control Systems, Available at: <https://media.digikey.com/pdf/data%20sheets/vishay%20ir%20pdfs/tsop%2017...pdf>